

# Q<sub>2</sub>E: Query-to-Event Decomposition for Zero-Shot Multilingual Text-to-Video Retrieval

Shubhashis Roy Dipta, Francis Ferraro

Department of Computer Science and Electrical Engineering  
University of Maryland Baltimore County  
Baltimore, MD 21250 USA  
[{sroydip1,ferraro}@umbc.edu](mailto:{sroydip1,ferraro}@umbc.edu)

## Abstract

Recent approaches have shown impressive proficiency in extracting and leveraging parametric knowledge from Large-Language Models (LLMs) and Vision-Language Models (VLMs). In this work, we consider how we can improve the retrieval of videos related to complex real-world events by automatically extracting latent parametric knowledge about those events. We present **Q<sub>2</sub>E**: a Query-to-Event decomposition method for zero-shot multilingual text-to-video retrieval, adaptable across datasets, domains, LLMs, or VLMs. Our approach demonstrates that we can enhance the understanding of otherwise overly simplified human queries by decomposing the query using the knowledge embedded in LLMs and VLMs. We additionally show how to apply our approach to both visual and speech-based inputs. To combine this varied multimodal knowledge, we adopt entropy-based fusion scoring for zero-shot fusion. **Q<sub>2</sub>E** outperforms the previous SOTA on the MultiVENT dataset by **8 NDCG points**, while improving on MSR-VTT and MSVD by **4 and 3 points**, respectively, outperforming several existing retrieval methods, including many fine-tuned and SOTA zero-shot approaches. We have released both code and data.<sup>1</sup>

## 1 Introduction

Making sense of complex real-world events, such as a natural disaster or fire, requires understanding more than just one individual event or action. In addition to the multiple events that occur during that broader event (such as structures burning or evacuation), one might need to understand the events that lead up to it (such as dry conditions), or that arise from it (such as recovery and rebuilding). Videos present a rich medium for conveying this multi-faceted information, as they contain much more than what is visible at any given point in time: they can show changes and dynamic actions, and

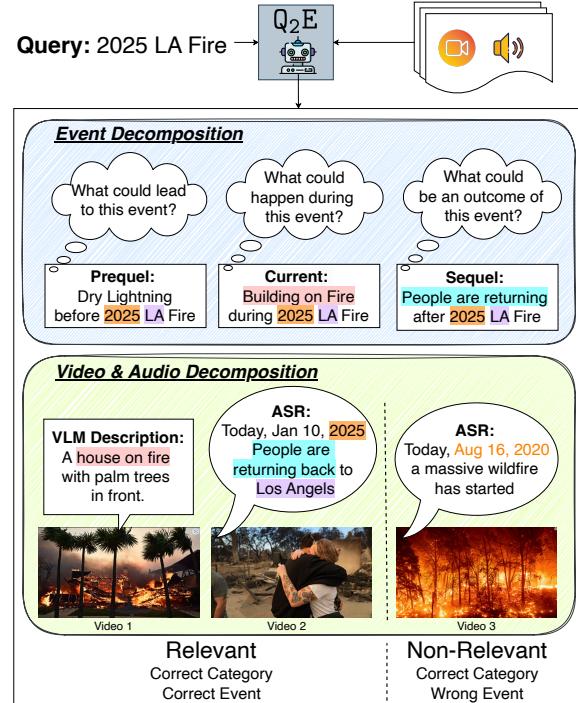


Figure 1: In **Q<sub>2</sub>E**, we extract prequel, current, and sequel events, along with an audio transcript and video description to enrich the query and video context, respectively. This decomposed queries are matched across visual, textual, and speech-based descriptions (matching phrases are highlighted in the same color) enabling the retrieval of the correct videos while effectively filtering out a visually similar but non-relevant video.

through an audio track, allow for natural language descriptions, either in-the-moment or after-the-fact, about that event. Being able to automatically retrieve videos about a queried event could provide the information necessary for users to better make sense of events they need to know about. However, there are multiple challenges to this: first, while a query might be a concise reference to a particular event, such as “2025 LA fire,” a user would still want all this information. Second, this information might be scattered across multiple videos, with any given video only showing a portion of the informa-

<sup>1</sup><https://dipta007.github.io/Q2E/>

tion needed. Third, these videos might not be in the language that the user knows.

We present a novel zero-shot method, **Q<sub>2</sub>E**, that addresses these problems. It consists of a proposed event decomposition method that **leverages the embedded knowledge that LLMs have about those types of complex events** and uses this knowledge to **automatically decompose and elaborate (or expand) a query**. We show how to use this knowledge to **identify and score salient visual and multi-lingual audio-based features from the video**, and combine this information to return the sought after videos.

To achieve this, **Q<sub>2</sub>E** integrates the pre-existing knowledge of LLMs and VLMs to enhance text-to-video retrieval. Underlying our approach are three key insights: **(1)** Transferring knowledge from LLMs through query decomposition enhances the understanding of coarse-grained human queries. This provides richer context and enables the retrieval of videos showing what can lead to or result from the target event that would otherwise be overlooked. **(2)** While VLM-based captioning and large ASR models can generally describe an image/video frame and transcribe a portion of speech, the outputs can be repetitive, noisy, missing broader themes that occur across frames. However, a LLM based refiner can help to remove those noises and redundancies for empirical effectiveness. **(3)** In our approach, determining whether an individual video is relevant requires aggregating across multiple forms of similarity and relevancy judgments, each of varying strength. Entropy-based rank fusion provides an intuitive way of performing this aggregation and ranking, while surpassing both simple fusion methods (mean, max) and more complex approaches like Reciprocal Rank Fusion. Using these insights, **Q<sub>2</sub>E** achieves strong performance gains on three varied and challenging datasets (Chen and Dolan, 2011a; Xu et al., 2016; Sanders et al., 2023).

**But isn't video retrieval solved?** Widely used video retrieval websites can give the impression that video retrieval is a solved problem: a user types in a query, and gets back a nicely curated list of relevant videos. However, such approaches can leverage rich platform-specific metadata or human-annotated data (like titles or search-optimized descriptions) that go beyond the information conveyed *directly* in that video. For example, as shown in Fig. 1, there are many different videos of *fires*, but if a user provides a basic query such as “2025 *LA fire*” in order to find videos providing infor-

mation about *that* specific fire, how can a system satisfy this request? Most current text-to-video retrieval systems are trained and evaluated on classic datasets, such as MSR-VTT (Xu et al., 2016) and MSVD (Chen and Dolan, 2011a), which feature generic, high-level queries like “*a person is explaining something*” (a random query from MSR-VTT), rather than referencing a complex real-world event. Recent work (Liu et al., 2024b) has also found that users typically input basic queries, expecting the system to capture “fine-grained semantic concepts” and understand complex relationships between videos and text. Finally, Sanders et al. (2023) demonstrate the disparity that exists when retrieving non-English videos. Our work addresses these challenges by showing how to automatically expand the query with information needed to understand the event, and how to get video+audio features reflective of this information to retrieve highly specific videos across different languages. In doing so, we show how the extensive knowledge embedded in Large-Language Models (LLMs) and Vision-Language Models (VLMs) can be effectively transferred to text-to-video retrieval.

In summary, our contributions are:

- We propose a novel method that leverages the embedded world knowledge of LLMs to enrich human queries with prequel, current, and sequel-based events.
- We use large video, speech, and language models to capture critical event- and query-relevant information communicated across those multiple modalities, how to assess that information for query relevance, and how to combine those assessments via entropy-based rank fusion.
- Through extensive ablation studies, we have shown the importance of each component of our method in the final performance.
- Our zero-shot method improved performance on the widely used text-to-video retrieval dataset, MSR-VTT (**+4 NDCG**) & MSVD (**+3 NDCG**) as well as on the multi-lingual event-focused retrieval dataset, MultiVENT (**+8 NDCG**).

## 2 Q<sub>2</sub>E: Query-to-Event Decomposition

Modern dense retrieval approaches commonly compute an embedding-based similarity (e.g., cosine) between a query and each candidate; those scores rank candidates, and the top-scoring candidates are returned. For example, in video retrieval, one

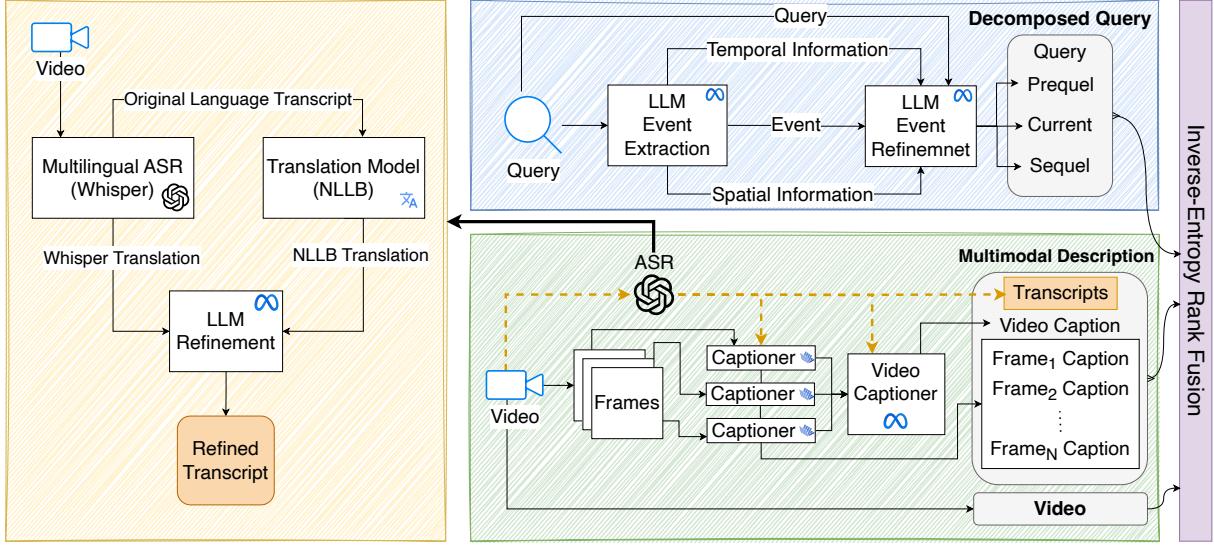


Figure 2: **Q<sub>2</sub>E**, the complete framework of our text-to-video retrieval system. The blue box represents the event decomposition module, while the green box illustrates the video decomposition module. The orange box represents our multi-layer Audio Decomposition module. The purple box fuses the ranks calculated from query-video, query-descriptions, and event-descriptions (event = prequel + current + sequel, description = multimodal description). In the non-ASR variant, the components within the orange box and orange dotted lines are excluded.

could embed the user’s query using an appropriate LLM, embed the video using an appropriate video embedder, and then compute the cosine similarity between those embeddings (Sanders et al., 2023). This “query vs. video” similarity score is used to determine how relevant to video is to the query.

**Q<sub>2</sub>E** (shown in Fig. 2) builds upon this general embedding similarity computation. To better capture the diverse and nuanced multimodal information conveyed in videos and human queries—including the event decompositions—we must extend those similarity computations. In order to capture the rich event information we get from our decomposition (§2.1), a single embedding of the video is not sufficient. We need to additionally get *multimodal descriptions* of the video (§ 2.2 and 2.3). Using these components, **Q<sub>2</sub>E** computes five types of scores  $S_{i,v}$  for each video  $v$ : (a) a query vs. video score, as described above; (b) prequel, (c) current (d) sequel vs. multimodal-description score and (e) a query vs. multimodal-description score. While (a) helps to identify coarse-grained features such as “fire”, (b-e) provide fine-grained contextual details. These scores complement one another, which we merge through entropy-based fusion and ranking (§2.4).

## 2.1 Event Decomposition

Like any traditional text-video retrieval system, a user provides a natural language query, which can

be a couple of words to 10+ words. In **Q<sub>2</sub>E**, we have used an LLM to decompose it into three sub-events: (1) Prequel, (2) Current, and (3) Sequel. Based on initial experiments, and to manage computation, we only considered five of prequel, current, and sequel events. We provide our template prompts for each in App. A.6.

**Prequel** “Prequel” refers to events that can lead to the current event described in the query. An event can have multiple prequels; for example, a wildfire can have strong wind, high-temperature alert, and dry lightning as prequels. Prequel events may *enable* the event in the query, but do not necessarily *cause* it.

**Current** “Current” mainly refers to the simple decomposed events that occur during the event described in the query. This method decomposes the queries into sub-events likely to be observed. E.g., “2025 LA Fire” can be decomposed into “Building on Fire”, or “Smoke everywhere.”

**Sequel** “Sequel” events can be the results from the event described in the query. Like the prequel, one event can lead to multiple sequel events.

**Decomposition Refinement** We refine these event decompositions with temporal, spatial, and primary event data. For this, we have used the same LLM to extract the temporal (2025), spatial (LA), and primary event (Fire) information. Although

this rich information can be combined template-  
ically, e.g., ‘‘P/S/C (2025, LA, Fire),’’ during initial  
development experiments, we found that refining  
it to read as a more natural, human-provided query  
markedly improved performance. We produced  
this refined query automatically using an LLM; we  
show this in App. A.6.3.

## 2.2 Video Decomposition

In practice, we found that embeddings of videos / video frames were not sufficient to capture the varied actions described by our event decomposition through basic similarity computations. While alternative similarity approaches, such as cross-encoders, may be effective, they present computational challenges, requiring running the cross-encoder on every video/query pair. To address this, we instead extract captions from the videos themselves, with the aim that these descriptions would be sufficiently similar to our query event decomposition. We found two approaches to be effective for capturing complementary local and global elements across the video: (1) generating contextualized descriptions per selected frame, and (2) generating holistic video descriptions.

**Contextualized Frame Descriptions** We uniformly sampled 16 frames from the videos (determined by trading off performance vs. computation—ablation in App. A.3.3). We have tried uniform sampling and a scene-change-based sampling method<sup>2</sup> for frame sampling, and found that uniform sampling, despite its simplicity, works better than scene-change-based sampling (ablation in App. A.3.4). As the video contains temporal information and by design, video frames should be connected semantically, so rather than captioning each frame in isolation, we used a sliding window (window size=2) approach and provided the VLM model with the previous frame’s caption (when available). Conditioning on previous frame’s captions helped contextualize subsequent captions. Per video, we have K contextualized frame captions.

**Video Descriptions** We found that current VLM models can perform dense captioning but miss the ‘‘bigger picture.’’ For instance, a VLM might describe a video of a wildfire with *people running* and *fire everywhere* but fail to recognize the higher-level conclusion of ‘‘wildfire.’’ To address this limitation, we provided an LLM model with all

16-frame descriptions from above and asked it to summarize them into a single video-dense caption, preserving the order of the frame captions. We carefully designed the prompt (App. A.6.5) such that it preserves temporal information and focuses on broader picture rather than color, shape or objects.

## 2.3 Audio Decomposition

Speech is a critical component of many videos, especially when visual elements alone may not be sufficient enough to convey the full context. Despite using a multilingual ASR model (Radford et al., 2023), the multilingual videos posed a unique challenge to get consistent and accurate ASR transcriptions due to the background noise and different dialects of the real life videos in MultiVENT. To address this, we propose a multi-layer translation pipeline (Fig. 2, left orange box). This pipeline leverages three models: (1) a multilingual ASR agent, whisper-v3 (Radford et al., 2023), which transcribes audio into both the original language and its English translation; 2) a translator agent, NLLB (Costa-jussà et al., 2022) to generate English transcriptions by translating the original transcript; and (3) a refiner agent, Llama-70B (Dubey et al., 2024), which refines both the English translations. Results showed all three are critical for sufficient transcription and translation quality (App. A.3.2).

## 2.4 Scoring and Score Fusion

**Query vs. Video** We utilized the original query and video to compute a similarity score. To do this, we follow Sanders et al. (2023) in adapting an image encoder: we uniformly sample and embed 16 frames, averaging their embeddings to generate the final video embedding. Cosine similarity was applied to compute the similarity score, which was then scaled to the 0–100 range to ensure comparability with other scores:  $S_{i,v}|Q = 100.0 \times \text{Sim}_v(Q, V)$ , where  $S_{i,v}|Q$  denotes the score vector for a given query  $Q$ , and  $\text{Sim}_v(x, y)$  returns the similarity score between query  $x$  and video  $y$  using an embedding model. Primarily we have used MultiCLIP (Delitzas et al., 2023) due to its superior performance with multilingual videos compared to the CLIP (Radford et al., 2021) model, although we show in our experiments that our approach successfully uses other image encoders too.

**Query vs. Multimodal Descriptions** In this approach, we used the original query but replaced the video content with the Multimodal Descrip-

<sup>2</sup><https://www.scenedetect.com/>

Model	Knowledge Augmentation	R@1↑	R@5↑	R@10↑	P@1↑	P@5↑	P@10↑	MRR↑	NDCG↑	MAP↑	MnR↓	MdR↓
<b>MultiVENT</b>												
MultiCLIP	-	9.83	44.32	70.82	88.80	80.77	65.25	0.92	75.34	86.33	22.12	<b>6</b>
MultiCLIP	Event	10.24	46.71	75.76	92.28	85.25	69.73	<b>0.95</b>	80.04	89.42	21.13	<b>6</b>
MultiCLIP	ASR + Event	<b>10.32</b>	<b>49.00</b>	<b>79.60</b>	<b>93.05</b>	<b>88.73</b>	<b>73.09</b>	<b>0.95</b>	<b>83.24</b>	<b>91.20</b>	<b>16.44</b>	<b>6</b>
InternVideo2-1B	-	5.60	28.92	49.12	51.35	53.28	45.44	0.68	50.43	63.77	235.49	11
InternVideo2-1B	Event	9.54	40.88	63.40	87.26	75.21	58.73	0.92	69.15	83.96	53.84	7
InternVideo2-1B	ASR + Event	<b>10.24</b>	<b>44.94</b>	<b>70.79</b>	<b>92.28</b>	<b>81.85</b>	<b>65.14</b>	<b>0.95</b>	<b>76.10</b>	<b>88.09</b>	<b>42.81</b>	<b>6</b>
<b>MSR-VTT-1kA</b>												
MultiCLIP	-	43.52	69.05	76.88	43.62	13.85	7.71	0.54	59.72	54.27	20.29	2
MultiCLIP	Event	44.52	71.26	79.40	44.62	14.29	7.96	0.56	61.51	55.84	<b>17.91</b>	2
MultiCLIP	ASR + Event	<b>46.23</b>	<b>73.37</b>	<b>81.71</b>	<b>46.33</b>	<b>14.71</b>	<b>8.19</b>	<b>0.58</b>	<b>63.59</b>	<b>57.83</b>	18.73	2
InternVideo2-1B	-	52.56	73.07	80.10	52.66	14.65	8.03	0.62	66.07	61.62	25.12	<b>1</b>
InternVideo2-1B	Event	53.47	73.57	82.11	53.57	14.75	8.23	0.62	67.16	62.47	16.73	<b>1</b>
InternVideo2-1B	ASR + Event	<b>56.28</b>	<b>76.58</b>	<b>83.72</b>	<b>56.38</b>	<b>15.36</b>	<b>8.39</b>	<b>0.65</b>	<b>69.53</b>	<b>65.06</b>	<b>15.85</b>	<b>1</b>
<b>MSVD</b>												
MultiCLIP	-	54.91	80.82	87.18	56.24	16.96	9.23	0.67	71.69	66.79	11.46	1
MultiCLIP	Event	<b>57.29</b>	<b>83.26</b>	<b>89.18</b>	<b>58.68</b>	<b>17.48</b>	<b>9.45</b>	<b>0.70</b>	<b>74.10</b>	<b>69.29</b>	<b>9.56</b>	<b>1</b>
InternVideo2-1B	-	63.59	84.85	89.46	65.03	17.85	9.49	<b>0.74</b>	77.51	73.68	12.32	<b>1</b>
InternVideo2-1B	Event	<b>63.77</b>	<b>85.30</b>	<b>89.99</b>	<b>65.24</b>	<b>17.94</b>	<b>9.54</b>	<b>0.74</b>	<b>77.84</b>	<b>73.95</b>	<b>9.40</b>	<b>1</b>

Table 1: Performance comparison on MultiVENT, MSRVTT “1k-A” and MSVD dataset. Our method with **Q<sub>2</sub>E**, even without ASR, improves NDCG by 5–19 points on the MultiVENT dataset, 1–2 points on the MSRVTT dataset and 1–3 points on the MSVD dataset. Adding ASR further enhances performance, with an additional 3–7 points on MultiVENT and 2 points on MSRVTT, resulting in a total improvement of **8–26 points** on MultiVENT and **3–4 points** on MSRVTT, across two different encoders. Since the original MSVD videos are muted (Xu et al., 2016), we have excluded the ASR-based version. “Event” refers to the event decomposition (§2.1) and “ASR” refers to the audio decomposition (§2.3). Results from our whole method is highlighted.

tion Set, yielding a text-to-text similarity score instead of text-to-video. Specifically, we compute the maximum similarity between the query and all multimodal descriptions. Given  $C$  descriptions associated with the video:

$$S_{i,v}|Q = \max_{c=1:C} Sim_t(Q, Cap(v)_c), \quad (1)$$

where  $Cap(v)$  returns the multimodal description of the video  $v$  and  $Sim_t(x, y)$  computes the ColBERT<sup>3</sup> similarity score between  $x$  and  $y$ .

**Event Decomposition vs. Multimodal Descriptions** We use the refined prequel/current/sequel events from the query (§2.1) and the multimodal description set (§2.2) to compute a many-to-many similarity score. Specifically, we compute the maximum similarity score across all descriptions for

<sup>3</sup>We experimented with cosine-based SBERT (Reimers and Gurevych, 2019) and found that ColBERT outperforms SBERT in both accuracy and speed. Notably, the captions generated by LLMs may have some noise. ColBERT mitigates the noise’s impact by leveraging maximum aggregation at the token level, rather than compressing information into a single vector, which effectively reduces or ignores noisy elements in the multimodal descriptions. Additionally, prior studies (Warner et al., 2024) have shown that ColBERT’s approach minimizes information loss introduced by the averaging process used in SBERT, further enhancing its performance.

each event and subsequently take the maximum similarity score among all the events. Formally:

$$S_{i,v}|Q = \max_{l=1:L} \max_{c=1:C} Sim_t(E_i(Q)_l, Cap(v)_c), \quad (2)$$

where  $L$  is the number of events (prequel / current / sequel) in the query, and  $E_i()$  returns the respective event decomposition (prequel / current / sequel).

This global maximum based approach reduces the negative impact of a single noisy decomposition by leveraging the global maximum instead of averaging, which is more vulnerable to noisy data. This global maximum effectively decrease the effect of plausible hallucination that can be expected with the LLM generation. For instance, for the query “LA Fire” one of the decomposed prequels might be “Man is playing with a dog” (a hallucinated event). As our method generates multiple decomposition of the same event but also takes the global maximum, **Q<sub>2</sub>E** effectively reduce the effect of hallucinated decompositions.

**Score Fusion** A traditional approach of combining ranks from different experts is to train a fusion model. But as **Q<sub>2</sub>E** is based on zero-shot use of LLM agents, fusing all the scores presents a chal-

Model	R@1↑	R@5↑	R@10↑	P@1↑	P@5↑	P@10↑	MRR↑	NDCG↑	MAP↑	MnR↓	MdR↓
<b>Arabic</b>											
MultiCLIP	<b>10.91</b>	44.32	72.31	<b>92.16</b>	76.47	62.55	<b>0.94</b>	76.10	84.59	14.44	<b>6.0</b>
+ Event	10.82	45.86	74.46	90.20	78.43	63.92	<b>0.94</b>	78.09	86.08	14.91	<b>6.0</b>
+ ASR	10.79	<b>49.66</b>	<b>79.41</b>	90.20	<b>83.53</b>	<b>68.24</b>	<b>0.94</b>	<b>82.07</b>	<b>88.32</b>	<b>11.59</b>	<b>6.0</b>
<b>Chinese</b>											
MultiCLIP	9.12	45.59	74.22	84.62	83.85	68.65	0.91	77.29	86.52	12.60	6.0
+ Event	<b>10.66</b>	48.81	82.13	<b>98.08</b>	90.00	76.35	<b>0.99</b>	86.32	<b>94.64</b>	9.59	6.0
+ ASR	10.46	<b>50.55</b>	<b>83.02</b>	96.15	<b>93.08</b>	<b>76.92</b>	0.97	<b>86.61</b>	93.96	<b>8.73</b>	<b>5.5</b>
<b>English</b>											
MultiCLIP	<b>10.61</b>	50.82	85.79	<b>100.00</b>	96.15	81.92	<b>1.00</b>	89.98	97.47	<b>7.64</b>	<b>5.0</b>
+ Event	<b>10.61</b>	<b>51.66</b>	86.45	<b>100.00</b>	<b>97.69</b>	82.50	<b>1.00</b>	90.57	98.04	8.07	<b>5.0</b>
+ ASR	<b>10.61</b>	<b>51.66</b>	<b>87.80</b>	<b>100.00</b>	<b>97.69</b>	<b>83.85</b>	<b>1.00</b>	<b>91.43</b>	<b>98.12</b>	7.74	<b>5.0</b>
<b>Korean</b>											
MultiCLIP	9.30	39.98	65.33	86.54	75.00	62.12	0.89	70.38	82.86	20.78	7.0
+ Event	<b>9.68</b>	43.65	72.65	<b>90.38</b>	82.31	69.04	<b>0.93</b>	76.92	87.04	15.87	<b>6.0</b>
+ ASR	<b>9.68</b>	<b>45.93</b>	<b>76.90</b>	<b>90.38</b>	<b>86.15</b>	<b>73.27</b>	0.92	<b>80.58</b>	<b>89.02</b>	<b>13.85</b>	<b>6.0</b>
<b>Russian</b>											
MultiCLIP	10.57	49.57	79.09	92.31	87.69	71.15	0.96	82.84	91.20	10.65	6.0
+ Event	10.39	50.55	81.61	92.31	89.62	73.46	0.96	84.62	91.46	10.88	6.0
+ ASR	<b>11.21</b>	<b>53.49</b>	<b>85.40</b>	<b>98.08</b>	<b>94.23</b>	<b>76.54</b>	<b>0.99</b>	<b>88.70</b>	<b>95.02</b>	<b>9.13</b>	<b>5.0</b>

Table 2: Comprehensive comparison of different languages. Results are reported on the MultiVENT dataset.

lengen. Naively averaging all scores would assume equal importance for each score, which does not reflect real-world scenarios. E.g., for videos relating to earthquakes, the Sequel vs. Descriptions score carries significantly more weight than the Prequel vs. Descriptions, as there is no visual indication of an earthquake before it occurs. [Yin and Jiang \(2024\)](#) demonstrated that entropy-based methods outperform others in zero-shot rank fusion tasks. In this entropy ranking, disparate similarity scores for an individual video are normalized to a softmax distribution, where low entropy indicates high confidence, while high inverse entropy corresponds to high confidence. Building on this insight, we use inverse entropy rank fusion to combine the scores.

Specifically, we convert each of the five previously computed similarity scores  $S_i$  to a weighted distribution  $P_i = \text{softmax}(S_i)$ , and compute the entropy  $H(P_i)$  of this distribution. This  $P_i$  provides an initial likelihood weighting across the  $V$  videos, according to the  $i$ th scoring criteria. We rescale  $P_i$  by its inverse entropy, computing a final score  $\hat{S}$  by summing across all rescaled  $P_i$ :

$$\hat{S} = \sum_i^5 \frac{1}{H(P_i)} * P_i. \quad (3)$$

By applying Eq. (3), we effectively assign greater

weights to scores with lower entropy, emphasizing more confident predictions. This  $\hat{S}$  contains an aggregate score for each video;  $\hat{S}$  can be sorted to return the ranked list of videos for the query.

## 3 Results

### 3.1 Datasets and Implementation

We used the event-centric MultiVENT dataset ([Sanders et al., 2023](#)), the widely-used MSR-VTT ([Xu et al., 2016](#)) and MSVD ([Chen and Dolan, 2011b](#)) dataset. We used the standard ‘‘MSR-VTT-1k-A’’ test split of 1,000 videos for computational constraint, as used in prior works ([Jiang et al., 2022](#); [Guan et al., 2023a](#); [Jiang et al., 2023a](#)).

Most of the MultiVENT videos are event-specific human queries with complex visual content (i.e., flood, people running) and rapid scene changes (i.e., news reports where multiple scenes are shown), where the average video is approximately 5.5 times longer than the MSR-VTT dataset (Table 16). On the other hand, MSR-VTT and MSVD focus on general scenarios of everyday life, i.e., a man is playing with the dog. We have used different datasets to show the robustness of our method across different datasets of complexity. Full implementation details are in App. A.5

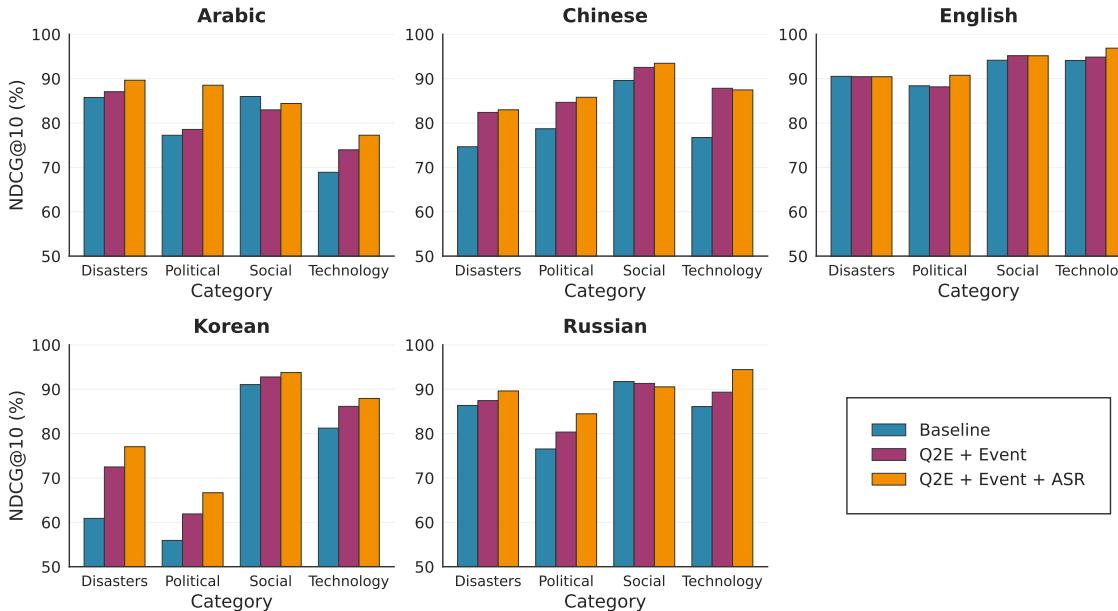


Figure 3: Comparison of NDCG@10 scores across topic categories (Disasters, Political, Social, Technology) for five languages—Arabic, Chinese, English, Korean, and Russian. The results demonstrate the performance improvement from the baseline to our method (with audio or without), highlighting consistent gains across languages and domains.

### 3.2 Zero-Shot Text-to-Video Retrieval

We use MultiCLIP (per Sanders et al. (2023)) and InternVideo2 (the best-performing method on the MSR-VTT leaderboard<sup>4</sup>) as our baseline encoders. The comparison of our method with the baseline for MultiVENT, MSR-VTT-1kA and MSVD is shown on the Table 1. We use the standard retrieval metrics: recall/precision at rank K, mean reciprocal rank (MRR), mean average precision (MAP), normalized discounted cumulative gain (NDCG), and mean and median rank (MnR, MdR).

**Overall, combined with event decomposition and ASR, Q<sub>2</sub>E gives the best score in all the metrics across three different datasets and two different text-to-video encoders. But even without ASR, we outperformed the baseline in all metrics.** To show the generalizability of our method across different text-video encoders, we applied it to both the MultiCLIP and InternVideo2 encoder.

Table 1 shows the generality of our method. When applied to the InternVideo2, our method improved the MSR-VTT NDCG score by 1 point without audio and 4 points with audio. Additionally, the significant performance drop observed on the MultiVENT dataset underscores the dataset’s richness and complexity. Despite this drop, incorporating our method boosted the score by **19 points** without audio and **26 points** with audio, further validating

its effectiveness.

Conversely, MultiCLIP achieved the best baseline score on the MultiVENT dataset (Sanders et al., 2023). Applying our method to MultiCLIP resulted in an **8-point** NDCG improvement, setting a new state-of-the-art for the MultiVENT dataset. However, unlike InternVideo2, MultiCLIP does not use a cross-encoder reranker, which explains its comparatively lower performance on the MSR-VTT dataset. Our method still led to a **4-point** improvement in the MSR-VTT NDCG score, demonstrating effectiveness across different encoders. Additionally, a full comparison with recent and popular baselines are provided in App. A.1.

Next, we examine Q<sub>2</sub>E’s performance across languages (§3.3), the effect of different LLM sizes (§3.4), different rank fusion methods (§3.5) and various components of our method (§3.6). In the appendix, we examine the impact of VLM sizes (App. A.3.1), different layers of ASR module (App. A.3.2), number of frames (App. A.3.3), and frame selection methods (App. A.3.4). We provide a qualitative analysis in App. A.2.

### 3.3 Performance Across Languages

MultiVENT includes five languages – Arabic, Chinese, English, Korean, and Russian. Table 2 shows that building upon the MultiCLIP baseline, incorporating Q<sub>2</sub>E consistently improves retrieval performance across all metrics. outperforming the

<sup>4</sup><https://paperswithcode.com/sota/zero-shot-video-retrieval-on-msr-vtt>

Size	R@10↑	P@10↑	MRR↑	NDCG↑
MultiCLIP	70.82	65.25	0.92	75.34
1B	78.71	72.28	<b>0.95</b>	82.50
3B	79.17	72.74	<b>0.95</b>	83.03
8B	79.04	72.59	<b>0.95</b>	82.91
70B (Q <sub>2</sub> E)	<b>79.60</b>	<b>73.09</b>	<b>0.95</b>	<b>83.24</b>

Table 3: Comparison of different LLM sizes within the Llama-3 family (1B to 70B parameters), specifically used for query decomposition and refining frame captions to video caption. Results are reported on the MultiVENT dataset. Full results in Table 12, App. A.3.5.

baseline by high numbers. We observed that audio-based decomposition plays a more significant role in languages like Arabic and Russian, whereas event-based decomposition is more crucial for Chinese and Korean. However, when combined, the two approaches complement each other, resulting in an aggregated performance boost.

MultiCLIP struggles the most with Arabic, Chinese, and Korean video retrieval. However, when our method is added, retrieval performance improves by approximately 6, 9, and 10 NDCG points for these languages, respectively.

To analyze the bias of the events and the effect of the bias on the performance of our method, we have used event category (provided in the main dataset) to divide the queries into 4 different buckets – Disaster, Political, Social, Technology. The results of different bucket across different language is provided in the Fig. 3. We can see that our method improves the NDCG score irrespective of language or event category (except Arabic-Social). Also, the performance improvement is boosted with addition of audio features. The Social category, in particular, shows the most variation, with Arabic and Russian even experiencing a slight dip in performance, which could be linked to cultural differences in how queries are interpreted. These varying results across languages support the idea that query decomposition brings out language-specific nuances.

#### 3.4 Effect of LLM size

Our original method used the Llama-3.3-70B LLM. While this model is highly robust and generalizable, its large size demands significant computational power and increased inference time. To demonstrate the robustness of our method, Table 3 presents results for different LLM sizes, ranging from 1B to 70B parameters. Although the best performance is typically achieved with the 70B model,

Method	R@10↑	P@10↑	MRR↑	NDCG↑
Neg. Exp. Ent.	67.23	61.97	0.93	73.20
RRF	70.91	65.29	0.93	76.29
Max	76.10	70.04	0.93	80.04
Mean	78.64	72.47	<b>0.95</b>	82.44
Inv. Ent. (Q <sub>2</sub> E)	<b>79.60</b>	<b>73.09</b>	<b>0.95</b>	<b>83.24</b>

Table 4: Comparison of Zero-Shot Rank Fusion methods: Neg. Exp. Ent. (Negative Exponential Entropy), RRF (Reciprocal Rank Fusion), Inv. Ent. (Inverse Entropy), Mean and Max. Results are reported on the MultiVENT dataset. Full results in Table 13, App. A.3.6.

even the 1B model consistently outperforms the baseline NDCG by at least 8 points. These results demonstrate that our method can achieve comparable performance even when using significantly smaller LLMs, highlighting its practicality for real-world retrieval systems.

#### 3.5 Effect of Rank Fusion Approaches

Fusing different ranks in a zero-shot setting is a non-trivial task. Traditional approaches often involve training dense networks to combine scores (Yu et al., 2024) or leveraging LLMs to ensemble multiple expert outputs (Lu et al., 2024b,a; Jiang et al., 2023b). However, rank or score fusion is a significant challenge for LLMs, as they must process and compare a large volume of numerical data, including floating-point values (for scores) and discrete values (for ranks). In both cases, LLMs struggle to produce reasonable fusion ranks. We considered several aggregation methods, from simple techniques like Mean to more complex approaches such as Reciprocal Rank Fusion (RRF) and Inverse Entropy; due to space, we describe these formulations in App. A.3.6. Results in Table 4 indicate that even the simple Mean function performs well compared to RRF and Negative Exponential Entropy. Inverse Entropy demonstrated better performance, leading us to adopt it as the aggregation method.

While we adopted global max aggregation, for a fine-grained analysis of the max and mean operator, we have also tried different combination of max-mean aggregation over captions and events. The results are shown in the Table 14. The results show that (1) Q<sub>2</sub>E performs the best compared with other aggregation methods. (2) Mean over captions underperformed the most, specifically learning nothing. We hypothesize that, as previously seen, the VLM models make noisy captions, and averaging over them hurts the model beyond what can be overcome. (3) Most of the other methods perform

Components	R@10↑	P@10↑	MRR↑	NDCG↑
<b>Q<sub>2</sub>E</b>	<b>79.60</b>	<b>73.09</b>	<b>0.95</b>	<b>83.24</b>
w/o Video	67.74	62.43	0.93	73.96
w/o Query	78.12	71.74	0.93	81.54
w/o Event	77.77	71.47	0.94	81.75

Table 5: Analysis of different components in our method. Event = (Prequel + Current + Sequel). Results are reported on the MultiVENT dataset. The full result is reported on Table 15 in App. A.3.7.

comparably with the global max, though we think global max wins because of its small performance gain and high simplicity. (4) The consistent performance across different evaluation metrics indicates that our method offers reliable improvements rather than metric-specific optimizations.

### 3.6 Effect of Different Components

Table 5 presents an ablation study of event decomposition and multimodal description components of **Q<sub>2</sub>E**. As expected, the results show that the video is the most critical. Removing the query results in a 1-point performance drop, while removing events reduces performance by 2 points, highlighting the importance of events for retrieving relevant videos.

## 4 Related Work

With the rapid advancements in LLMs (Devlin et al., 2019; Radford et al., 2019; Touvron et al., 2023; Dubey et al., 2024; Gunasekar et al., 2023; Liu et al., 2024a; Jiang et al., 2024; Chu et al., 2024), methodologies such as zero-shot, few-shot, and in-context learning have gained substantial traction due to their reduced computational requirements and enhanced time efficiency. These have influenced approaches for event causal reasoning, multimodal language models, and retrieval.

### 4.1 Event Causal Reasoning

Reasoning based on event causality plays a crucial role across various domains. Sun et al. (2024) demonstrated that event-causal graphs, even generated by LLMs, can enhance story understanding and better align with human evaluations. Similarly, in **Q<sub>2</sub>E**, we leverage causation from human queries to construct prequel, current, and sequel events, enabling a deeper understanding of user intent. While Chan et al. (2024) utilized LLMs to infer temporal, causal, and discourse relationships between two events, our approach applies the same temporal (prequel or sequel) and causal (current) reasoning

to extract relevant events directly from queries.

### 4.2 Vision Language Model

Vision-Language Models (VLMs) have been used significantly in text-to-video retrieval tasks by leveraging large-scale pre-trained models trained on image-text pairs (Ibrahimi et al., 2023). These models have demonstrated impressive zero-shot capabilities and transfer learning potential for video-related tasks (Zhao et al., 2024). Recent advancements have focused on efficiently adapting image-based VLMs to the video domain, addressing temporal dynamics and computational costs (Nishimura et al., 2024). Some approaches propose novel architectures like sparse-and-correlated adapters (Cao et al., 2024) to enhance retrieval performance while maintaining efficiency. Additionally, researchers have explored incorporating audio information (Ibrahimi et al., 2023) and using synthesized instructional data (Zhao et al., 2024) to improve video-language models further.

### 4.3 Zero-Shot Text-to-Video Retrieval

Though much has been done in the domain of fine-tuned text-to-video retrieval (Wang et al., 2025; Cicchetti et al., 2024; Chen et al., 2023; Xu et al., 2023), there is a gap in zero-shot text-to-video retrieval performance. Fine-tuning a model on a task-specific dataset can lead to impressive performance, but it is not always suitable due to high computing costs and time constraints. While zero-shot text-to-video retrieval has been previously studied (Wang et al., 2025; Cicchetti et al., 2024; Chen et al., 2023; Xu et al., 2023) results suggest a sizable gap between that and fine-tuned retrieval.

## 5 Conclusion & Future Work

We leveraged the world knowledge embedded in LLMs and VLMs to decompose both queries and videos for the video retrieval task, demonstrating how to match this knowledge against visual and audio signals. We employed inverse entropy as a rank fusion method without fine-tuning any additional fusion network. Experimental results across three datasets with different characteristics demonstrate the robustness and generalizability of our approach. While we show how the preexisting knowledge of LLMs and VLMs can be effectively transferred to text-to-video retrieval, our work sets the stage for future research to explore how factual and counterfactual information can be leveraged to positively or negatively align with the query.

## 6 Limitations

While our approach demonstrates superior performance across multiple datasets, it is computationally expensive and time-intensive. Future research should focus on optimizing efficiency to reduce both time and cost.

Due to the LLM and VLM’s inherent nature, some amount of misinformation or hallucination may be possible. Although we did not find any misinformation during our evaluation, our prompting of what *could* happen before, during, or after an event could result in hallucinations. Similarly, by relying on the parametric knowledge of these models, there is a potential for biases inherent in the models or their outputs to be propagated. We did not explicitly explore or mitigate the biases inherent in these models, though we also did not notice objectionable or stereotypical output.

The primary focus of our method is to demonstrate how query enrichment through decomposition can improve understanding of human intent and ground specific events to video content. As we have not applied any enhancements to the video captions or audio transcripts, video-to-text retrieval is outside the scope of this work.

## 7 Acknowledgment

This material is based in part upon work supported by the National Science Foundation under Grant No. IIS-2024878. Some experiments were conducted on the UMBC HPCF, supported by the National Science Foundation under Grant No. CNS-1920079. This material is also based on research that is in part supported by the Army Research Laboratory, Grant No. W911NF2120076, and by DARPA for the SciFy program under agreement number HR00112520301. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either express or implied, of ARL, DARPA or the U.S. Government.

## References

Max Bain, Arsha Nagrani, Gü̈l Varol, and Andrew Zisserman. 2022. *Frozen in Time: A Joint Video and Image Encoder for End-to-End Retrieval*. *arXiv preprint*. ArXiv:2104.00650.

Meng Cao, Haoran Tang, Jinfa Huang, Peng Jin, Can Zhang, Ruyang Liu, Long Chen, Xiaodan Liang, Li Yuan, and Ge Li. 2024. Rap: Efficient text-video retrieval with sparse-and-correlated adapter. *arXiv preprint arXiv:2405.19465*.

Chunkit Chan, Cheng Jiayang, Weiqi Wang, Yuxin Jiang, Tianqing Fang, Xin Liu, and Yangqiu Song. 2024. *Exploring the potential of ChatGPT on sentence level relations: A focus on temporal, causal, and discourse relations*. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 684–721, St. Julian’s, Malta. Association for Computational Linguistics.

David Chen and William Dolan. 2011a. *Collecting highly parallel data for paraphrase evaluation*. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 190–200, Portland, Oregon, USA. Association for Computational Linguistics.

David L Chen and Bill Dolan. 2011b. Building a persistent workforce on mechanical turk for multilingual data collection. In *Building a Persistent Workforce on Mechanical Turk for Multilingual Data Collection*.

Sihan Chen, Handong Li, Qunbo Wang, Zijia Zhao, Mingzhen Sun, Xinxin Zhu, and Jing Liu. 2023. *VAST: A Vision-Audio-Subtitle-Text Omni-Modality Foundation Model and Dataset*. *arXiv preprint*. ArXiv:2305.18500.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhipang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, et al. 2024. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*.

Giordano Cicchetti, Eleonora Grassucci, Luigi Sigillo, and Danilo Comminiello. 2024. Gramian multimodal representation learning and alignment. *arXiv preprint arXiv:2412.11959*.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Alexandros Delitzas, Maria Parelli, Nikolas Hars, Georgios Vlassis, Sotirios Anagnostidis, Gregor Bachmann, and Thomas Hofmann. 2023. Multi-clip: Contrastive vision-language pre-training for question answering tasks in 3d scenes. *arXiv preprint arXiv:2306.02329*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, volume 1, page 2.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman,

Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Han Fang, Pengfei Xiong, Luhui Xu, and Yu Chen. 2021. **CLIP2Video: Mastering Video-Text Retrieval via Image CLIP**. *arXiv preprint*. ArXiv:2106.11097.

Han Fang, Pengfei Xiong, Luhui Xu, and Wenhan Luo. 2023. **Transferring Image-CLIP to Video-Text Retrieval via Temporal Relations**. *IEEE Transactions on Multimedia*, 25:7772–7785.

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**. *arXiv preprint*. ArXiv:2305.05665 [cs].

Satya Krishna Gorti, Noel Vouitsis, Junwei Ma, Keyvan Golestan, Maksims Volkovs, Animesh Garg, and Guangwei Yu. 2022. **X-Pool: Cross-Modal Language-Video Attention for Text-Video Retrieval**. *arXiv preprint*. ArXiv:2203.15086 [cs].

Peiyan Guan, Renjing Pei, Bin Shao, Jianzhuang Liu, Weimian Li, Jiaxi Gu, Hang Xu, Songcen Xu, Youliang Yan, and Edmund Y Lam. 2023a. Pidro: Parallel isomeric attention with dynamic routing for text-video retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11164–11173.

Peiyan Guan, Renjing Pei, Bin Shao, Jianzhuang Liu, Weimian Li, Jiaxi Gu, Hang Xu, Songcen Xu, Youliang Yan, and Edmund Y. Lam. 2023b. **PIDRo: Parallel Isomeric Attention with Dynamic Routing for Text-Video Retrieval**. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11130–11139. ISSN: 2380-7504.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojgan Javaheripir, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. 2023. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*.

Sarah Ibrahimi, Xiaohang Sun, Pichao Wang, Amanmeet Garg, Ashutosh Sanan, and Mohamed Omar. 2023. Audio-enhanced text-to-video retrieval using text-conditioned feature alignment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12054–12064.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Chen Jiang, Hong Liu, Xuzheng Yu, Qing Wang, Yuan Cheng, Jia Xu, Zhongyi Liu, Qingpei Guo, Wei Chu, Ming Yang, et al. 2023a. Dual-modal attention-enhanced text-video retrieval with triplet partial margin contrastive learning. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 4626–4636.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. 2023b. Llm-blender: Ensembling large language models with pairwise ranking and generative fusion. *arXiv preprint arXiv:2306.02561*.

Jie Jiang, Shaobo Min, Weijie Kong, Hongfa Wang, Zhifeng Li, and Wei Liu. 2022. Tencent text-video retrieval: Hierarchical cross-modal interactions with multi-level representations. *IEEE Access*.

Yili Li, Jing Yu, Keke Gai, Bang Liu, Gang Xiong, and Qi Wu. 2024. **T2VIndexer: A Generative Video Indexer for Efficient Text-Video Retrieval**. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 3955–3963. ArXiv:2408.11432 [cs].

Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, et al. 2024a. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*.

Haowei Liu, Yaya Shi, Haiyang Xu, Chunfeng Yuan, Qinghao Ye, Chenliang Li, Ming Yan, Ji Zhang, Fei Huang, Bing Li, et al. 2024b. Unifying latent and lexicon representations for effective video-text retrieval. *arXiv preprint arXiv:2402.16769*.

Yuqi Liu, Pengfei Xiong, Luhui Xu, Shengming Cao, and Qin Jin. 2022. **TS2-Net: Token Shift and Selection Transformer for Text-Video Retrieval**. *arXiv preprint*. ArXiv:2207.07852 [cs].

Keming Lu, Hongyi Yuan, Runji Lin, Junyang Lin, Zheng Yuan, Chang Zhou, and Jingren Zhou. 2024a. **Routing to the expert: Efficient reward-guided ensemble of large language models**. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1964–1974, Mexico City, Mexico. Association for Computational Linguistics.

Xudong Lu, Qi Liu, Yuhui Xu, Aojun Zhou, Siyuan Huang, Bo Zhang, Junchi Yan, and Hongsheng Li. 2024b. **Not all experts are equal: Efficient expert pruning and skipping for mixture-of-experts large language models**. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6159–6172, Bangkok, Thailand. Association for Computational Linguistics.

Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. 2021. **CLIP4Clip: An Empirical Study of CLIP for End to End Video Clip Retrieval**. *arXiv preprint*. ArXiv:2104.08860 [cs] version: 2.

Taichi Nishimura, Shota Nakada, and Masayoshi Kondo. 2024. **Vision-language models learn super images for efficient partially relevant video retrieval**. *ACM Trans. Multimedia Comput. Commun. Appl.* Just Accepted.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Kate Sanders, David Etter, Reno Kriz, and Benjamin Van Durme. 2023. Multivent: Multilingual videos of events and aligned natural text. *Advances in Neural Information Processing Systems*, 36:51065–51079.

Yidan Sun, Qin Chao, and Boyang Li. 2024. Event causality is key to computational story understanding. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3493–3511, Mexico City, Mexico. Association for Computational Linguistics.

Mingkai Tian, Guorong Li, Yuankai Qi, Shuhui Wang, Quan Z. Sheng, and Qingming Huang. 2024. Rethink video retrieval representation for video captioning. *Pattern Recognition*, 156:110744.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajwala Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Qiang Wang, Yanhao Zhang, Yun Zheng, Pan Pan, and Xian-Sheng Hua. 2022. Disentangled Representation Learning for Text-Video Retrieval. *arXiv preprint arXiv:2203.07111 [cs]*.

Yi Wang, Kunchang Li, Xinhao Li, Jiashuo Yu, Yinan He, Guo Chen, Baoqi Pei, Rongkun Zheng, Zun Wang, Yansong Shi, et al. 2025. Internvideo2: Scaling foundation models for multimodal video understanding. In *European Conference on Computer Vision*, pages 396–416. Springer.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, et al. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *arXiv preprint arXiv:2412.13663*.

Wenhai Wu, Haipeng Luo, Bo Fang, Jingdong Wang, and Wanli Ouyang. 2023. Cap4Video: What Can Auxiliary Captions Do for Text-Video Retrieval? *ArXiv:2301.00184 [cs]*.

Haiyang Xu, Qinghao Ye, Ming Yan, Yaya Shi, Jiabo Ye, Yuanhong Xu, Chenliang Li, Bin Bi, Qi Qian, Wei Wang, et al. 2023. mplug-2: A modularized multi-modal foundation model across text, image and video. In *International Conference on Machine Learning*, pages 38728–38748. PMLR.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msrvtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296.

Hongwei Xue, Yuchong Sun, Bei Liu, Jianlong Fu, Ruihua Song, Houqiang Li, and Jiebo Luo. 2023. CLIP-VIP: Adapting Pre-trained Image-Text Model to Video-Language Representation Alignment. *arXiv preprint arXiv:2209.06430 [cs]*.

Siqi Yin and Lifan Jiang. 2024. Multi-method integration with confidence-based weighting for zero-shot image classification. *arXiv preprint arXiv:2405.02155*.

Shi Yu, Chenghao Fan, Chenyan Xiong, David Jin, Zhiyuan Liu, and Zhenghao Liu. 2024. Fusion-in-t5: Unifying variant signals for simple and effective document ranking with attention fusion. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 7556–7561, Torino, Italia. ELRA and ICCL.

Yue Zhao, Long Zhao, Xingyi Zhou, Jialin Wu, Chun-Te Chu, Hui Miao, Florian Schroff, Hartwig Adam, Ting Liu, Boqing Gong, et al. 2024. Distilling vision-language models on millions of videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13106–13116.

## A Appendix

### A.1 Comparison with Popular Baselines

To show the robustness of our zero-shot method, in the Table 6, we have reported the comparison of our method with different recent models: Frozen in Time (Bain et al., 2022), ImageBind (Girdhar et al., 2023), CLIP (Radford et al., 2021), CLIP4CLIP (Luo et al., 2021), CLIP2Video (Fang et al., 2021), X-Pool (Gorti et al., 2022), TS2-Net (Liu et al.,

2022), TDB+TAB (Fang et al., 2023), IVRC (Tian et al., 2024), Cap4Video (Wu et al., 2023), DRL (Wang et al., 2022), T2VIndexer (Li et al., 2024), PIDRo (Guan et al., 2023b), CLIP-VIP (Xue et al., 2023), huge (Jiang et al., 2022).

Models	Venue	FT	R@1↑	R@5↑	R@10↑
<b>MSR-VTT</b>					
Frozen in Time	ICCV '21	✗	27.30	-	68.10
ImageBind	CVPR '23	✗	36.80	61.80	70.00
CLIP	PMLR '21	✗	39.70	72.30	82.20
CLIP4CLIP	arXiv '21	✓	44.50	71.40	81.60
CLIP2Video	arXiv '21	✓	45.60	72.60	81.70
X-Pool	CVPR '22	✓	46.90	72.80	82.20
TS2-Net	ECCV '22	✓	45.60	74.40	82.70
TDB+TAB	TMM '23	✓	45.60	72.60	81.70
IVRC	Patt. Recog. '24	✓	47.00	75.00	82.40
Cap4Video	CVPR '23	✓	51.40	75.70	83.90
DRL	arXiv '22	✓	53.30	80.30	87.60
T2VIndexer	ACM MM '24	✓	55.10	77.20	85.00
PIDRo	ICCV '23	✓	55.90	79.80	87.60
<b>Q<sub>2</sub>E</b>	-	✗	56.28	76.58	83.72
CLIP+ViP	ICLR '23	✓	57.70	80.50	88.20
huge	arXiv '22	✓	<b>62.90</b>	<b>84.50</b>	<b>90.80</b>
<b>MSVD</b>					
Frozen in Time	ICCV '21	✗	33.70	64.70	76.30
CLIP	PMLR '21	✗	37.00	64.10	73.80
CLIP4CLIP	arXiv '21	✓	46.20	76.10	84.60
TDB+TAB	TMM '23	✓	46.90	76.90	86.10
CLIP2Video	arXiv '21	✓	47.00	76.80	85.90
X-Pool	CVPR '22	✓	47.20	77.40	86.00
PIDRo	ICCV '23	✓	47.50	77.50	86.00
DRL	arXiv '22	✓	48.30	79.10	87.30
Cap4Video	CVPR '23	✓	51.80	80.80	88.30
T2VIndexer	ACM MM '24	✓	55.40	81.90	88.50
huge	arXiv '22	✓	59.00	84.00	<b>90.30</b>
<b>Q<sub>2</sub>E</b>	-	✗	<b>63.77</b>	<b>85.30</b>	89.99

Table 6: Results from different baseline models reported in the recent papers. Rows are sorted based on R@1 score. FT represents the fine-tuned models for text-to-video retrieval task.

On the **MSR-VTT** dataset, our method outperforms all of the zero-shot methods by high margin. However, while fine-tuning can achieve moderately higher performance, our zero-shot method performs competitively. On the other hand, in the **MSVD** dataset, our method outperforms all baseline models, whether fine-tuned or not. This again underlines the generalizability and robustness of our method across different datasets compared to the previous baselines.

## A.2 Qualitative Analysis

We have reported two examples from MSRVTT (Xu et al., 2016) and MultiVENT (Sanders et al., 2023) dataset on the Table 7. We observed that while event decomposition may not be as grounded or clearly delineated as in the Multi-

VENT dataset (due to the inherent ungrounded nature of MSRVTT queries), our approach still enriches the query by introducing events that could occur before, during or after the main event. This enrichment significantly enhances the retrieval process.

From the retrieval ranking perspective, incorporating event-based decomposition substantially improves video retrieval accuracy. For example, for the provided MSR-VTT query, the ground-truth video is ranked 12th (with audio) and 13th (without audio) by MultiCLIP, whereas **Q<sub>2</sub>E** (which integrates decomposition module, multimodal features, textual descriptions, and our ASR module) elevates it to 1st place with audio and 7th place without audio.

## A.3 Ablation Study (continued)

### A.3.1 Effect of VLM Sizes

In **Q<sub>2</sub>E**, we utilized the InternVL-2.5-78B model. Similar to the Llama-3.3-70B, this model is highly robust and generalizable but comes with significant computational and time constraints. To assess the efficiency and flexibility of our method, Table 8 reports results across various VLM sizes within the InternVL-2.5 family.

The results indicate that models ranging from 4B to 38B parameters achieve nearly identical performance, with negligible differences in scores. This demonstrates that our method can achieve comparable results even with smaller VLMs, reducing computational requirements. Furthermore, with the inclusion of audio, the performance gap narrows even further; for instance, the 1B VLM experiences only a 1-point drop in NDCG score. These findings underscore the robustness and generalizability of our method, regardless of the VLM size.

### A.3.2 Effect of different layers in Translation Module

As described in the §2.3, in our method we have used a multi-layer ASR system consisting of ASR, an automatic translator and an LLM refiner. To assess the impact of different layers of translation, we have provided ablation studies in Table 9. In summary, while excluding one translation layer results in a small reduction in NDCG score (0.1-0.5 points), excluding all translation layers leads to a much more significant decline (2-7 points). This suggests strong complementary aspects and a strong aggregate effect.

Dataset	MSR-VTT
Query	Crocodiles about to eat the group of people walking across the water
Prequel	<ul style="list-style-type: none"> <li>• People wading through the water, unaware of the crocodiles before they attack</li> <li>• Crocodiles swimming stealthily towards the group during an attack on people</li> <li>• The group noticing the crocodiles and trying to hurry across the water</li> <li>• Crocodiles rising out of the water, jaws open</li> <li>• People walking cautiously across the water to avoid crocodiles attacking them</li> </ul>
Current	<ul style="list-style-type: none"> <li>• Crocodiles attacking people while swimming towards a group of them</li> <li>• Crocodiles emerging from the water with their mouths open during attacks on people</li> <li>• People screaming and trying to run or swim away from crocodiles during attacks on humans</li> <li>• People trying to hurry or run across the water when crocodiles attack them</li> <li>• Crocodiles snapping their jaws near people during attacks</li> </ul>
Sequel	<ul style="list-style-type: none"> <li>• Crocodiles attacking people while chasing after a group</li> <li>• People defending themselves against crocodile attacks when crocodiles attack them</li> <li>• Crocodiles dragging the people underwater</li> </ul>
Dataset	MultiVENT
Query	November 30 earthquake in South Central Alaska 2018
Prequel	<ul style="list-style-type: none"> <li>• Buildings shaking and swaying due to the earthquake in Anchorage, Alaska, on November 30, 2018</li> <li>• People running out of buildings and evacuating the area in Anchorage, Alaska during the earthquake on November 30, 2018</li> <li>• Cars stopped on the road as the earthquake strikes in Anchorage, Alaska on November 30, 2018</li> <li>• Debris and objects falling from shelves and ceilings during the 30 November 2018 Anchorage, Alaska earthquake</li> <li>• Emergency responders rushing to the scene to assist with evacuation and relief efforts after the 30 November 2018 earthquake in Anchorage, Alaska, South Central Alaska</li> </ul>
Current	<ul style="list-style-type: none"> <li>• Buildings shaking and crumbling during the 30 November 2018 earthquake in Anchorage, Alaska, South Central Alaska</li> <li>• People running out of buildings and evacuating the area during the 30 November 2018 Anchorage, Alaska earthquake</li> <li>• Emergency responders rushing to the scene after the 2018 Anchorage, Alaska earthquake on November 30, 2018</li> <li>• Debris falling from buildings and damaging streets during the November 30, 2018 earthquake in Anchorage, South Central Alaska</li> <li>• Cars stopped or abandoned on the road in Anchorage, Alaska, South Central Alaska due to the earthquake on November 30, 2018</li> </ul>
Sequel	<ul style="list-style-type: none"> <li>• Buildings crumbling or collapsing during the November 30, 2018 earthquake in Anchorage, Alaska</li> <li>• People running for cover or evacuating buildings during the November 30, 2018 earthquake in Anchorage, Alaska</li> <li>• Emergency vehicles rushing to the scene of the 2018 Anchorage, Alaska earthquake on November 30, 2018</li> <li>• Cracks forming in roads and highways in Anchorage, Alaska, South Central Alaska, after the 30 November 2018 earthquake</li> <li>• Debris falling from damaged structures during the 30 November 2018 earthquake in Anchorage, Alaska</li> </ul>

Table 7: Qualitative analysis on two examples from MSRVTT (Xu et al., 2016) and MultiVENT(Sanders et al., 2023) dataset respectively.

### A.3.3 Effect of Number of Frames

Fig. 4 illustrates the impact of the number of frames on performance across different methods. The results show that increasing the number of sampled frames generally improves performance. However, as the number of comparisons per second

increases (i.e., fewer frames are sampled), NDCG declines across all methods, highlighting a trade-off between computational efficiency and retrieval performance.

While the upward trend continues beyond 64 frames, we hypothesize that further increasing the number of frames would yield additional perfor-

VLM Size	R@1↑	R@5↑	R@10↑	P@1↑	P@5↑	P@10↑	MRR↑	NDCG↑	MAP↑	MnR↓	MdR↓
MultiCLIP	9.83	44.32	70.82	88.80	80.77	65.25	0.92	75.34	86.33	22.12	<b>6</b>
<b>Without Audio</b>											
1B	9.80	43.76	71.73	89.19	79.85	66.02	0.93	75.86	86.05	24.96	<b>6</b>
2B	9.94	45.23	73.86	89.96	82.55	68.03	0.93	77.90	87.49	22.45	<b>6</b>
4B	9.97	46.75	75.85	90.73	85.17	69.92	0.93	79.81	89.05	<b>19.68</b>	<b>6</b>
8B	10.03	46.40	75.73	90.35	84.71	69.81	0.93	79.72	88.98	21.24	<b>6</b>
26B	<b>10.24</b>	<b>47.03</b>	<b>76.16</b>	<b>92.66</b>	<b>85.64</b>	<b>70.19</b>	<b>0.95</b>	<b>80.47</b>	<b>89.94</b>	19.87	<b>6</b>
38B (Q <sub>2</sub> E)	<b>10.24</b>	46.71	75.76	92.28	85.25	69.73	<b>0.95</b>	80.04	89.42	21.13	<b>6</b>
<b>With Audio</b>											
1B	10.18	47.87	78.15	91.89	86.56	71.74	0.94	81.77	90.33	18.79	<b>6</b>
2B	10.14	48.27	78.46	91.51	87.26	72.05	0.94	82.14	90.58	17.21	<b>6</b>
4B	10.19	48.90	79.72	92.28	88.57	73.17	0.94	83.13	91.05	16.77	<b>6</b>
8B	<b>10.40</b>	48.87	79.88	<b>93.82</b>	88.34	73.36	<b>0.95</b>	83.52	<b>91.53</b>	17.27	<b>6</b>
26B	<b>10.40</b>	48.64	<b>80.12</b>	<b>93.82</b>	88.03	<b>73.51</b>	<b>0.95</b>	<b>83.61</b>	91.13	<b>15.97</b>	<b>6</b>
38B (Q <sub>2</sub> E)	10.32	<b>49.00</b>	79.60	93.05	<b>88.73</b>	73.09	<b>0.95</b>	83.24	91.20	16.44	<b>6</b>

Table 8: Comprehensive comparison of different VLM sizes within the InternVL-2.5 family (ranging from 1B to 38B parameters), specifically used for frame captioning. Results are reported on the MultiVENT dataset.

Components	R@10↑	P@10↑	MRR↑	NDCG↑
<b>Q<sub>2</sub>E</b>	<b>79.60</b>	<b>73.09</b>	<b>0.95</b>	<b>83.24</b>
w/o ASR	79.30	72.86	<b>0.95</b>	83.02
w/o Translator	79.47	73.01	<b>0.95</b>	83.14
w/o Refiner	79.25	72.82	<b>0.95</b>	82.95
w/o (ASR + Translator + Refiner)	75.76	69.73	<b>0.95</b>	80.04

Table 9: Analysis of different layers of the translation module. Results are reported on the MultiVENT dataset.

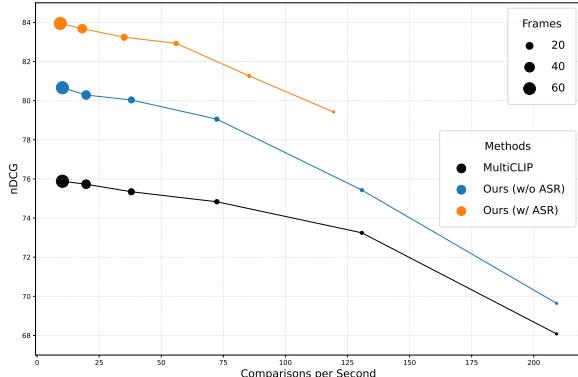


Figure 4: Impact of the number of frames on retrieval performance. The figure shows the nDCG performance against comparisons per second for three methods: MultiCLIP (black), Q<sub>2</sub>E (without ASR) (blue), and the Q<sub>2</sub>E (with ASR) (orange). The point size is proportional to the number of sampled frames (i.e., 2, 4, 8, 16, 32, 64). Comparison per Second = Runtime / (num. of Query × num. of Video). Results are reported on the MultiVENT dataset. The full table is reported on Table 10.

mance gains. However, the diminishing returns and increased computational cost would reduce the model’s practicality as a retrieval system. There-

fore, we limited our evaluation to 64 frames and selected 16-frame sampling as the optimal balance between performance and speed.

#### A.3.4 Effect of Frame Selection Methods

In Q<sub>2</sub>E, we employed a uniform sampling method for frame selection. However, to compare its effectiveness against a more complex approach, we conducted an ablation study using an adaptive detector from PySceneDetector<sup>6</sup>. As shown in Table 11, uniform sampling achieves the best performance across both variants, demonstrating its robustness regardless of audio availability. Furthermore, the scene-detection algorithm yields a variable number of frames (ranging from 2 to 100), significantly increasing retrieval time by approximately 2.8 times compared to 16-uniform sampling.

#### A.3.5 Effect of LLM Sizes

The detailed results for the experiment described in §3.4 are reported on Table 12.

#### A.3.6 Effect of different Rank Fusion

The detailed results for the experiment described in §3.5 are reported on Tables 13 and 14. These experiments consider five different approaches for aggregating the scores and performing zero-shot rank fusion. In all approaches, we compute a distribution  $P_i$  across all videos, for each of the  $i$  score components, as discussed in §3.5. The specific rank fusion approaches we consider then vary in how we

<sup>6</sup><https://www.scenedetect.com/>

# frames	R@1↑	R@5↑	R@10↑	P@1↑	P@5↑	P@10↑	MRR↑	NDCG↑	MAP↑	MnR↓	MdR↓
MultiCLIP	9.83	44.32	70.82	88.80	80.77	65.25	0.92	75.34	86.33	22.12	<b>6</b>
<b>Without Audio</b>											
2	9.31	41.71	64.46	83.78	76.37	59.54	0.90	69.64	83.75	36.17	7
4	9.75	44.00	71.05	88.03	80.39	65.64	0.92	75.43	86.54	25.32	<b>6</b>
8	10.10	46.56	74.84	90.73	84.79	68.88	0.94	79.05	89.05	23.01	<b>6</b>
16	<b>10.24</b>	46.71	75.76	<b>92.28</b>	85.25	69.73	<b>0.95</b>	80.04	89.42	21.13	<b>6</b>
32 (Q <sub>2</sub> E)	10.15	<b>47.22</b>	75.95	91.89	<b>85.95</b>	70.00	0.94	80.29	<b>90.15</b>	20.05	<b>6</b>
64	10.09	47.15	<b>76.77</b>	91.12	<b>85.95</b>	<b>70.66</b>	0.94	<b>80.66</b>	89.74	<b>19.98</b>	<b>6</b>
<b>With Audio</b>											
2	10.26	46.63	74.93	92.28	84.48	68.88	0.94	79.42	89.33	21.66	<b>6</b>
4	10.16	47.37	77.54	91.51	85.79	71.27	0.94	81.26	90.05	19.20	<b>6</b>
8	10.26	48.74	79.39	92.66	88.11	72.86	<b>0.95</b>	82.93	90.99	16.97	<b>6</b>
16	10.32	<b>49.00</b>	79.60	93.05	<b>88.73</b>	73.09	<b>0.95</b>	83.24	91.20	16.44	<b>6</b>
32 (Q <sub>2</sub> E)	<b>10.41</b>	48.83	80.09	<b>93.82</b>	88.26	73.55	<b>0.95</b>	83.68	91.32	16.38	<b>6</b>
64	10.40	48.90	<b>80.42</b>	<b>93.82</b>	88.49	<b>73.82</b>	<b>0.95</b>	<b>83.95</b>	<b>91.58</b>	<b>15.90</b>	<b>6</b>

Table 10: Full comparison of different number of frames (ranging from 2–64). Results are reported on the MultiVENT dataset.

Frame Sampling	R@1↑	R@5↑	R@10↑	P@1↑	P@5↑	P@10↑	MRR↑	NDCG↑	MAP↑	MnR↓	MdR↓
MultiCLIP	9.83	44.32	70.82	88.80	80.77	65.25	0.92	75.34	86.33	22.12	<b>6</b>
<b>Without Audio</b>											
scene_detect <sup>5</sup>	10.06	46.02	74.68	91.12	83.78	68.73	0.94	78.90	88.59	21.84	<b>6</b>
uniform (Q <sub>2</sub> E)	<b>10.24</b>	<b>46.71</b>	<b>75.76</b>	<b>92.28</b>	<b>85.25</b>	<b>69.73</b>	<b>0.95</b>	<b>80.04</b>	<b>89.42</b>	<b>21.13</b>	<b>6</b>
<b>With Audio</b>											
scene_detect	<b>10.36</b>	48.55	79.34	<b>93.44</b>	87.95	72.78	<b>0.95</b>	82.99	90.98	17.58	<b>6</b>
uniform (Q <sub>2</sub> E)	10.32	<b>49.00</b>	<b>79.60</b>	93.05	<b>88.73</b>	<b>73.09</b>	<b>0.95</b>	<b>83.24</b>	<b>91.20</b>	<b>16.44</b>	<b>6</b>

Table 11: Comprehensive comparison of Different Frame Sampling Methods. In the scene-detect approach, we utilized the AdaptiveDetector with its default settings. For the uniform sampling method, we selected 16 evenly spaced frames, whereas the number of frames in the scene-detect approach varied based on the algorithm’s output. Results are reported on the MultiVENT dataset.

LLM Size	R@1↑	R@5↑	R@10↑	P@1↑	P@5↑	P@10↑	MRR↑	NDCG↑	MAP↑	MnR↓	MdR↓
MultiCLIP	9.83	44.32	70.82	88.80	80.77	65.25	0.92	75.34	86.33	22.12	<b>6</b>
<b>Without Audio</b>											
1B	10.05	46.07	75.45	90.35	83.94	69.54	0.93	79.34	88.15	22.34	<b>6</b>
3B	10.18	46.70	75.50	91.51	<b>85.25</b>	69.54	0.94	79.78	89.40	21.66	<b>6</b>
8B	10.21	46.60	75.02	<b>92.28</b>	84.94	69.11	0.94	79.41	89.37	<b>19.74</b>	<b>6</b>
70B (Q <sub>2</sub> E)	<b>10.24</b>	<b>46.71</b>	<b>75.76</b>	<b>92.28</b>	<b>85.25</b>	<b>69.73</b>	<b>0.95</b>	<b>80.04</b>	<b>89.42</b>	21.13	<b>6</b>
<b>With Audio</b>											
1B	10.26	48.31	78.71	92.66	87.49	72.28	<b>0.95</b>	82.50	91.23	17.64	<b>6</b>
3B	10.32	48.88	79.17	93.05	88.57	72.74	<b>0.95</b>	83.03	<b>91.55</b>	16.51	<b>6</b>
8B	<b>10.41</b>	48.62	79.04	<b>93.82</b>	88.03	72.59	<b>0.95</b>	82.91	91.28	16.79	<b>6</b>
70B (Q <sub>2</sub> E)	10.32	<b>49.00</b>	<b>79.60</b>	93.05	<b>88.73</b>	<b>73.09</b>	<b>0.95</b>	<b>83.24</b>	91.20	<b>16.44</b>	<b>6</b>

Table 12: Comprehensive comparison of different LLM sizes within the Llama-3 family (ranging from 1B to 70B parameters), specifically used for query decomposition and refining frame captions to video caption. Results are reported on the MultiVENT dataset.

Method	R@1↑	R@5↑	R@10↑	P@1↑	P@5↑	P@10↑	MRR↑	NDCG↑	MAP↑	MnR↓	MdR↓
MultiCLIP	9.83	44.32	70.82	88.80	80.77	65.25	0.92	75.34	86.33	22.12	<b>6</b>
<b>Without Audio</b>											
Neg. Exp. Ent.	9.66	39.97	60.05	87.26	73.36	55.60	0.90	66.61	83.60	68.44	7
RRF	9.92	42.10	62.91	89.58	77.37	58.30	0.92	69.74	86.94	33.80	7
Max	9.98	45.15	71.98	90.35	82.32	66.41	0.93	76.60	87.77	29.31	<b>6</b>
Mean	10.19	45.99	73.54	<b>92.28</b>	84.09	67.88	0.94	78.37	<b>89.50</b>	26.32	<b>6</b>
Inv. Ent. ( <b>Q<sub>2</sub>E</b> )	<b>10.24</b>	<b>46.71</b>	<b>75.76</b>	<b>92.28</b>	<b>85.25</b>	<b>69.73</b>	<b>0.95</b>	<b>80.04</b>	89.42	<b>21.13</b>	<b>6</b>
<b>With Audio</b>											
Neg. Exp. Ent.	10.11	43.98	67.23	90.73	80.00	61.97	0.93	73.20	87.91	59.09	7
RRF	10.01	45.75	70.78	90.73	83.17	65.17	0.93	76.14	88.35	26.77	6
Max	9.94	46.58	76.20	89.96	84.79	70.12	0.93	79.86	88.82	22.89	6
Mean	10.23	48.06	78.30	92.66	87.41	72.16	0.95	82.20	91.04	20.32	6
Inv. Ent. ( <b>Q<sub>2</sub>E</b> )	<b>10.37</b>	<b>48.42</b>	<b>79.06</b>	<b>93.44</b>	<b>87.80</b>	<b>72.66</b>	<b>0.95</b>	<b>82.89</b>	<b>91.10</b>	<b>17.14</b>	<b>6</b>

Table 13: Full comparison of different aggregation methods for zero-shot rank fusion: Neg. Exp. Ent. (Negative Exponential Entropy), RRF (Reciprocal Rank Fusion), Inv. Ent. (Inverse Entropy), Mean and Max. Results are reported on the MultiVENT dataset.

Aggregation Method	R@10↑	P@10↑	MRR↑	NDCG↑
Mean over events & Max over captions	77.91	71.58	<b>0.95</b>	81.97
Max over events & Mean over captions	0.46	0.39	0.00	0.44
Max over events & Mean of Top 3 captions	78.90	72.47	0.94	82.54
Max over event & Mean over Top 5 captions	78.91	72.47	0.94	82.50
Mean over Top 3 events & Max over captions	79.11	72.70	<b>0.95</b>	82.91
Mean over Top 3 events & Mean over Top 3 captions	78.96	72.51	0.94	82.62
<b>Q<sub>2</sub>E</b>	<b>79.25</b>	<b>72.82</b>	<b>0.95</b>	<b>82.95</b>

Table 14: Performance comparison of different aggregation methods across retrieval metrics. **Q<sub>2</sub>E**, with global maximum, achieves the best performance across all metrics.

combine each of the five  $P_i$  in order to get a single ranking  $\hat{S}$  across all videos. Those approaches are:

**Mean Aggregation** We simply average out  $P_i$ , computing  $\hat{S} = \frac{1}{5} \sum_i P_i$ .

**Max Aggregation** For each video  $v$ , we maximize across the  $P_i$  score for that video:  $\hat{S}_v = \max_i P_{i,v}$ .

**Reciprocal Rank Fusion (RRF)** For each video  $v$ , we compute the rank of that video in  $P_i$ , and then sum across those reciprocal ranks. The final rank of  $v$  is then  $\sum_i (1/\text{Rank}(P_{i,v}))$ .

**Negative Exponential Entropy** We compute  $\hat{S} = \sum_i \exp(-H(P_i)) * P_i$ , where  $H(P_i)$  is the entropy of  $P_i$ .

**Inverse Entropy** This is the approach described in the main paper, in §3.5.

### A.3.7 Effect of Different Components

The detailed results for the experiment described in §3.6 are reported on Table 15. The fine-grained results for each component of events show a slight reduction in NDCG score (0.1-0.3) if we exclude one of the prequel, sequel, or current generations, if we exclude all, we see a much significant reduction (1-2 points), suggesting strong complementary aspects and a strong aggregate effect.

## A.4 Dataset Statistics

The detailed dataset statistics are presented on Table 16.

## A.5 Implementation Details

We used Llama-3<sup>7</sup> as the LLM, InternVL-2.5<sup>8</sup> as the VLM, and Whisper-large-v3<sup>9</sup> as the multilin-

<sup>7</sup>[meta-llama/Llama-3.3-70B-Instruct](https://meta-llama/Llama-3.3-70B-Instruct)

<sup>8</sup>[OpenGVLab/InternVL2.5-38B](https://opengvlab/InternVL2.5-38B)

<sup>9</sup>[openai/whisper-large-v3](https://openai/whisper-large-v3)

Components	R@1↑	R@5↑	R@10↑	P@1↑	P@5↑	P@10↑	MRR↑	NDCG↑	MAP↑	MnR↓	MdR↓
MultiCLIP	9.83	44.32	70.82	88.80	80.77	65.25	0.92	75.34	86.33	22.12	<b>6</b>
<b>Without Audio</b>											
<b>Q<sub>2</sub>E</b>	<b>10.24</b>	<b>46.71</b>	<b>75.76</b>	<b>92.28</b>	<b>85.25</b>	<b>69.73</b>	<b>0.95</b>	<b>80.04</b>	<b>89.42</b>	21.13	<b>6</b>
w/o Video	9.28	39.56	58.00	84.17	72.97	53.90	0.89	64.83	82.66	65.97	8
w/o Query	10.08	45.97	74.74	91.12	83.78	68.80	0.93	78.78	87.92	21.00	<b>6</b>
w/o Event	9.95	46.38	74.77	89.96	84.48	68.88	0.93	79.02	88.98	<b>18.35</b>	<b>6</b>
w/o Prequel	10.11	46.73	75.73	91.12	85.17	69.73	0.94	79.85	89.13	20.13	<b>6</b>
w/o Current	10.20	46.79	75.85	91.89	85.33	69.81	0.94	80.02	89.29	20.29	<b>6</b>
w/o Sequel	10.07	46.66	75.75	91.12	85.02	69.73	0.94	79.85	89.19	20.15	<b>6</b>
<b>With Audio</b>											
<b>Q<sub>2</sub>E</b>	<b>10.32</b>	<b>49.00</b>	<b>79.60</b>	<b>93.05</b>	<b>88.73</b>	<b>73.09</b>	<b>0.95</b>	<b>83.24</b>	<b>91.20</b>	16.44	<b>6</b>
w/o Video	10.16	44.61	67.74	91.51	81.16	62.43	0.93	73.96	88.97	50.96	<b>6</b>
w/o Query	10.08	47.39	78.12	91.12	85.95	71.74	0.93	81.54	89.18	<b>16.32</b>	<b>6</b>
w/o Event	10.22	48.09	77.77	91.89	87.18	71.47	0.94	81.75	90.38	16.64	<b>6</b>
w/o Prequel	10.28	48.86	79.65	92.66	88.49	73.17	0.95	83.25	91.04	15.89	<b>6</b>
w/o Current	10.32	48.78	79.62	93.05	88.34	73.13	0.95	83.25	91.04	15.86	<b>6</b>
w/o Sequel	10.32	48.91	79.34	93.05	88.57	72.86	0.95	83.09	91.25	15.81	<b>6</b>

Table 15: Full analysis of different components in our method. The top block shows the performance of our method with audio transcripts, while the bottom block represents the performance without audio transcripts. The first row in each block corresponds to our full method with all components included. In subsequent rows, one main component (video, query, Event) is removed at a time to demonstrate its contribution to the overall method. While Event consists of Prequel, Current and Sequel, on the bottom three rows, we have removed each of them one at a time to demonstrate the contribution of each. Results are reported on the MultiVENT dataset.

Dataset	# of query	Avg. words	# of video	Avg. length (sec.)
MSR-VTT-1kA	995	9	1000	15
MSVD	22285	8	670	10
MultiVENT	259	27	2394	83

Table 16: Overview of the datasets used in this study. The table summarizes key statistics of the MSR-VTT-1kA (Xu et al., 2016), MSVD (Chen and Dolan, 2011b) and MultiVENT (Sanders et al., 2023) datasets.

gual ASR model for query decomposition, frame captioning and audio transcription. For embedding computations, we employed ColBERT<sup>10</sup> for text-to-text similarity and MultiCLIP<sup>11</sup> for text-to-video similarity score calculation. Query decomposition and video captioning were performed with a temperature of 0.8 and top-p sampling of 0.95. For frame extraction, we applied uniform sampling with 16 frames. Our evaluation framework is based on torchmetrics<sup>12</sup>. The data extraction prompts were executed using two A100 GPUs, while the text-to-video retrieval was conducted on an RTX 6000 GPU.

## A.6 Prompts

In the following sections, we provide all the prompts used for the event decomposition (Apps. A.6.1 to A.6.3), video decomposition (Apps. A.6.4 and A.6.5) and ASR module (App. A.6.6).

### A.6.1 Decomposing Query to Prequel, Current, and Sequel Events

The following prompt was utilized to extract prequel, current, and sequel events from the query, with minimal modifications to maintain generalizability across all cases.

#### Prequel / Current / Sequel

You are given a video search query. Assume the video search query is an ongoing event. You have to find out what previous events could lead to this given event / decompose the ongoing event in multiple simple events / find out what events can be the outcome of this ongoing event. The extracted events should be concrete enough to be visualized in a video.

Here are some guidelines:

- Extract events that could be visualized in a video.
- Before extracting an event, consider whether the event text could serve as a reasonable caption for a frame of a video associated with the query.
- Give specific events related to the query, use your previous WORLD KNOWLEDGE to extract events that are likely to be seen in a video. For example, if the query is "2019

<sup>10</sup>[hltcoe/plaidx-large](https://hltcoe/plaidx-large)

<sup>11</sup>[laion/Clip-ViT-H14](https://laion/Clip-ViT-H14)

<sup>12</sup><https://lightning.ai/docs/torchmetrics>

Christmas Eve Protests in Hong Kong", you might return "Protesters marching in the streets", but not "Protestors marching in snow" because it is not likely to have snow in Hong Kong, though its pretty common in other places.

- First, explain your reasoning in 1-2 sentences about which events it makes sense to extract. Then, provide a numbered list with the events (anywhere from one to five event; use your best judgement to determine how many to extract)
- Structure your response into two sections, ""EXPLANATION:"" and then ""EVENTS:"". The events sections should contain nothing but the events.

Here is the query I'd like you to extract events from: {{query}}

### A.6.2 Extracting Spatial, Temporal, and Event Data

The following three prompts were utilized to extract the primary event, location, and time of the events from the query, respectively. The only input for these prompts is the original query. In cases where the query does not mention a specific time or location, the LLM was instructed to output "NOT AVAILABLE."

#### Extract Primary Event

You are given a video search query. Your task is to extract the primary event(s) being referenced or discussed in the query.

Here are some guidelines:

- Identify the main event(s) that the query is about, whether explicitly stated or implied
- If multiple events are mentioned, list all of them
- If no clear event can be identified from the query, output "NOT AVAILABLE" in the EVENTS section
- First, explain your reasoning in 1-2 sentences about how you identified the event(s) from the query
- Structure your response into two sections, "EXPLANATION:" and then "EVENTS:"
- The events section should contain either the identified event(s) in a numbered list, or "NOT AVAILABLE" if no event can be determined
- Don't include temporal or spatial information in this event, just focus on the primary event(s) being discussed or referenced

Here is the query I'd like you to extract events from: {{query}}

#### Extract Spatial Information

You are given a video search query. Assume the video search query is an ongoing event. Your task is to extract any locations mentioned in the query.

Here are some guidelines:

- Extract any location information mentioned in the query, including specific cities, towns, countries, regions, areas, neighborhoods, or other geographical references
- If no location information is found in the query, output "NOT AVAILABLE" in the LOCATION INFORMATION section
- First, explain your reasoning in 1-2 sentences about what location information appears in the query (or lack thereof) and why you're extracting it

- Structure your response into two sections, "EXPLANATION:" and then "LOCATION INFORMATION:"
- The location information section should contain either the extracted location information in one line, or "NOT AVAILABLE" if no location information is provided.
- Output in the following format: CITY, COUNTRY, REGION, AREA, NEIGHBORHOOD, etc. (e.g. New York, USA, Europe, Downtown). If one of the fields is not available, you can leave it out.

Here is the query I'd like you to extract location information from: {{query}}

#### Extract Temporal Information

You are given a video search query. Assume the video search query is an ongoing event. Your task is to extract any dates, times, or years mentioned in the query.

Here are some guidelines:

- Extract any temporal information mentioned in the query, including specific dates, times of day, years, seasons, or other time-related references
- If no temporal information is found in the query, output "NOT AVAILABLE" in the TEMPORAL INFORMATION section
- First, explain your reasoning in 1-2 sentences about what temporal information appears in the query (or lack thereof) and why you're extracting it
- Structure your response into two sections, "EXPLANATION:" and then "TEMPORAL INFORMATION:"
- The temporal information section should contain either the extracted date/time information in one line, or "NOT AVAILABLE" if no temporal information is provided.
- Output in the following format: DAY MONTH YEAR, TIME, SEASON, etc. (e.g. 1 January 2022, 3:00 PM, Summer). If one of the fields is not available, you can leave it out.

Here is the query I'd like you to extract temporal information from: {{query}}

### A.6.3 Refining Query

After extracting information from the previous prompts, the following prompt was employed to refine the prequel, current, and sequel events by incorporating the extracted event, temporal, and spatial details. The goal was to construct natural, coherent, human-like search queries while preserving all relevant information.

#### Refine Query

You will receive a query along with optional time, location, and main event information. Your task is to integrate this information naturally into a refined, human-like search query.

Here are some guidelines:

- Take the original query and incorporate any provided time, location, and event details seamlessly
- If any of these elements (time, location, event) are not provided, work with just the available information
- The refined query should sound natural, as if someone is searching with that query in Google, YouTube, etc.
- The base query information must always be preserved in the refined version

- First, explain your reasoning in 1-2 sentences about which events it makes sense to extract. Then, provide the refine query in one line
- Structure your response into two sections, ""EXPLANATION:"" and then ""REFINED QUERY:"". The Refined Query section should contain nothing but the refined query in one line.

Base Query: {{prequel/Current/sequel}}  
 Event: {{event}}  
 Place: {{place}}  
 Time: {{time}}

#### A.6.4 Frame Captioner

The following prompt was used to generate captions for each video frame. Unlike other prompts, a VLM was utilized to handle the multimodal inputs effectively. As illustrated in Fig. 2, we implemented two variants: (1) ASR and (2) Non-ASR. The **Orange** texts were excluded in the prompt for the Non-ASR variant.

##### Frame Caption

You are provided with the description of the previous frames, the current frame (image) of a single video, **and the original transcript of the audio (ASR) from the entire video**. Your task is to describe the main event or activity depicted in the current frame.

###### # Instructions:

- **Use the original ASR to enhance your understanding of the video context and inform your description of the current frame.**
- Focus on identifying and detailing the key actions, participants, and context of the event depicted in the current frame.
- Maintain continuity with the previous frame's description, connecting the current frame to the overall flow of the video.
- Ignore unrelated visual elements like colors or shapes unless they are critical to understanding the event.
- Use any visible text in the image to enhance your interpretation of the frame.

Only output the current frame description, integrating insights from **the original ASR and** previous frame description. Don't output any additional commentary or content.

# Original ASR Transcript (Entire Video):  
 {{ASR}}

# Previous Frame Description:  
 {{prev\_caption}}

# Frame:  
 {{frame}}

# Current Frame Description:

#### A.6.5 Full Video Captioner

The following prompt generated a summarized caption for the entire video. Given the high computational requirements and the limited capability of VLMs in handling videos, we utilized an LLM to summarize all the captions extracted in the previous

step. The full refined audio transcript was also included in the ASR variant to provide additional context. As before, the **Orange** text elements were excluded from the prompt in the Non-ASR variant.

##### Video Caption using Frame Captions

You will receive individual descriptions of consecutive frames from a single video, **along with the original transcript of the audio (ASR) from the entire video**. Your task is to provide a detailed description of the whole video, summarizing the key actions, participants, and context of the event depicted in the video.

###### # Instructions:

- **Use the original ASR to enhance your understanding of the video context and inform your description of the whole video.**
- Focus only on relevant aspects, such as what is happening, who is involved, and the overall context or setting.
- Ignore unrelated visual details like colors, shapes, or other non-essential information.
- Your summary should concisely and accurately capture the essence of the event, integrating insights from the frame descriptions, **the Original ASR and the ASR summary**.

Only output the video summary, without any additional commentary or content.

###### # Frame Descriptions:

## Frame [i] Description  
 {{frame\_i\_description}}

# Original ASR Transcript (Entire Video):  
 {{ASR}}

###### # Summary:

#### A.6.6 Refining Audio Transcripts

As illustrated in Fig. 2 (left orange box), we utilized an LLM to refine the translations generated by the multilingual-ASR and a translation-specific model. The following prompt was employed to refine the audio transcript using three inputs: (1) the original language transcript from ASR, (2) the translated transcript from the multilingual-ASR, and (3) the translated transcript from the translator.

##### Refine Audio Transcript

You are an expert in language translation and text refinement. Your task is to produce a highly accurate and natural-sounding translation by analyzing the original transcript and two provided translations.

###### # Instructions:

- Combine the strengths of both translations and use the original transcript as a reference to improve accuracy and contextual meaning.
- Prioritize clarity, fluency, and maintaining the original intent and tone of the text.
- Where translations differ significantly, cross-check with the original transcript for fidelity.
- Ensure the final output is free from any awkward phrasing or mistranslation.
- If the transcripts can't be refined with reasonable accuracy, just output "Not Available".

###### # Transcripts:

```
## Original Transcript:  
{ {original_language_asr} }  
## Translation 1:  
{ {whisper_translated_asr} }  
## Translation 2:  
{ {nllb_translated_asr} }  
Output the refined translation only, without additional notes  
or explanations.  
# Refined Translation:
```

### A.7 Use of AI Assistance

The authors have used GitHub copilot <sup>13</sup> during development and ChatGPT<sup>14</sup> for proofreading and polishing the final writing. Content provided to those tools were original to the authors.

---

<sup>13</sup><https://github.com/features/copilot>

<sup>14</sup><https://chatgpt.com/>