FastV-RAG: Towards Fast and Fine-Grained Video QA with Retrieval-Augmented Generation

Gen Li
University of Electronic Science and Technology of China
Chengdu, China
leog3n@gmail.com

Peiyu Liu*
University of International Business and Economics
Beijing, China
liupeiyustu@163.com

Abstract

Vision–Language Models (VLMs) excel at visual reasoning but still struggle with integrating external knowledge. Retrieval-Augmented Generation (RAG) is a promising solution, but current methods remain inefficient and often fail to maintain high answer quality. To address these challenges, we propose *VideoSpeculateRAG*, an efficient VLM-based RAG framework built on two key ideas. First, we introduce a speculative decoding pipeline: a lightweight draft model quickly generates multiple answer candidates, which are then verified and refined by a more accurate heavyweight model, substantially reducing inference latency without sacrificing correctness. Second, we identify a major source of error—incorrect entity recognition in retrieved knowledge—and mitigate it with a simple yet effective similarity-based filtering strategy that improves entity alignment and boosts overall answer accuracy. Experiments demonstrate that VideoSpeculateRAG achieves comparable or higher accuracy than standard RAG approaches while accelerating inference by approximately $2\times$ speedup. Our framework highlights the potential of combining speculative decoding with retrieval-augmented reasoning to enhance efficiency and reliability in complex, knowledge-intensive multimodal tasks. Our code is available at https://github.com/FastVRAG/FastV-RAG.

1 Introduction

Vision–Language Models (VLMs) have advanced multimodal understanding, enabling machines to interpret visual inputs and generate contextually relevant text. Despite impressive progress (Lin et al., 2023; Zhu et al., 2023; Team, 2025; Hurst et al., 2024), many real-world scenarios remain knowledge-intensive and require external information beyond the visual input. Retrieval-Augmented Generation (RAG Lewis et al., 2020; Luo et al., 2024) offers a promising solution, yet existing multimodal RAG approaches still incur high computational costs and struggle to maintain accuracy when integrating large knowledge sources or long-context inputs. This motivates the need for a more efficient and reliable generation framework for multimodal RAG.

To address such knowledge-intensive tasks, VLMs are increasingly combined with RAG, enabling them to retrieve relevant external information to support answer generation. In most existing approaches (Izacard and Grave, 2020; Shuster et al., 2022; Chen et al., 2022), the retrieved passages are directly concatenated with the original multimodal inputs and processed jointly by the model. While this strategy enhances factual grounding and broadens the accessible knowledge scope, it also introduces several critical challenges. Traditional RAG pipelines often suffer from inefficiency, as inference latency and computational cost increase sharply with the number and length of retrieved documents. Moreover, in multimodal settings, the alignment between retrieved textual entities and visual entities is frequently imperfect, leading to mismatched reasoning and degraded accuracy. Overcoming these limitations is essential for building VLM–RAG systems that are both efficient and reliable, capable of reasoning coherently over complex visual and textual inputs.

We are motivated by speculative decoding (Cai et al., 2024; Wang et al., 2024), which is designed to improve efficiency by generating multiple candidate outputs in parallel using a lightweight draft model and then verifying or refining these candidates with a more accurate heavyweight model. This separation of drafting and verification maintains high output quality while reducing inference latency compared with standard sequential decoding. Our key idea is to let a lightweight model handle multimodal retrieval and answer generation,

*Corresponding author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 879–889

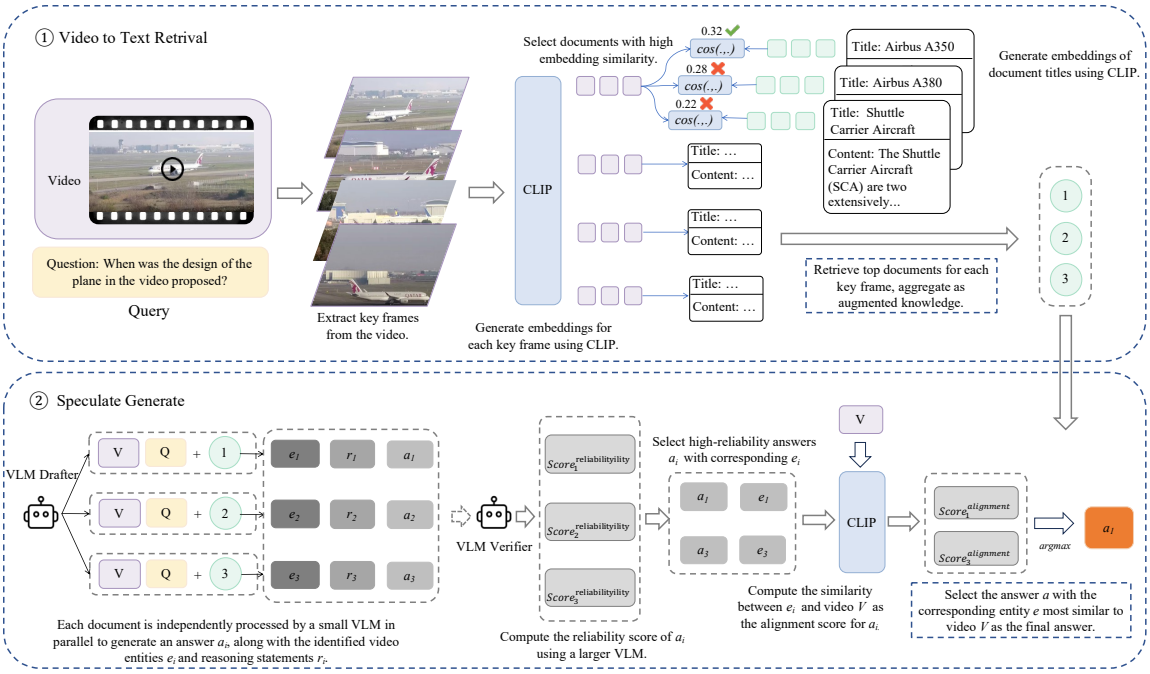

Figure 1: Illustration of Video Speculate RAG.

while invoking a stronger model only for calibration. When applied to VLM–RAG systems such as KVQA, this speculative decoding paradigm enables the draft stage to rapidly propose answers from retrieved documents, and the verification stage to refine and validate them, substantially reducing inference cost while preserving accuracy and visual grounding.

Moreover, we observe that naïve speculative decoding may still produce errors caused by fine-grained entity confusion, where superficially similar but incorrect entities in retrieved texts mislead the model. To address this, we introduce a fine-grained entity alignment mechanism: the drafter explicitly extracts entities and reasoning traces during answer generation, and the verifier measures alignment between candidate answers and video frames via CLIP-based similarity. This two-stage verification ensures that the final answer is both factually accurate and visually grounded, effectively mitigating entity-level errors while retaining the efficiency advantages of speculative decoding.

Finally, we evaluate VideoSpeculateRAG on two KVQA benchmarks, VideoSimpleQA (Antol et al., 2015) and Encyclopedic VQA (Mensink et al., 2023), considering both answer accuracy and inference latency. Compared with the standard RAG framework, our approach achieves comparable or improved accuracy while providing nearly a 2× speedup, highlighting its effectiveness in enhancing answer reliability and computational efficiency for complex multimodal reasoning tasks.

2 Related Work

Knowledge-aware VQA. Knowledge-aware VQA refers to a class of question-answering tasks that require external knowledge, with questions and answers potentially spanning multiple modalities such as text, video, and images. Marino et al. (2019); Schwenk et al. (2022); Chen et al. (2023b); Mensink et al. (2023) provide sets of image-based questions whose answers cannot be directly inferred from the images themselves but instead rely on external textual knowledge. Zhan et al. (2025); Ma et al. (2024) extend beyond textual knowledge: some of their questions require retrieving information from external images to answer. VideoSimpleQA (Cao et al., 2025) offers a collection of video–text question–answer pairs designed for evaluating question-answering systems in video-based contexts.

Speculative Decoding. Speculative decoding (Stern et al., 2018; Chen et al., 2023a; Leviathan et al., 2023) is an efficient generation strategy designed to accelerate large language model (LLM) inference by coupling a lightweight draft model with a more powerful target model. In conventional autoregressive decoding, the target model must generate tokens sequentially,

which is computationally expensive and leads to high latency, especially for long sequences. Speculative decoding mitigates this inefficiency by allowing the draft model to rapidly produce multiple candidate tokens in parallel, effectively "speculating" on what the target model would generate. These proposed tokens are then passed to the target model for verification: if the target model's predictions match the draft's proposals, they are accepted directly, enabling the system to advance multiple tokens in a single step. If discrepancies occur, only the mismatched tokens are regenerated by the target model, ensuring correctness. This mechanism substantially reduces the number of costly forward passes through the large model while maintaining output fidelity, thereby offering a practical balance between computational efficiency and generation quality.

Algorithm 1: Video-to-Text Retrieval with Two-Stage Verification

Input: Video V, Question Q
Output: Final answer a^*
$F \leftarrow \varnothing$; // Initialize keyframe set
for *each frame* f_i *in* V **do**
  if $sim(f_i, f_{i-1}) < \theta$ **then**
    $F \leftarrow F \cup \{f_i\}$; // Eq.(1)
for *each keyframe* $f \in F$ **do**
  retrieve $T^* \leftarrow$ Top-K $\text{sim}(f, t)$; // Eq.(2,3)
for *each* $t_i \in T^*$ **do**
  $e_i \leftarrow \text{Drafter}(V, t_i)$; // Eq.(8)
  $r_i \leftarrow \text{Drafter}(V, Q, e_i, t_i)$; // Eq.(9)
  $a_i \leftarrow \text{Drafter}(V, Q, e_i, r_i)$; // Eq.(10)
for *each tuple* (a_i, r_i) **do**
  $\text{Score}_i^{\text{reliability}} \leftarrow \frac{p_i^{\text{Yes}}}{p_i^{\text{Yes}} + p_i^{\text{No}}}$; // Eq.(12)
$A_H \leftarrow \{(a_i, e_i) \mid \text{Score}_i^{\text{reliability}} \geq \max_j \text{Score}_j^{\text{reliability}} - \delta\}$; // Eq.(13)
for *each* $a_i \in A_H$ **do**
  $\text{Score}_{\text{alignment}}(a_i) \leftarrow \max_{f \in F} \cos(\text{CLIP}(e_i), \text{CLIP}(f))$; // Eq.(14,15)
$a^* \leftarrow \arg\max_{a_i \in A_H} \text{Score}_{\text{alignment}}(e_i)$; // Select final answer
return a^*

3 Methods

In this work, we aim to enhance the efficiency and reliability of multimodal RAG by integrating speculative decoding with targeted answer verification. The core idea is to delegate most retrieval-conditioned generation to a lightweight VLM and invoke a stronger VLM only for calibration when needed. Section 3.1 presents the proposed *VideoSpeculateRAG* framework, and Section 3.2 introduces an additional alignment enhancement to further improve answer accuracy.

3.1 VideoSpeculateRAG

Multimodal Retrieval. Multimodal RAG aims to enhance video question answering by leveraging relevant external textual knowledge, enabling the model to reason over both visual content and supporting text. In this framework, the retrieval stage identifies and collects external documents that are semantically relevant to the video content and the posed question, providing the generation model with pertinent background knowledge. The subsequent generation stage produces the final answer by conditioning on both the video input and the retrieved textual knowledge, integrating multimodal information to generate accurate and contextually grounded responses.

To enable video-to-text retrieval, we first extract keyframes from the video. For each frame in video V, we compute its similarity with the previous frame. If the similarity falls below a predefined threshold, the frame is added to the keyframe set F, as shown in Eq. (1):

$$\text{sim}(f_i, f_{i-1}) < \theta, F \leftarrow F \cup \{f_i\}. \quad (1)$$

Here, histograms are used to represent inter-frame similarity. For each keyframe f, we retrieve its top-k most similar texts and aggregate them into a text set T. Given an image frame and a text, their similarity is computed using CLIP embeddings and cosine similarity, as shown in Eq. (2):

$$\text{sim}(f, t) = \cos\big(\text{CLIP}(f), \text{CLIP}(t)\big). \quad (2)$$

We select the top-K texts from the set T with the highest similarity scores, which serve as external knowledge to enhance the VLM input, as shown in Eq. (3):

$$T^* = \underset{t \in T}{\text{Top-}K}\ \text{sim}(f, t). \quad (3)$$

Finally, the retrieved external texts are concatenated after the question, and the VLM generates the answer A based on the video V, the question Q, and the text set T^*:

$$a \sim \text{VLM}(Q, V, T^*). \quad (4)$$

An analysis of the retrieval module's quality (Recall@k) is provided in Appendix A.2.

Speculative Generation. Speculative decoding extends the generation stage of standard multimodal RAG by introducing a two-stage verification process that improves both efficiency and answer reliability. In this framework, a lightweight model first rapidly generates draft answers for each retrieved document, producing multiple candidate responses in parallel. These drafts are then evaluated during the verification stage, where a scoring mechanism determines the most reliable answer to be selected as the final output.

Unlike standard multimodal RAG, after retrieving the external document set T^*, we do not concatenate all documents into a single input. Instead, each document $t_i \in T^*$ is appended to a separate input, and these inputs are processed in parallel by a lightweight draft VLM to rapidly generate a set of draft answers:

$$A_{\text{draft}} = \{a_i \sim \text{VLM}_{\text{Drafter}}(Q, V, t_i) \mid t_i \in T^*\}. \tag{5}$$

Next, a larger verifier VLM evaluates each draft answer $a_i \in A_{\text{draft}}$, computing a corresponding score:

$$\text{score}_i = \{\text{VLM}_{\text{Verifier}}(Q, V, a_i) \mid a_i \in A_{\text{draft}}\}. \tag{6}$$

Finally, the draft with the highest score is selected as the final answer:

$$a^* = \text{argmax}(\{\text{score}_i \mid a_i \in A_{\text{draft}}\}). \tag{7}$$

By parallelizing draft generation over individual documents, the framework mitigates the computational overhead and memory burden induced by long-context concatenation, thereby enhancing efficiency while preserving answer reliability.

3.2 Fine-grained Entity Alignment

While speculative decoding efficiently generates multiple draft answers and allows the verifier to select the most likely one, this process alone may not fully address errors arising from fine-grained entity confusion. Specifically, some drafts can appear plausible yet contain subtle mistakes caused by retrieved documents referencing entities that are visually or semantically similar but not identical to those in the video. To systematically detect such errors during verification, we introduce a fine-grained entity alignment mechanism.

Error Analysis. Given a video V, suppose that an erroneous document T_{error} containing entities that are superficially similar to but inconsistent with the actual video content is provided to the VLM. In this situation, the model can be misled into producing incorrect answers. We categorize such error scenarios into two representative types, assuming that the actual entity in the video is E_{actual} and the erroneous entity in the text is E_{error}:

• *Cross-Entity Transfer*: The VLM successfully identifies the actual visual entity E_{actual} within the video, yet its reasoning is influenced by external knowledge associated with E_{error}. This contamination leads the model to apply attributes or contextual information from the erroneous entity to the correct one. As a result, the generated answer often carries subtle traces of E_{error}, giving rise to inconsistencies at the entity level. While such answers may retain a degree of surface-level alignment with the video, they ultimately reflect a fragile and unreliable grounding.

• *Entity Substitution*: The VLM is fully influenced by the erroneous document, resulting in a final answer that is grounded entirely on the incorrect entity E_{error} rather than the true visual entity E_{actual}. The reasoning process is dominated by the misleading text, producing outputs that are internally coherent and linguistically fluent, yet detached from the actual content of the video. The answer may superficially appear plausible, but it fails to reflect the correct fine-grained entities and their relationships present in the visual input. As a consequence, while the response may seem consistent and self-contained, it systematically replaces the authentic entity information with spurious attributes.

Structured Draft Reasoning. To enable the verification stage to identify such errors, we design a multi-hop reasoning procedure in which the draft VLM explicitly extracts entities from the video and structures its reasoning trajectory. This structured information equips the subsequent verification with the necessary signals to detect misleading drafts. Given a video V, a question Q, and a retrieved text t_i, the drafter first extracts the salient entity e_i from the video:

$$e_i = \text{VLM}_{\text{Drafter}}(V, t_i). \tag{8}$$

Conditioned on this entity and the question, the model incorporates the retrieved text t_i as evidence to form an intermediate reasoning statement r_i, explicitly linking the entity, the question, and the

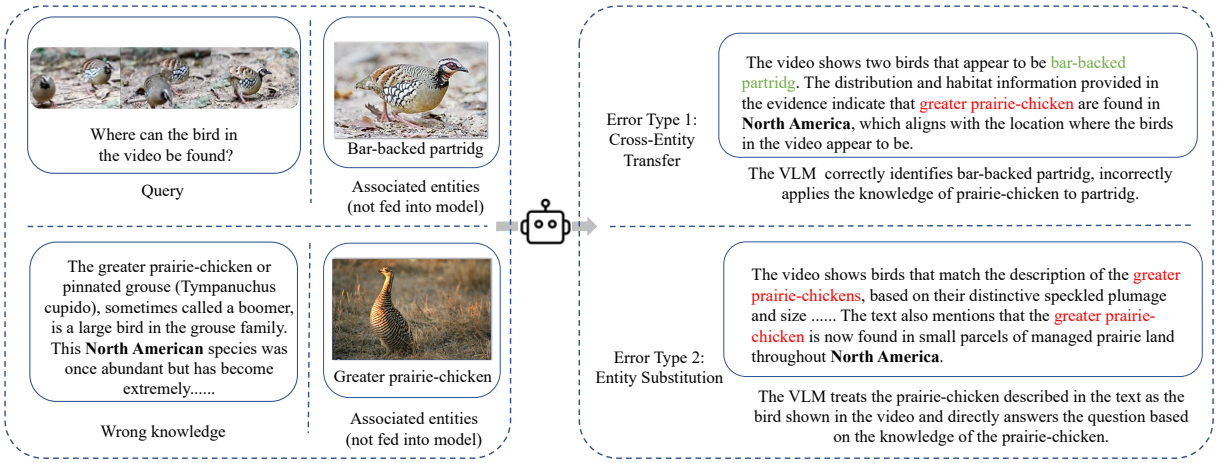

Figure 2: Error Analysis. RED color stands for entities inconsistent with the video content, while GREEN color refers to entities consistent with the video.

supporting text:

$$r_i = \text{VLM}_{\text{Drafter}}(V, Q, e_i, t_i). \tag{9}$$

Finally, leveraging both the extracted entity and the reasoning statement, the drafter produces a candidate answer a_i:

$$a_i = \text{VLM}_{\text{Drafter}}(V, Q, e_i, r_i), A_{\text{Draft}} \leftarrow A_{\text{Draft}} \cup \{a_i\}. \tag{10}$$

Two-Stage Verification. After generating candidate answers a_i along with the corresponding evidence e_i and reasoning statements r_i, these drafts are evaluated through a two-stage verification process to select the final answer. In the first stage, the verifier VLM assesses the reliability of each candidate, determining whether the reasoning supports the answer. A single forward pass computes the probability that the first generated token is "Yes":

$$p_i^{\text{Yes}} = P(\text{"Yes"} \mid Q, V, a_i, e_i, r_i), \tag{11}$$

and similarly for "No". The reliability score is then calculated as:

$$\text{Score}_i^{\text{reliable}} = \frac{p_i^{\text{Yes}}}{p_i^{\text{Yes}} + p_i^{\text{No}}}. \tag{12}$$

Then we select candidate answers A_H whose scores satisfy the following requirement:

$$\text{Score}_i^{\text{reliable}} \geq \max_{a_j \in A_{\text{Draft}}} (\text{Score}_j^{\text{reliable}}(a_j)) - \delta, \tag{13}$$

where δ denotes a tolerance margin from the maximum reliability score.

Even within A_H, the entities described in the candidate answers may still differ subtly from the actual video entities. To mitigate this discrepancy, we compute an entity alignment score for each candidate answer $a_i \in A_H$ by measuring the similarity between its corresponding entity e_i and the associated video frames f_i using CLIP:

$$\text{sim}(e_i, f_j) = \cos\big(\text{CLIP}(e_i), \text{CLIP}(f_j)\big). \tag{14}$$

The maximum similarity over all frames is taken as the alignment score:

$$\text{Score}_{\text{alignment}}(a_i) = \max_{f_i \in F} \text{sim}(e_i, f_i). \tag{15}$$

The candidate with the highest alignment score is then selected as the final answer, as illustrated in Algorithm 1.

4 Experiments

We conduct experiments on video and image datasets to evaluate the accuracy and inference efficiency of our system on the KVQA task.

4.1 Datasets

VideoSimpleQA. VideoSimpleQA (Cao et al., 2025) is a video-based KVQA dataset that consists of videos collected from Wikimedia Commons and questions extracted from corresponding Wikipedia entries. For our experiments, we curate a subset of VideoSimpleQA by selecting videos that are clear and contain unambiguous entities, along with their associated questions, to serve as our video KVQA dataset.

Encyclopedic VQA. Due to the lack of established benchmarks for video-based KVQA, we additionally adopt an image-based KVQA dataset, Encyclopedic VQA (Mensink et al., 2023), hereafter

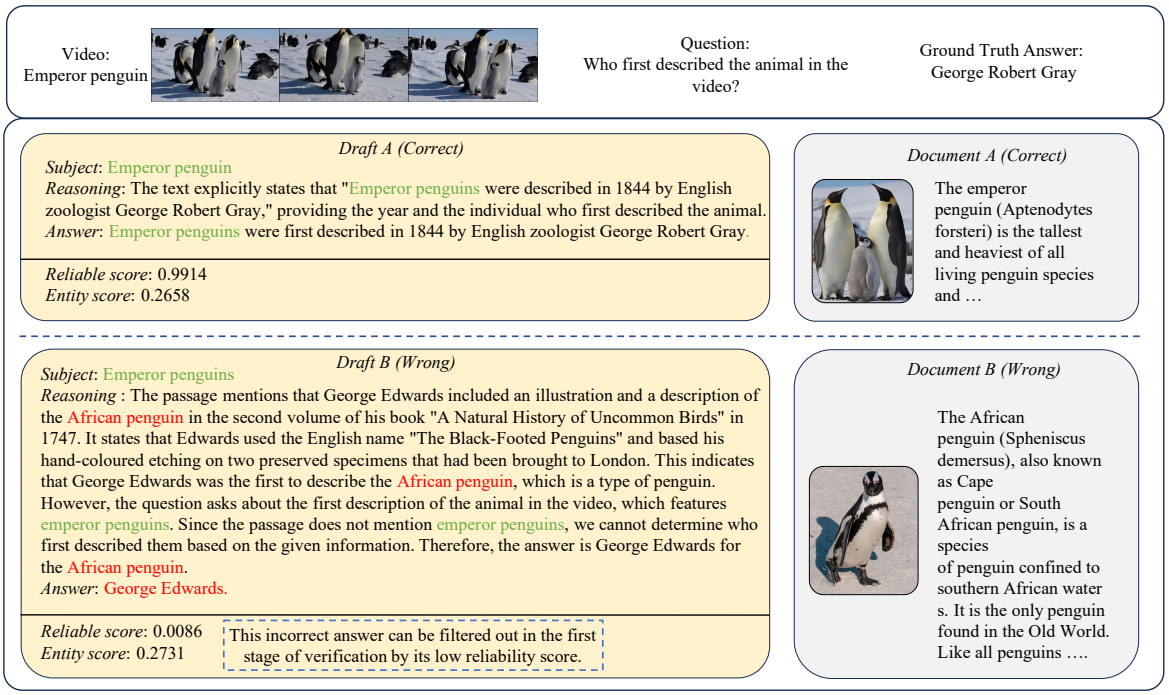

Figure 3: An illustrative example of how our method detects Cross-Entity Transfer.

referred to as EncVQA. This dataset is constructed from images in the iNaturalist 2021 (Van Horn et al., 2021) and Google Landmarks Dataset V2 collections (Weyand et al., 2020), with questions derived from corresponding Wikipedia entries. All datasets used in this work are open-source and used in accordance with their respective licenses: VideoSimpleQA is under CC BY-NC-SA 4.0, Google Landmarks Dataset V2 is under CC BY 4.0, and iNaturalist 2021 is under CC BY-NC.

4.2 Baselines

No RAG. The video and the question are directly fed into the VLM, which produces the answer solely based on its internal knowledge. This baseline evaluates the inherent capability of VLMs to answer knowledge-intensive questions without external documents. We evaluate several open-source VLMs, including *Qwen2.5-VL-Instruct 3B*, *Qwen2.5-VL-Instruct 32B*, *LLaVA-NeXT-Video 34B*, and *InternVL3 38B*. More implementation details can be found in Appendix A.1.

Standard RAG. Retrieved documents are directly concatenated into the prompt alongside the visual content and the question, and then passed into the VLM to produce answers. This baseline assesses the model's ability to comprehend the retrieved text and align it with the video content. The same set of open-source VLMs used in the No RAG baseline are evaluated here.

4.3 Implementation Details

For each dataset, we employ *clip-vit-large-patch14-336* in the retrieval stage to compute the similarity between the text and visual content, retrieving $k = 3$ documents for each query. We employ *Qwen2.5-VL-Instruct-3B* as the Drafter and *Qwen2.5-VL-Instruct-32B* as the Verifier, with δ set to 0.05, which yields the best empirical performance in our experiments. All inference experiments were conducted on 2×A100 80GB GPUs, consuming approximately 100 GPU hours in total.

4.4 Main Results

VideoSpeculateRAG can achieve high accuracy on KVQA tasks. As shown in Table 1, without external knowledge, VLMs struggle to answer questions in both VideoSimpleQA and EncVQA: for instance, *Qwen2.5-VL-Instruct-32B* achieves only 57.73% and 22.00% accuracy on the two datasets, respectively. With external knowledge augmentation, performance improves substantially. On VideoSimpleQA and EncVQA, *Qwen2.5-VL-Instruct-32B (RAG)* achieves 91.30% and 46.75% accuracy, corresponding to relative improvements of 58.15% and 112.50% over the non-RAG setting. Under the speculative paradigm, our method matches or even surpasses the performance of standard RAG. Specifically, *Qwen2.5-VL-Instruct-32B*

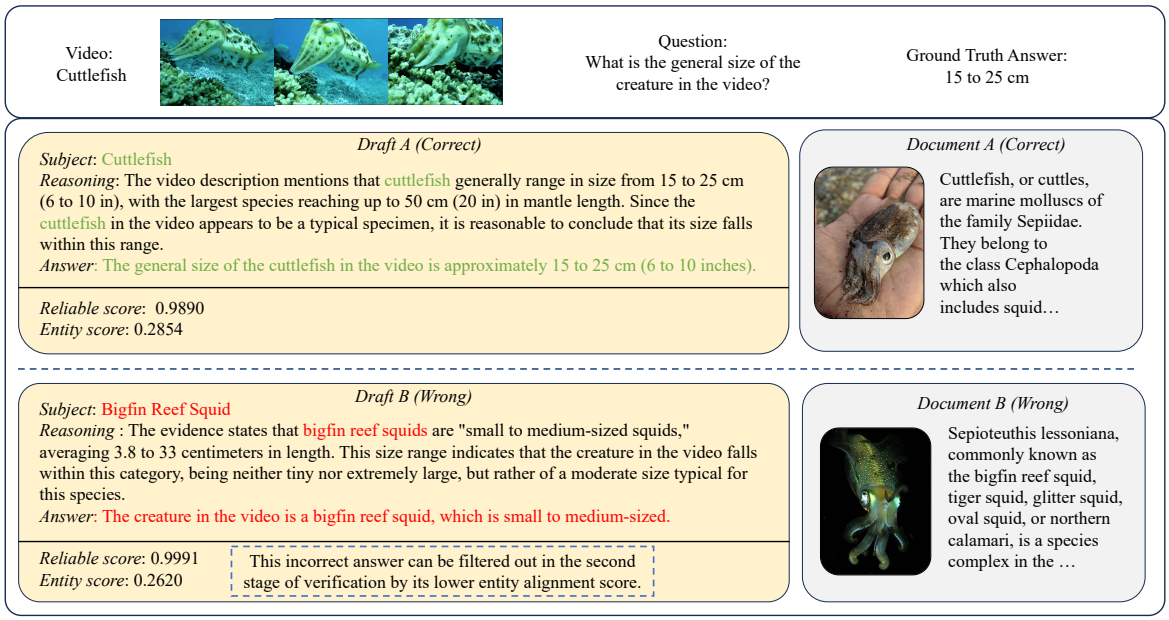

Figure 4: An illustrative example of how our method detects Entity Substitution.

(VideoSpeculateRAG) attains 91.12% accuracy on VideoSimpleQA—comparable to the 91.30% of standard RAG—and 47.64% accuracy on EncVQA, outperforming the 46.75% of standard RAG. To further validate generality, we additionally evaluate with the Qwen-3-VL series as the backbone. *Qwen3-VL-Instruct-4B+32B (VideoSpeculateRAG)* achieves 90.45% on VideoSimpleQA and 48.50% on EncVQA, consistently matching or exceeding the corresponding standard RAG results (91.17% and 46.72%), confirming that our framework generalizes well across different VLM backbones.

VideoSpeculateRAG can effectively reduce the inference latency of RAG. Our experiments show that VLM inference time increases substantially with longer input contexts. After augmenting with external documents, *Qwen2.5-VL-Instruct-32B* exhibits a latency increase from 19.18s to 47.72s on VideoSimpleQA, and from 20.13s to 40.42s on EncVQA. While, *Qwen2.5-VL-Instruct-3B+32B (VideoSpeculateRAG)* significantly alleviates this latency growth, achieving 25.74s and 16.61s on the two datasets, corresponding to reductions of 46.06% and 58.90% compared to *Qwen-32B (RAG)*, and even outperforming the non-RAG in terms of speed. This improvement primarily stems from two factors: (i) the lightweight draft model processes inputs efficiently, and (ii) splitting the retrieved documents mitigates the effect of input context expansion.

4.5 Ablation Study

In the ablation study, we progressively remove the reliability score and the entity alignment score. Removing the reliability score causes a sharp accuracy drop (25.41% on VideoSimpleQA and 4.44% on EncVQA), with a latency reduction of about 4s, mainly due to skipping one VLM forward pass. Removing the alignment score decreases accuracy by 5.82% and 9.62% on the two datasets, while reducing latency by about 1s, since CLIP is faster than the 32B VLM. When both components are removed, the accuracy of *VideoSpeculateRAG* collapses to roughly half of the original. These results confirm that reliability estimation and entity alignment are both critical and complementary.

4.6 Further Analysis

Case Studies. We illustrate how our framework addresses the errors analyzed in Section 3.2 through two representative examples. In Figure 3, the video shows an emperor penguin. Retrieved documents cover both emperor and African penguins. Draft A and Draft B achieve similar entity alignment, but Draft B incorporates incorrect external knowledge and thus receives a much lower **reliability score**. Leveraging this score, our framework filters out Draft B in the first-stage verification and selects Draft A as the final answer. In Figure 4, the video shows a cuttlefish, but documents about both cuttlefish and squid are retrieved. Draft A correctly recognizes the cuttlefish, while

Methods	VideoSimpleQA	Latency (s)	EncVQA	Latency (s)	Avg Accuracy	Avg Latency (s)
			No RAG			
Qwen-2.5VL-Instruct-3B	37.62	2.15	14.00	3.30	25.81	2.72
Qwen-2.5VL-Instruct-32B	57.73	19.18	22.00	20.13	39.87	19.65
Qwen-3VL-Instruct-4B	31.02	5.59	18.37	5.27	24.70	5.43
Qwen-3VL-Instruct-32B	48.97	11.01	19.00	13.39	33.99	12.20
LLaVA-NeXT-Video-34B	27.31	29.24	8.50	25.68	17.92	27.46
InternVL-3-38B	40.20	4.06	13.50	5.61	26.85	4.84
			Standard RAG			
Qwen-2.5VL-Instruct-3B	75.26	4.84	39.00	4.76	57.13	4.80
Qwen-2.5VL-Instruct-32B	**91.30**	47.72	46.75	40.42	69.03	44.07
Qwen-3VL-Instruct-4B	78.14	10.20	41.33	9.37	59.74	9.79
Qwen-3VL-Instruct-32B	<u>91.17</u>	36.68	46.72	26.48	68.95	31.43
LLaVA-NeXT-Video-34B	76.80	56.54	37.50	51.37	57.15	53.96
InternVL-3-38B	78.86	14.00	43.50	7.75	61.18	10.88
			Speculate RAG			
Qwen-2.5VL-Instruct-3B+7B	82.60	23.04	41.00	13.03	61.80	19.54
Qwen-2.5VL-Instruct-3B+32B	91.12	25.74	<u>47.64</u>	16.61	<u>69.38</u>	21.18
Qwen-3VL-Instruct-4B+8B	78.75	23.43	46.50	12.96	62.63	18.20
Qwen-3VL-Instruct-4B+32B	90.45	25.91	**48.50**	14.42	**69.48**	20.17

Table 1: Main results of vision-language models on knowledge-intensive video QA tasks. The annotation after each model indicates the model size. Values under each task indicate accuracy (higher is better), while the latency represents inference time (lower is better). The best results are highlighted in **bold**, while the second best are <u>underlined</u>.

Methods	VideoSimpleQA	Latency	EncVQA	Latency
Ours	91.12	25.74	47.64	16.61
w/o ett	85.30	24.14	38.02	15.31
w/o rel	65.71	21.62	43.22	12.46
Random	47.59	20.85	27.08	11.59

Table 2: Ablation studies. Here, "ett" stands for entity alignment score, while "rel" stands for reliability score.

Draft B is fully misled and confidently predicts squid. Both pass the first-stage reliability check, but in the second stage Draft A achieves a higher **entity alignment score**, allowing our framework to select it as the final answer. These results further highlight the necessity of the proposed two-stage verification.

Strategies	Model	VideoSimpleQA	EncVQA
Self Consistent	3B+7B	65.59	29.00
	3B+32B	68.63	36.50
Addition	3B+7B	73.91	42.00
	3B+32B	87.57	46.39
Invert	3B+7B	73.33	40.10
	3B+32B	74.86	41.23

Table 3: Comparison of verification scoring strategies. We compare three key scoring methods: "Self Consistent", "Addition", and "Invert".

Analysis of Scoring Mechanisms. To study the impact of verification strategies, we compare three scoring mechanisms using different model setups, where "3B+7B" and "3B+32B" denote a 3B draft model paired with a 7B or 32B verifier. The *self-consistent* strategy, adopted from Speculative RAG, relies solely on the verifier's internal likelihoods. The *addition* strategy directly sums the reliability and alignment scores, while the *invert* strategy first filters candidates by alignment score and then ranks them by reliability. As shown in Table 3, the *self-consistent* method performs the worst, as its confidence measure is affected by token length and fails to capture fine-grained multimodal alignment. The *addition* strategy achieves the best overall results, though imperfect calibration may arise since reliability and alignment are measured in different semantic spaces. The *invert* approach yields moderate but unstable gains due to the limited discriminative power of the entity alignment score in the first-stage filtering.

Analysis of Settings of δ. The tolerance margin δ in Eq. 13 controls the threshold for selecting high-reliability candidates. We evaluate the impact of different δ values on answer accuracy to analyze how this parameter affects the performance of *VideoSpeculateRAG*. As shown in Figure 5, the accuracy on both datasets first increases and then decreases as δ grows, reaching its peak when $\delta = 0.05$. Smaller δ values cause the system to rely almost exclusively on reliability scores, leading to more frequent *Entity Substitution* errors,

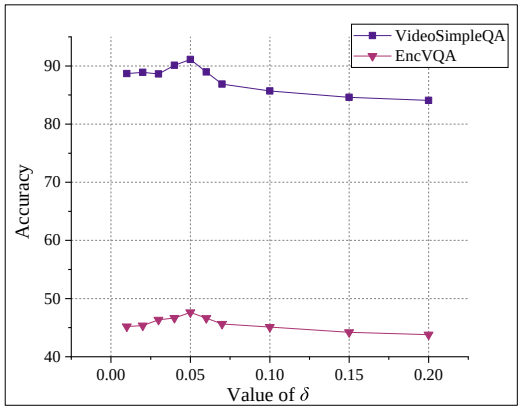

Figure 5: Accuracy variation with different δ values.

whereas larger δ values make the system depend excessively on entity alignment, resulting in more *Cross-Entity Transfer* errors.

Impact of Entity Alignment Models. Our entity alignment mechanism uses CLIP-based similarity by default. We additionally evaluate alternative vision-language embedding models, including different CLIP variants and SigLIP2 (Tschannen et al., 2025), to assess the flexibility of the approach. Results show that performance remains comparable across models, confirming that our method is not overly dependent on a specific CLIP version. Details are provided in Appendix A.3.

5 Conclusion

In this work, we propose VideoSpeculateRAG, a multimodal RAG framework that combines speculative decoding with two-stage answer verification, where a lightweight VLM drafts retrieval-conditioned answers and a stronger VLM calibrates them to reduce inference latency without sacrificing accuracy. To further enhance reliability, we introduce an entity-aware verification strategy to resolve fine-grained entity confusion. Experiments on VideoSimpleQA and Encyclopedic VQA show that our method matches or outperforms standard RAG and alternative speculative methods, achieving higher accuracy with substantially lower inference cost. We believe this approach paves the way for real-time, knowledge-intensive video interaction in future multimodal systems.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China under Grant No. 62506077. Peiyu Liu is the corresponding author.

Limitations

For a fair comparison with prior work, we evaluate VideoSpeculateRAG using widely adopted foundation vision–language models. However, the performance of video RAG may be constrained by the underlying model capability, which we leave for future investigation. In addition, our experiments focus on knowledge-intensive video QA datasets, while real-world scenarios can be more diverse and complex; extending the analysis to broader tasks and settings remains an important direction for future work.

References

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. 2015. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433.

Tianle Cai, Yuhong Li, Zhengyang Geng, Hongwu Peng, Jason D Lee, Deming Chen, and Tri Dao. 2024. Medusa: Simple llm inference acceleration framework with multiple decoding heads. *arXiv preprint arXiv:2401.10774*.

Meng Cao, Pengfei Hu, Yingyao Wang, Jihao Gu, Haoran Tang, Haoze Zhao, Jiahua Dong, Wangbo Yu, Ge Zhang, Ian Reid, and 1 others. 2025. Video simpleqa: Towards factuality evaluation in large video language models. *arXiv preprint arXiv:2503.18923*.

Charlie Chen, Sebastian Borgeaud, Geoffrey Irving, Jean-Baptiste Lespiau, Laurent Sifre, and John Jumper. 2023a. Accelerating large language model decoding with speculative sampling. *Preprint*, arXiv:2302.01318.

Wenhu Chen, Hexiang Hu, Xi Chen, Pat Verga, and William W Cohen. 2022. Murag: Multimodal retrieval-augmented generator for open question answering over images and text. *arXiv preprint arXiv:2210.02928*.

Yang Chen, Hexiang Hu, Yi Luan, Haitian Sun, Soravit Changpinyo, Alan Ritter, and Ming-Wei Chang. 2023b. Can pre-trained vision and language models answer visual information-seeking questions? *Preprint*, arXiv:2302.11713.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Gautier Izacard and Edouard Grave. 2020. Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282*.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. 2023. Fast inference from transformers via speculative decoding. In *International Conference on Machine Learning*, pages 19274–19286. PMLR.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. 2023. Video-llava: Learning united visual representation by alignment before projection. *arXiv preprint arXiv:2311.10122*.

Man Luo, Xin Xu, Zhuyun Dai, Panupong Pasupat, Mehran Kazemi, Chitta Baral, Vaiva Imbrasaite, and Vincent Zhao. 2024. Dr.icl: Demonstration-retrieved in-context learning. *DATA INTELLIGENCE*, 6(4):909–922.

Zi-Ao Ma, Tian Lan, Rong-Cheng Tu, Yong Hu, Yu-Shi Zhu, Tong Zhang, Heyan Huang, Zhijing Wu, and Xian-Ling Mao. 2024. Multi-modal retrieval augmented multi-modal generation: Datasets, evaluation metrics and strong baselines. *arXiv preprint arXiv:2411.16365*.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. 2019. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pages 3195–3204.

Thomas Mensink, Jasper Uijlings, Lluis Castrejon, Arushi Goel, Felipe Cadar, Howard Zhou, Fei Sha, André Araujo, and Vittorio Ferrari. 2023. Encyclopedic vqa: Visual questions about detailed properties of fine-grained categories. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3113–3124.

Dustin Schwenk, Apoorv Khandelwal, Christopher Clark, Kenneth Marino, and Roozbeh Mottaghi. 2022. A-okvqa: A benchmark for visual question answering using world knowledge. In *Computer Vision – ECCV 2022*, pages 146–162, Cham. Springer Nature Switzerland.

Kurt Shuster, Jing Xu, Mojtaba Komeili, Da Ju, Eric Michael Smith, Stephen Roller, Megan Ung, Moya Chen, Kushal Arora, Joshua Lane, and 1 others. 2022. Blenderbot 3: a deployed conversational agent that continually learns to responsibly engage. *arXiv preprint arXiv:2208.03188*.

Mitchell Stern, Noam Shazeer, and Jakob Uszkoreit. 2018. Blockwise parallel decoding for deep autoregressive models. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Qwen Team. 2025. Qwen2.5-vl.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, and 1 others. 2025. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*.

Grant Van Horn, Elijah Cole, Sara Beery, Kimberly Wilber, Serge Belongie, and Oisin Mac Aodha. 2021. Benchmarking representation learning for natural world image collections. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12884–12893.

Zilong Wang, Zifeng Wang, Long Le, Huaixiu Steven Zheng, Swaroop Mishra, Vincent Perot, Yuwei Zhang, Anush Mattapalli, Ankur Taly, Jingbo Shang, and 1 others. 2024. Speculative rag: Enhancing retrieval augmented generation through drafting. *arXiv preprint arXiv:2407.08223*.

Tobias Weyand, Andre Araujo, Bingyi Cao, and Jack Sim. 2020. Google landmarks dataset v2-a large-scale benchmark for instance-level recognition and retrieval. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2575–2584.

Zaifu Zhan, Jun Wang, Shuang Zhou, Jiawen Deng, and Rui Zhang. 2025. Mmrag: multi-mode retrieval-augmented generation with large language models for biomedical in-context learning. *Journal of the American Medical Informatics Association*, page ocaf128.

Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. 2023. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*.

A Appendix

A.1 Details of Baseline Models.

All models evaluated in this work are publicly accessible and obtained from official open sources, including *Qwen2.5-VL-Instruct 3B* [1], *Qwen2.5-VL-Instruct 7B* [2], *Qwen2.5-VL-Instruct 32B* [3], *Qwen3-VL-Instruct 4B* [4], *Qwen3-VL-Instruct 8B* [5], *Qwen3-*

[1] https://huggingface.co/Qwen/Qwen2.5-VL-3B-Instruct
[2] https://huggingface.co/Qwen/Qwen2.5-VL-7B-Instruct
[3] https://huggingface.co/Qwen/Qwen2.5-VL-32B-Instruct
[4] https://huggingface.co/Qwen/Qwen3-VL-4B-Instruct
[5] https://huggingface.co/Qwen/Qwen3-VL-8B-Instruct

VL-Instruct 32B [6], *LLaVA-NeXT-Video 34B* [7] and *InternVL3 38B* [8].

A.2 Analysis of Retrieval Quality

To assess the quality of the retrieval module, we report Recall@k ($k = 1, 2, 3$) on both datasets. As shown in Table 4, retrieval performance on VideoSimpleQA is strong (Recall@3 = 0.96), which aligns well with the high downstream accuracy of 91.30%. In contrast, retrieval on EncVQA is more challenging (Recall@3 = 0.52), corresponding to a lower top accuracy of 47.64%. These results indicate that retrieval quality is a key factor affecting overall task performance, and improving retrieval on more difficult datasets remains an important direction for future work.

Metric	VideoSimpleQA	EncVQA
Recall@1	0.88	0.38
Recall@2	0.92	0.45
Recall@3	0.96	0.52

Table 4: Retrieval quality measured by Recall@k on VideoSimpleQA and EncVQA.

A.3 Impact of Entity Alignment Models

To examine the sensitivity of the entity alignment mechanism to the choice of vision-language embedding model, we compare several CLIP variants, including *clip-vit-large-patch14-336* [9], *clip-vit-large-patch14* [10] and *clip-vit-base-patch32* [11], as well as SigLIP2 (*siglip2-base-patch16-224* [12]), a cross-modal entity linking model that enhances CLIP embeddings with detection and grounding information. All experiments use Qwen-2.5VL-Instruct-3B+7B as the VideoSpeculateRAG backbone. As shown in Table 5, *clip-vit-large-patch14-336* achieves the best accuracy, while other variants and SigLIP2 yield comparable results, confirming that our entity alignment mechanism is not overly dependent on a specific CLIP version and remains effective across different vision-language embedding models.

[6] https://huggingface.co/Qwen/Qwen3-VL-32B-Instruct

[7] https://huggingface.co/llava-hf/LLaVA-NeXT-Video-34B-hf

[8] https://huggingface.co/OpenGVLab/InternVL3-38B

[9] https://huggingface.co/openai/clip-vit-large-patch14-336

[10] https://huggingface.co/openai/clip-vit-large-patch14

[11] https://huggingface.co/openai/clip-vit-base-patch32

[12] https://huggingface.co/google/siglip2-base-patch16-224

Model Name	Emb. Dim.	VideoSimpleQA
clip-vit-large-patch14-336	768	82.60
clip-vit-large-patch14	768	81.59
clip-vit-base-patch32	512	79.71
siglip2-base-patch16-224	768	81.19

Table 5: Comparison of different vision-language embedding models for entity alignment on VideoSimpleQA.

A.4 Statistics of VideoSimpleQA

Table 6 presents an overview of the VideoSimpleQA dataset, including the number of questions, videos, average video duration, and reference documents.

# questions	# videos	Average Video Length(s)	# documents
203	102	22.27	156

Table 6: Statistics of VideoSimpleQA.

A.5 LLM Usage

We only use the LLMs for correcting the grammar and improving the phrasing during the writing process.