Q-Focus: Let the Question Guide What to See in Long Videos

Junbo Qiao[1,3] ⋆*, **Xinning Chai**[2,3] ⋆, **Dong Fang**[3] †, **Sifa Xie**[3] †, **Wenxuan Huang**[1]
[1]East China Normal University [2]Shanghai Jiaotong University [3]LIGHTSPEED
georgelucien667@gmail.com, chaixinning@sjtu.edu.cn
df572@outlook.com, xie.sifaxie@gmail.com

Abstract

Video Large Language Models face severe computational and memory bottlenecks in long-video understanding. Existing visual token compression methods predominantly rely on static, query-agnostic redundancy elimination. Consequently, they lack the ability to dynamically preserve query-relevant details, failing to emulate the human "coarse-to-fine" cognitive process. To bridge this gap, we propose **Q-Focus**, a plug-and-play, two-stage visual token compression framework. Q-Focus first constructs a holistic semantic representation via a global compression stage. This is followed by a question-guided focusing stage driven by two novel components: the Adaptive Contrastive Module (ACM) and the Question Context Module (QCM). Specifically, ACM achieves adaptive query-visual alignment by treating the user queries as positive samples and dynamically sampling negative samples from the video's feature distribution. Complementarily, QCM leverages visual affinity matrices to diffuse semantic relevance from highly matched visual anchors to surrounding contextual regions, effectively capturing implicit yet critical details. Extensive experiments demonstrate that Q-Focus seamlessly integrates with diverse compression baselines. Remarkably, while retaining only 10% of the original visual tokens, it achieves relative accuracy gains of 7.3% and 6.6% on LLaVA-OneVision and LLaVA-Video, respectively, and consistently boosts the performance across other state-of-the-art models such as Qwen3-VL-8B-Instruct.

1 Introduction

Video large language models (VLLMs) [1, 2, 3, 4, 5] have recently demonstrated significant potential in video understanding tasks. However, the inherent nature of videos—comprising sequences of consecutive frames—yields a massive number of visual tokens. This imposes substantial computational and memory burdens on the model [6]. In particular, for long-video understanding, the computational complexity of the attention mechanism scales quadratically with the sequence length, which severely limits practical applicability and deployment [7]. To address this issue, visual token compression [8, 9, 10] has been extensively studied and has proven effective. These methods exploit spatial and spatiotemporal redundancy in visual data by pruning or merging less informative tokens, thereby shortening the sequence length to improve inference efficiency.

While existing methods [11, 12, 13, 14] have made significant advancements, they still suffer from fundamental limitations in their design paradigm. As illustrated in Fig. 1(a), human cognition during video understanding typically follows a coarse-to-fine process comprising two stages: (1) forming a holistic perception of the video content and (2) selectively attending to question-relevant details. In contrast, existing visual token compression methods (Fig. 1(b)) predominantly rely on static importance scoring or diversity-aware sampling, primarily aiming to eliminate redundancy while

*indicates equal contribution, † indicates the corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

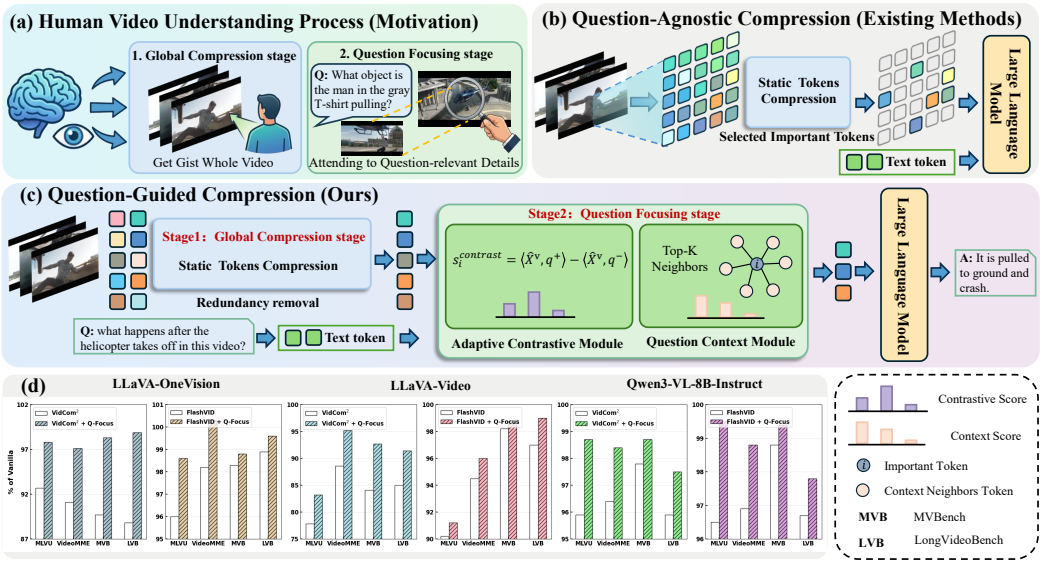

Figure 1: Motivation and Overview of the proposed Q-Focus framework. (a) Human video understanding follows a coarse-to-fine process: global perception followed by selective attention to question-relevant details. (b) Existing methods rely on static, question-agnostic compression. (c) Q-Focus mimics this cognition via a two-stage pipeline: Stage 1 removes global redundancy, while Stage 2 employs an Adaptive Contrastive Module (ACM) and Question Context Module (QCM) to dynamically prioritize and propagate query semantics. (d) Integrating Q-Focus yields consistent accuracy gains across benchmarks and baseline compressors.

preserving global video semantics. However, these query-agnostic mechanisms cannot dynamically adjust visual focus based on query semantics, leading to insufficient specificity when addressing specific questions or inadequate allocation of resources to query-relevant tokens under high compression ratios. In essence, existing approaches emulate only the global compression stage, failing to capture the subsequent question focusing stage.

To bridge this critical gap, we propose **Q-Focus**, a plug-and-play two-stage compression framework to sequentially decompose the compression process, as show in Fig. 1(c). First, during the **Global Compression stage**, we leverage established approaches (e.g., VidCom2 [15], FlashVid [16]) to perform video-level redundancy reduction, constructing a holistic semantic representation of the input. Second, in the **Question Focusing stage**, building upon the compressed result from the first stage, we propose two complementary key components: the Adaptive Contrastive Module (ACM) and the Question Context Module (QCM), to inject question-guided semantics. This enables the token sequence to prioritize question-relevant visual tokens while preserving holistic understanding, thereby yielding more precise, context-aware responses.

Specifically, within the ACM, we employ the query text as positive samples and construct negative samples from the video's intrinsic semantic distribution, achieving video-level adaptive alignment. Moreover, we observe that queries often serve merely as semantic anchors that may not explicitly contain the answer. Typically, the true answer resides implicitly in the context around the anchor (e.g., the query "What object is the man in the gray T-shirt pulling" in Fig. 1(a)). To address this, we propose QCM, which leverages the affinity matrix between visual features as a bridge for semantic diffusion. This module propagates semantic scores from highly-matched tokens to their visual context tokens, effectively expanding the attentional scope from anchors to semantically related regions. As illustrated in Fig. 1(d), Q-Focus consistently delivers significant performance gains over existing methods across various settings. Notably, while retaining only 10% of visual tokens and employing VidCom^2 [15] as the stage-1 method, Q-Focus achieves relative accuracy gains of 7.3% and 6.6% on LLaVA-OneVision [1] and LLaVA-Video [5], respectively. Furthermore, when integrated into Qwen3-VL-8B-Instruct [2], Q-Focus seamlessly adapts to various compression strategies and yields significant performance improvements. To summarize, our main contributions are four-fold:

- We propose Q-Focus, a plug-and-play two-stage framework that emulates human coarse-to-fine cognition. It decouples compression process into global compression stage and question focusing stage.
- We propose an Adaptive Contrastive Module (ACM) for dynamic query-visual alignment. The module treats the query as a positive sample and draws negative sample from the video's feature distribution.
- We propose a Question Context Module (QCM), leveraging visual affinity matrices to diffuse semantic relevance from anchors to semantically related contextual regions.
- Extensive experiments show that Q-Focus adapts to diverse compression methods, consistently boosting performance across various baselines and retention ratios.

2 Related Work

2.1 Video Large Language Models

Rapid advances in Multimodal Large Language Models (MLLMs) [17, 18, 19, 20, 21, 22] have spurred interest in Video Large Language Models (Video LLMs). Typically, a Video LLM [23, 24, 1, 25, 26] encodes video frames into visual tokens via a visual encoder and projection module, then feeds them with textual instructions into an LLM for response generation. These models enhance video understanding through extensive datasets and targeted training. For instance, Video-LLaVA [25] employs a shared projection layer to obtain a unified visual representation across modalities, while LLaVA-OneVision [1] dproposes a unified model capable of handling images, videos, audios, and other modalities simultaneously. Similarly, LLaVA-Video [5] constructs a high-quality synthetic dataset tailored for video instruction-following. Among these, LLaVA-OneVision and the Qwen-VL [2] series have achieved widespread adoption due to their superior adaptability and performance. However, the extensive sequence of visual tokens induced by continuous video frames poses a critical bottleneck, substantially limiting the computational efficiency and scalability of these models.

2.2 Token Compression for VideoLLMs

To mitigate the computational bottleneck caused by extensive visual tokens, various compression strategies have been proposed. The most common approaches [27, 14, 28, 7] focus on selecting a subset of informative tokens from the input sequence. For instance, FastV [8] selects critical tokens during the pre-filling stage using attention maps. However, these methods are primarily designed for static images and largely overlook temporal redundancy across video frames. To better exploit spatiotemporal structures, video-specific methods such as PruneVID [29] and FastVID [12] segment videos based on scene boundaries and aggregate tokens within each segment. VisionZip [9] reduces spatial redundancy through attention-driven feature analysis within the visual encoder. Similarly, DyCoke [30] partitions videos into multiple segments and performs intra-segment compression via fixed even–odd token grouping. VidCom^2 [15] adjusts the budget for each frame by calculating the average similarity of both per-frame and global features. FlashVID [16] introduces tree-based spatiotemporal token merging, while similarly, V-CAST [31] proposes a training-free curvature-aware spatio-temporal pruning method to eliminate spatiotemporal redundancy. Despite these advancements, existing methods remain query-agnostic, often discarding critical query-relevant tokens under high compression and degrading precision. To address this, we propose Q-Focus, which can be seamlessly coupled with existing methods to inject question-guided semantics into the token compression process, thereby adaptively preserving task-relevant visual tokens.

2.3 Query-Guided Token Selection

In recent years, several image understanding works have incorporated textual prior into token selection to enhance query awareness. For instance, SparseVLM [10] employs attention-based cross-modal relevance estimation to preserve query-related regions. Other methods leverage gradient-based saliency [32, 33, 34] or CLIP-based similarity [35, 36, 37] to integrate language guidance. FlashVLM [38] introduces an explicit similarity formula to compute similarity between visual tokens and text embeddings, whereas VisionTrim [39] leverages textual information to guide the clustering and merging of visual tokens. However, extending question-guided token compression to video is

non-trivial: video produces a far larger number of post-encoder visual tokens, making query-guided computation—such as expensive text–vision cross-attention [40]—prohibitively costly at the token level. Consequently, existing video compression methods resort to coarser alternatives such as keyframe selection [41, 42, 43] or streaming text-guided understanding [44], which inherit question guidance only at the frame level and fall short of fine-grained token-level question awareness. To overcome this bottleneck, we propose a two-stage question-guided paradigm that first applies a lightweight Stage-1 compressor to drastically reduce the token count, then injects question guidance through an Adaptive Contrastive Module (ACM) and a Question Context Module (QCM) on the compressed token space—remaining training-free, introducing no learnable parameters, and flexibly attachable to different base VideoLLMs and Stage-1 compressors.

3 Method

3.1 Preliminaries

VideoLLM Architecture. The prevailing design for current VideoLLMs adheres to the "ViT-MLP-LLM" framework [1, 5]. Taking LLaVA-Video as a representative instance, the process begins by feeding a video sequence $\mathbf{V} = \{\mathbf{v}_t\}_{t=1}^{T} \in \mathbb{R}^{T\times H\times W\times 3}$ into a Vision Transformer (ViT). This yields a sequence of embeddings $\mathbf{Z} = \{\mathbf{z}_t\}_{t=1}^{T} \in \mathbb{R}^{T\times N\times D}$. Subsequently, these embeddings are projected through a two-layer MLP to generate visual tokens $\mathbf{X}^v = \{\mathbf{x}_t^v\}_{t=1}^{T} \in \mathbb{R}^{T\times M\times D'}$, where the sequence length is compressed ($M < N$). These tokens are then fed into LLM:

$$p(\mathbf{Y} \mid \mathbf{X}^v, \mathbf{X}^t) = \prod_{i=1}^{L} p(\mathbf{y}_i \mid \mathbf{X}^v, \mathbf{X}^t, \mathbf{Y}_{1:i-1}), \tag{1}$$

where $\mathbf{Y} = \{\mathbf{y}_i\}_{i=1}^{L}$ denotes the generated response tokens, conditioned on the visual tokens $\mathbf{X}^v$ and textual tokens $\mathbf{X}^t$.

Token Compression for VideoLLMs. The primary objective of token compression is to enhance inference efficiency by directly condensing token representations to minimize redundancy. In the context of VideoLLMs, this procedure generally entails transforming the original visual token sequences $\mathbf{X}^v$ into a more compact representation $\hat{\mathbf{X}}^v$:

$$\hat{\mathbf{X}}^v = \Phi(\mathbf{X}^v), \quad \text{s.t.} \quad |\hat{\mathbf{X}}^v| < |\mathbf{X}^v| \tag{2}$$

where Φ denotes the token compression operator, and $|\cdot|$ signifies the number of tokens.

Visual token compression is particularly critical for Video LLMs, as they process significantly more visual tokens than standard image-based VLMs. Due to the consecutive frames of video often exhibit high similarity, resulting in substantial visual redundancy. Existing token compression algorithms [9, 30, 12, 15, 16] typically design the compression function Φ by balancing representativeness and diversity. However, in these methods, the question information fails to actively participate in the token selection process, leading to a question-agnostic compression paradigm. Consequently, such strategies often discard question-critical details, particularly under high compression ratios.

3.2 Two-Stage Framework

Inspired by the coarse-to-fine cognitive prior discussed in Sec. 1, we note that the compression (Eq. 2) in existing methods operates independently of the textual query X^t. To address this, we propose Q-Focus, a two-stage compression framework that explicitly incorporates question guidance.

As illustrated in Fig. 2, Q-Focus decouples the compression pipeline into two sequential stages. Stage 1 focuses on eliminating global redundancy, leveraging established methods to construct a compact and holistic representation. This process aligns with the formulation in Eq. 2. Subsequently, Stage 2 refines this representation by incorporating textual query information. Specifically, this stage employs an **Adaptive Contrastive Module** (detailed in Sec. 3.3) to align visual tokens with query semantics, and utilizes a **Question Context Module** (detailed in Sec. 3.4) to diffuse score weights to relevant contextual regions. This design ensures that the final token sequence fed into the Large Language Model is not only computationally efficient but also highly relevant to the question semantics. The entire second stage can be formulated as:

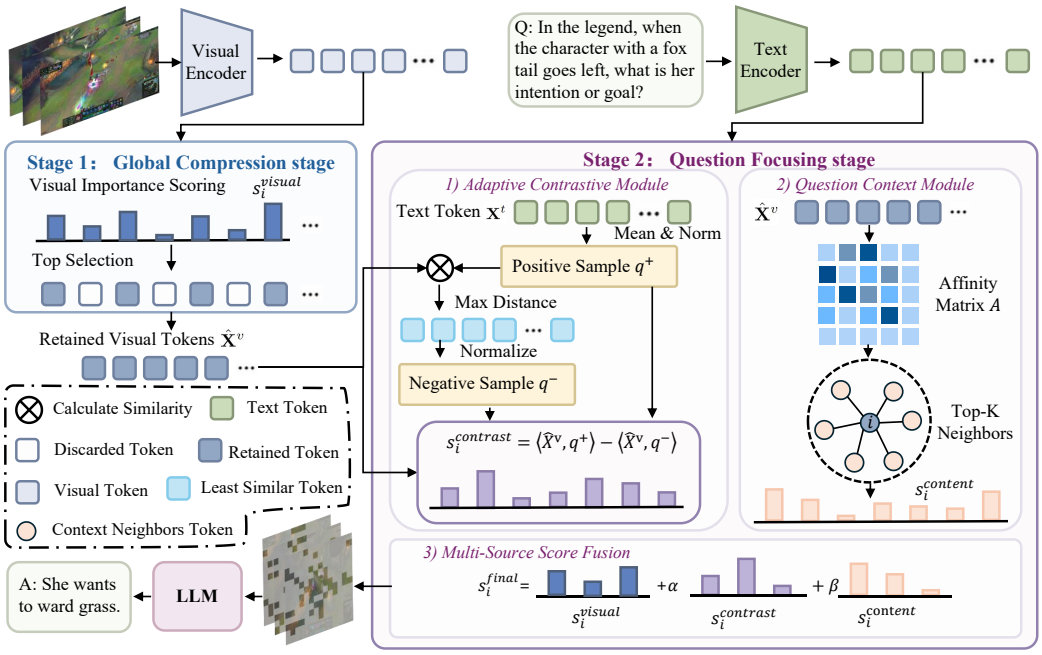

Figure 2: llustration of the proposed Q-Focus. Our framework decouples token compression into two sequential stages: (1) Global Compression, which removes video-level redundancy to obtain a compact holistic representation; and (2) Question Focusing, dynamically refining the token sequence by aligning visual features with query semantics and propagating relevance to contextual regions.

$$\hat{\mathbf{X}}^v_{\text{final}} = \Psi(\hat{\mathbf{X}}^v, \mathbf{X}^t), \quad \text{s.t.} \quad |\hat{\mathbf{X}}^v_{\text{final}}| \leq |\hat{\mathbf{X}}^v| \tag{3}$$

Where Ψ represents the Stage 2 token compression operator. Notably, Q-Focus is a plug-and-play framework where the first stage can be implemented with any off-the-shelf compression methods.

3.3 Adaptive Contrastive Module

To effectively align visual tokens with question semantics in Stage 2, we propose an **Adaptive Contrastive Module (ACM)**. Unlike traditional contrastive learning that relies on extensive training or external negative samples, ACM adopts a training-free, self-negative strategy that dynamically constructs query-aware compression criteria directly from the video content itself.

Specifically, given the textual query features $\mathbf{X}^t \in \mathbb{R}^{L\times d}$, we first obtain the positive sample $\mathbf{q}^+ = \text{Mean}(\mathbf{X}^t)/\|\text{Mean}(\mathbf{X}^t)\|_2$. Some image query-guided methods [38, 10] would directly use the similarity to the positive sample (i.e., $\text{Score} = \text{Sim}(V, \mathbf{q}^+)$) for token selection. However, the query $\mathbf{q}^+$ inherently contains both semantic signal (T_{Import}) and semantic noise (T_{Noise}), such as ambiguous phrasing or question-irrelevant keywords. Directly computing similarity without accounting for this noise can be formulated as:

$$\text{Score} = \text{Sim}(V, T_{\text{Import}}) + \text{Sim}(V, T_{\text{Noise}}). \tag{2}$$

To mitigate this interference, we dynamically construct a negative sample $\mathbf{q}^-$ directly from the visual content. By subtracting the similarity to $\mathbf{q}^-$, we effectively cancel out the noise component T_{Noise}:

$$\text{Score} = \text{Sim}(V, T_{\text{Import}}) + \text{Sim}(V, T_{\text{Noise}}) - \text{Sim}(V, \mathbf{q}^-). \tag{3}$$

Specifically, we first compute the cosine similarity between each visual token $\mathbf{v}_i \in \hat{\mathbf{X}}^v$ and the positive sample $\mathbf{q}^+$:

$$s_i^+ = \langle \mathbf{v}_i, \mathbf{q}^+ \rangle. \tag{4}$$

These similarities are then normalized via Z-score standardization:

$$\tilde{s}_i = \frac{s_i^+ - \mu_s}{\sigma_s + \epsilon}, \tag{5}$$

where μ_s and σ_s denote the mean and standard deviation of $\{s_i^+\}$, respectively.

Crucially, the negative sample $\mathbf{q}^-$ is constructed exclusively from the token candidates retained by Stage 1. This design prevents interference from tokens already deemed redundant in the global compression phase. Thus, the final contrastive score for each token is computed as:

$$s_i^{\text{contrast}} = \langle \hat{\mathbf{X}}^v, \mathbf{q}^+ \rangle - \langle \hat{\mathbf{X}}^v, \mathbf{q}^- \rangle. \tag{6}$$

This subtraction mechanism enhances the discriminability of query-relevant tokens by suppressing responses to semantic noise.

3.4 Question Context Module

While the Adaptive Contrastive Module effectively identifies tokens directly aligned with the query, it may overlook contextual cues that are semantically related but not explicitly mentioned in the query text. For instance, when asked "what happens after the helicopter takes off?", the query anchors on "helicopter", but the answer may reside in surrounding regions (e.g., the runway, nearby buildings). To address this, we propose a **Question Context Module (QCM)** that diffuses attention from high-matching tokens to their visual neighbors.

Specifically, given the visual features $\hat{\mathbf{X}}^v$ from Stage 1 and the contrastive scores $\mathbf{s}^{\text{contrast}} \in \mathbb{R}^N$ from ACM, we first construct a visual affinity graph that captures semantic relationships between vision tokens. For each token $\mathbf{v}_i$, we compute its affinity to all other tokens:

$$a_{ij} = \langle \mathbf{v}_i, \mathbf{v}_j \rangle, \tag{7}$$

where a_{ij} represents the semantic similarity between token i and token j. Higher affinity indicates that two tokens belong to the same visual context. To efficiently propagate scores without computing dense graph connections, for each token $\mathbf{v}_i$, we select its top-k most similar neighbors:

$$\mathcal{N}(i) = \text{TopK}(\{a_{ij}\}_{j=1}^N, k), \tag{8}$$

where k is a hyperparameter controlling the neighborhood size. This sparse connectivity reduces computational overhead while preserving the most relevant contextual relationships. The contrastive scores from neighbors are then aggregated derived from their affinity values. Specifically, we compute softmax-normalized weights over the top-k neighbors:

$$\alpha_{ij} = \frac{\exp(a_{ij}/\sigma)}{\sum_{l \in \mathcal{N}(i)} \exp(a_{il}/\sigma)}, \tag{9}$$

where σ is a temperature parameter controlling the sharpness of attention distribution. The content score for token i is obtained by weighted aggregation of its neighbors' contrastive scores:

$$s_i^{\text{content}} = \sum_{j \in \mathcal{N}(i)} \alpha_{ij} \cdot s_j^{\text{contrast}}. \tag{10}$$

3.5 Multi-Source Score Fusion.

The final token selection score integrates three complementary signals: (1) the global visual importance from Stage 1, (2) the query-aligned contrastive score from ACM, and (3) the contextually smoothed score from QCM. Formally, for each token $\mathbf{v}_i$, we compute:

$$s_i^{\text{final}} = \underbrace{\cdot s_i^{\text{visual}}}_{\text{Stage 1}} + \underbrace{\alpha \cdot s_i^{\text{contrast}}}_{\text{ACM}} + \underbrace{\beta \cdot s_i^{\text{content}}}_{\text{QCM}}, \tag{11}$$

where s_i^{visual} denotes the visual importance score from Stage 1. The weights α and β balance the contributions of three different scores.

4 Experiments

4.1 Experimental Setting

Benchmark. We conduct extensive comparative experiments across four established benchmarks: VideoMME [45], LongVideoBench [46], MVBench [47], and MLVU [48]. All evaluations are

Table 1: **Performance of Q-Focus integrated into state-of-the-art methods on LLaVA-OneVision and LLaVA-Video.** "Avg." shows the mean performance across different benchmarks. Δ represents the percentage improvement compared to the baseline.

Method	Retention Ratio R	VideoMME				MLVU	LongVideo Bench	MVBench	Avg.		
		Short	Medium	Long	Overall				Score	Rel. Acc (%)	Δ
LLaVA-OneVision											
Vanilla	100%	69.9	56.7	48.9	58.5	63.0	56.6	58.3	59.1	100.0	
VidCom^2	25%	69.8	56.4	49.4	58.6	62.5	54.9	57.2	58.3	98.6	
+ Q-Focus	25%	**71.7**	**56.6**	**50.1**	**59.4**	**63.1**	**59.0**	**57.8**	**59.8**	**101.2**	**+2.6**
FlashVID	25%	71.2	56.4	49.1	58.9	62.7	56.8	58.4	59.2	100.2	
+ Q-Focus	25%	**71.9**	**56.0**	**49.9**	**59.3**	**63.2**	**57.7**	**58.7**	**59.7**	**101.0**	**+0.8**
VidCom^2	20%	69.7	56.0	48.0	57.9	62.1	53.9	56.5	57.6	97.5	
+ Q-Focus	20%	**71.3**	**56.1**	**49.2**	**58.9**	**63.0**	**56.6**	**57.6**	**59.0**	**99.8**	**+2.3**
FlashVID	20%	70.2	55.1	48.7	58.0	62.5	56.7	58.2	58.9	99.7	
+ Q-Focus	20%	**71.7**	**55.3**	**49.2**	**58.7**	**62.9**	**57.7**	**58.5**	**59.5**	**100.7**	**+1.0**
VidCom^2	15%	65.8	54.8	48.1	56.2	60.9	53.3	54.4	56.2	95.1	
+ Q-Focus	15%	**69.7**	**55.9**	**48.3**	**58.0**	**62.4**	**56.0**	**57.9**	**58.6**	**99.2**	**+4.1**
FlashVID	15%	70.5	56.0	49.5	58.7	61.8	57.4	57.8	58.9	99.7	
+ Q-Focus	15%	**70.7**	**56.2**	**49.9**	**58.9**	**62.7**	**58.3**	**57.9**	**59.5**	**100.7**	**+1.0**
VidCom^2	10%	61.8	51.4	46.7	53.3	58.4	50.3	52.3	53.6	90.7	
+ Q-Focus	10%	**67.4**	**56.3**	**46.7**	**56.8**	**61.6**	**56.0**	**57.3**	**57.9**	**98.0**	**+7.3**
FlashVID	10%	68.3	55.8	48.5	57.5	60.5	56.0	57.3	57.8	97.8	
+ Q-Focus	10%	**70.6**	**56.4**	**49.2**	**58.7**	**62.1**	**56.4**	**57.6**	**58.7**	**99.3**	**+1.5**
LLaVA-Video											
Vanilla	100%	77.0	62.1	53.3	64.2	70.3	59.5	60.4	63.6	100.0	
VidCom^2	20%	71.9	59.6	51.6	61.0	57.5	54.8	55.4	57.2	89.9	
+ Q-Focus	20%	**73.8**	**60.4**	**52.4**	**62.2**	**59.1**	**56.2**	**57.8**	**58.8**	**92.5**	**+2.6**
FlashVID	20%	74.3	60.1	51.9	62.1	64.3	58.7	59.8	61.2	96.2	
+ Q-Focus	20%	**74.7**	**61.4**	**52.2**	**62.7**	**65.5**	**59.4**	**60.7**	**62.1**	**97.6**	**+1.4**
VidCom^2	10%	64.6	57.1	49.1	56.9	54.7	50.6	50.8	53.3	83.8	
+ Q-Focus	10%	**72.6**	**60.5**	**50.3**	**61.1**	**58.5**	**54.4**	**56.0**	**57.5**	**90.4**	**+6.6**
FlashVID	10%	72.1	58.9	51.0	60.7	63.4	57.7	59.3	60.3	94.8	
+ Q-Focus	10%	**73.5**	**60.4**	**52.6**	**62.2**	**64.7**	**59.3**	**60.0**	**61.6**	**96.9**	**+2.1**

Table 2: **Performance of Q-Focus integrated into state-of-the-art methods on Qwen3-VL-8B-Instruct across benchmarks under multiple settings.** Δ represents the percentage improvement compared to the baseline.

Method	Retention Ratio R	VideoMME				MLVU	LongVideo Bench	MVBench	Avg.		
		Short	Medium	Long	Overall				Score	Rel. Acc (%)	Δ
Qwen3-VL-8B-Instruct											
Vanilla	100%	76.0	60.4	57.0	64.5	63.5	60.3	68.6	64.2	100.0	
VidCom^2	25%	72.0	58.7	55.9	62.2	60.9	57.8	67.1	62.0	96.6	
+ Q-Focus	25%	**74.3**	**59.3**	**56.8**	**63.5**	**62.7**	**58.8**	**67.7**	**63.2**	**98.4**	**+1.8**
FlashVID	25%	74.6	58.9	54.0	62.5	61.3	58.3	67.8	62.5	97.4	
+ Q-Focus	25%	**75.1**	**59.8**	**56.2**	**63.7**	**63.2**	**59.0**	**68.2**	**63.5**	**98.9**	**+1.5**
VidCom^2	15%	68.9	56.7	53.3	59.6	60.0	56.0	64.3	60.0	93.5	
+ Q-Focus	15%	**71.8**	**57.5**	**54.5**	**61.3**	**60.6**	**56.9**	**65.8**	**61.2**	**95.3**	**+1.8**
FlashVID	15%	72.7	58.2	54.7	61.9	60.3	57.4	66.2	61.5	95.8	
+ Q-Focus	15%	**73.1**	**58.8**	**56.3**	**62.7**	**61.5**	**58.2**	**66.5**	**62.2**	**96.9**	**+1.1**

performed using the standardized LMMs-Eval framework [49]. Crucially, these benchmarks span a diverse range of video durations and complex reasoning scenarios, facilitating a rigorous assessment of our method's effectiveness and generalization capability. Further implementation details and benchmark statistics are provided in the Appendix A.2.

Implementations. We evaluate our method on four widely-adopted Video LLMs: LLaVA-OneVision (LLaVA-OV) [1], LLaVA-Video [5] and Qwen3-VL [50]. More results are provided in Appendix A.1.

Table 3: Efficiency and performance comparison across different methods.

Method	Retention Ratio R	GPU Peak Memory (GB)	Prefilling Time (ms)	VideoMME	MLVU	LongVideo Bench	MVBenchU	Avg.
Vanilla	100%	17.7	1257.5	58.5	63.0	56.6	58.3	59.1
VidCom^2	25%	16.0	163.3	58.6	62.5	54.9	57.2	58.3
+ Q-Focus	25%	17.2	176.5	59.4	63.1	59.0	57.8	59.8
FlashVID	25%	29.1	203.0	58.9	62.7	56.8	58.4	59.2
+ Q-Focus	25%	30.6	227.4	59.3	63.2	57.7	58.7	59.7

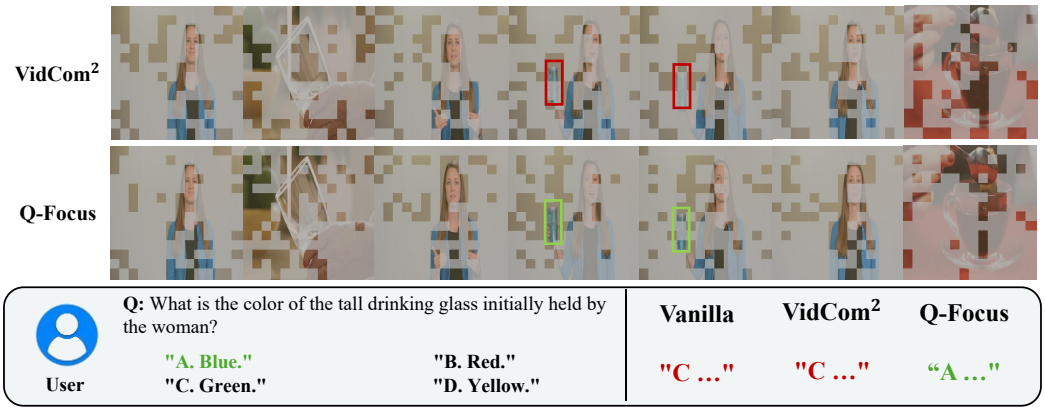

Figure 3: **Qualitative visualization of token retention.** Gray regions denote discarded tokens, while clear regions represent retained visual information. **Top:** The baseline VidCom^2 relies on static importance scoring and mistakenly prunes the critical token (red box). **Bottom:** Q-Focus successfully preserving the query-relevant token (green box) and yielding the correct answer.

Baselines. Q-focus can be coupled with existing methods. To validate generalizability, we integrated two state-of-the-art training-free Video LLM acceleration methods into the first stage: (1) VidCom^2 [15], identifying key tokens via global and per-frame average features; and (2) FlashVID [16], combining spatio-temporal token merging with attention-based diversity sampling.

4.2 Main Comparisons

Quantitative Comparisons. Tab. 1 presents the comprehensive comparison results on LLaVA-OV and LLaVA-Video. Q-Focus consistently improves performance across baselines and retention ratios. For instance, on LLaVA-OV with a 25% token retention ratio, the average score of VidCom^2 increases from 57.6 to 59.0, and FlashVID from 58.9 to 59.5. Notably, Q-Focus's advantages intensify under higher compression. At a 10% retention ratio, Q-Focus substantially boosts VidCom^2's performance on LLaVA-OV from 53.6 to 57.9, and on LLaVA-Video from 53.3 to 57.5, demonstrating significant relative accuracy gains. FlashVID also benefits consistently, albeit with smaller improvements. We attribute this to FlashVID's stronger baseline performance, which leaves less room for further enhancement as it approaches the performance ceiling. These results demonstrate that Q-Focus demonstrates strong compatibility with diverse compression strategies and delivers consistent gains, especially under tight token budgets where precise visual attention allocation is critical.

Similar performance trends are observed on Qwen3-VL in Tab. 2. Q-Focus consistently elevates performance across both compression backbones. Notably, at 25% retention, FlashVID + Q-Focus achieves an average score of 63.5, nearly matching the vanilla model's 64.2 despite using only a quarter of the tokens. Even at 15% retention, measurable gains persist across all benchmarks. These results corroborate the architecture-agnostic effectiveness of our approach, confirming that query-guided attention enhances diverse VLLM families under constrained token budgets.

Qualitative comparisons. Fig. 3 visualizes token retention strategies of the baseline VidCom^2 and Q-Focus. The gray mosaic regions denote compressed tokens, while clear regions represent the visual information retained for the LLM. As illustrated in the top row, the baseline VidCom^2, relying on static importance scoring, overlooks the semantic relevance of the glass to the query. Consequently, it mistakenly prunes this critical object (red box), leading to an incorrect prediction. In contrast,

Table 4: **Ablation on main components.** Progressively adding ACM and QCM improves performance across benchmarks.

Method	VideoMME	MLVU	LongVideo Bench	MVBench	Rel. Acc.
Vanilla	58.5	63.0	56.6	58.3	100
+ VidCom^2	58.6	62.5	54.9	57.2	98.6
+ two-stage	58.4	62.3	54.7	57.4	98.5
+ ACM	58.8	62.9	58.2	57.5	100.4
+ QCM	**59.4**	**63.1**	**59.0**	**57.8**	**101.2**

Table 5: **Ablation on Stage-1 retention ratio** R_1. $R_2 = 25\%$

R_1-R_2	VideoMME	MLVU	Prefilling Time (ms)
5%	59.4	63.1	176.5
10%	59.3	63.2	198.5
15%	59.5	63.1	237.2

Table 6: **Ablation on negative sample construction.**

Method	VideoMME	MLVU	LongVideo Bench	MVBench	Rel. Acc.
Vanilla	58.5	63.0	56.6	58.3	100
Empty Neg	58.9	63.0	58.3	57.1	100.3
Fixed Neg	58.7	62.8	58.0	57.3	100.1
Adaptive Neg	**59.4**	**63.1**	**59.0**	**57.8**	**101.2**

Table 7: **Ablation on neighborhood size** k**.**

k	VideoMME	MLVU	Prefilling Time (ms)	GPU Peak Memory (GB)
4	59.0	62.7	169.2	16.9
8	59.4	63.1	176.5	17.2
12	59.3	63.0	194.3	20.1

the bottom row demonstrates that Q-Focus effectively leverages question guidance to identify and preserve the visual tokens (green box).This retains the necessary visual evidence, resulting in the correct answer. More qualitative comparisons are provided in Appendix A.4.

4.3 Efficiency Analysis

Tab. 3 compares efficiency and performance across methods. Relative to vanilla, VidCom^2 and FlashVID reduce prefilling time about 6-8 times (from 1257.5 ms to 163 ms and 203 ms, respectively) and lower GPU memory. Notably, integrating Q-Focus adds minimal overhead. For example, at a 25% retention radio, it increases the prefilling time by only 13.2 ms and the peak memory by 1.2 GB over VidCom^2. Importantly, this slight cost yields clear gains (e.g., VidCom^2 + Q-Focus improves the average score by 1.5 points). Thus Q-Focus enhances query-aware token selection without sacrificing compression efficiency. This makes it well-suited for resource-constrained video understanding.

4.4 Ablation Study and Analysis

Ablation study on Main Component. We evaluate the contribution of each module on LLaVA-OneVision at a 25% token retention ratio. As shown in Tab. 4, simply dividing the VidCom^2 process into two stages does not improve performance. However, adding the Adaptive Contrastive Module (ACM) significantly surpasses the vanilla baseline (+ACM: 100.4%). This confirms that query-guided alignment effectively preserves critical semantics. Finally, adding the Question Context Module (QCM) achieves the best results across all benchmarks (+QCM: 101.2%). These steady improvements prove the effectiveness of each proposed component.

Ablation study on Retention Rate. We analyze how the Stage-1 retention ratio R_1 affects performance and efficiency, while fixing the final Stage-2 ratio R_2 at 25% (Tab. 5). Increasing R_1 from 0.30 to 0.40 allows more tokens to the second stage. However, the accuracy gain is minimal (VideoMME: 59.4 to 59.5). In contrast, the prefilling time increases significantly from 176.5 ms to 237.2 ms. Stage-1 over-retention merely inflates computational cost. Setting $R_1 = 0.30$ achieves the best balance.

Ablation study on negative sample construction. Tab. 6 compares different strategies for constructing negative samples. Both empty and fixed negative samples improve performance over the vanilla model. This confirms that introducing contrastive scoring is effective. In contrast, our adaptive negative strategy achieves the highest scores across all benchmarks (101.2% relative accuracy). This is because static negatives (Empty or Fixed) struggle to adapt to varying content. As a result, they cannot effectively model query-irrelevant noise.

Ablation study on top-k neighborhood size k. We analyze the effect of the neighbor size k in Tab. 7. Increasing k from 4 to 8 improves accuracy while keeping computational cost low. However, raising k to 12 brings no further accuracy gain but significantly increases prefilling time and GPU memory usage. This shows that a small neighborhood is sufficient to capture useful contextual cues.

Larger values only add redundant calculations. We set $k = 8$ as the default setting to achieve the best balance between performance and efficiency.

Compression Sensitivity Across Task Categories. To investigate the impact of evidence locality, we conduct a breakdown analysis on MLVU across its three task categories: Holistic (topic reasoning, anomaly recognition), Single-Detail (needle QA, ego reasoning, plot QA), and Multi-Detail (action ordering, action counting). Results are shown in Tab. 8.

Table 8: **Per-category breakdown** on MLVU at 10% retention ratio.

Method	MLVU				
	Retention	Overall	Holistic	Single-Detail	Multi-Detail
Vanilla	100%	63.0	79.6	66.1	41.9
Vidcom^2	10%	58.4	70.0	62.6	40.4
+ Q-Focus	10%	61.6	74.1	66.4	42.0

At 10% retention, VidCom^2 suffers a severe degradation on Holistic tasks ($79.6 \rightarrow 70.0$, -9.6 points), as these tasks require globally distributed evidence that is most vulnerable to aggressive compression; Q-Focus recovers 4.1 points ($70.0 \rightarrow 74.1$) because, even though holistic questions do not point to a specific spatial region, the question text still provides a semantic anchor that frees the limited budget to retain more diverse visual evidence across the video. On Single-Detail tasks, Q-Focus not only recovers the compression loss but *exceeds* the uncompressed vanilla model (66.4 vs. 66.1), which is expected since these tasks focus on specific visual details that align well with the query-guided selection mechanism. Multi-Detail tasks, interestingly, are relatively insensitive to compression—VidCom^2 at 10% retains near-vanilla performance—and Q-Focus further improves and slightly exceeds vanilla (42.0 vs. 41.9). We note that while current community practice (e.g., lmms-eval) reports only a single overall score for MLVU, the per-category breakdown is informative, and we will include it in the revised paper.

Additional ablation studies. We analyze the impact of α and β in Eq. 11 to understand the relative importance of each scoring component. As shown in Tab. 9, reducing either weight to 0.5 consistently degrades performance across all benchmarks, confirming that both query alignment and contextual diffusion are indispensable. Notably, setting $\alpha = 2$ and $\beta = 1$ yields the optimal results. This demonstrates that the contrastive score from ACM serves as the primary driver for token selection, as it directly captures query-relevant semantics. Meanwhile, maintaining $\beta = 1$ ensures the QCM provides sufficient contextual support to recover implicit details without diluting the core alignment signal. Consequently, we adopt $\alpha = 2$ and $\beta = 1$ as the default configuration to achieve the best trade-off between precise question focusing and robust scene understanding.

Table 9: **Ablation on α and β in Eq 11.**

α	β	VideoMME	MLVU	LongVideoBench	MVBench
1	1	59.1	63.0	55.6	57.0
1	0.5	58.7	62.3	56.8	57.3
0.5	1	58.8	62.5	56.4	57.3
2	1	**59.4**	**63.1**	**59.0**	**57.8**
1	2	58.9	62.8	57.9	57.1

5 Conclusion

In this work, we proposed Q-Focus, a plug-and-play two-stage framework designed to address a fundamental limitation in current compression paradigms: their reliance on static, query-agnostic redundancy removal. By decoupling global compression from query-guided focusing, Q-Focus effectively emulates a human-like "coarse-to-fine" cognitive process. The Adaptive Contrastive Module (ACM) dynamically aligns visual tokens with query semantics through stage-aware negative sampling. Meanwhile, the Question Context Module (QCM) diffuses relevance scores to semantically related contextual regions. Extensive experiments demonstrate that Q-Focus integrates seamlessly with various compression baselines. It consistently improves performance across multiple benchmarks even at extreme token retention ratios.

References

[1] Li, B., Y. Zhang, D. Guo, et al. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*, 2024.

[2] Yang, A., A. Li, B. Yang, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[3] Wang, Y., X. Li, Z. Yan, et al. Internvideo2. 5: Empowering video mllms with long and rich context modeling. *arXiv preprint arXiv:2501.12386*, 2025.

[4] Zhang, H., X. Li, L. Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. In *Proceedings of the 2023 conference on empirical methods in natural language processing: system demonstrations*, pages 543–553. 2023.

[5] Zhang, Y., J. Wu, W. Li, et al. Video instruction tuning with synthetic data. *URL https://arxiv.org/abs/2410.02713*, 17, 2024.

[6] Shao, K., K. Tao, K. Zhang, et al. When tokens talk too much: A survey of multimodal long-context token compression across images, videos, and audios. *arXiv preprint arXiv:2507.20198*, 2025.

[7] Wang, Y., X. Liu, X. Gui, et al. Accelerating streaming video large language models via hierarchical token compression. *arXiv preprint arXiv:2512.00891*, 2025.

[8] Chen, L., H. Zhao, T. Liu, et al. An image is worth 1/2 tokens after layer 2: Plug-and-play inference acceleration for large vision-language models. In *European Conference on Computer Vision*, pages 19–35. Springer, 2024.

[9] Yang, S., Y. Chen, Z. Tian, et al. Visionzip: Longer is better but not necessary in vision language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19792–19802. 2025.

[10] Zhang, Y., C.-K. Fan, J. Ma, et al. Sparsevlm: Visual token sparsification for efficient vision-language model inference. *arXiv preprint arXiv:2410.04417*, 2024.

[11] Sun, B., J. Zhao, X. Wei, et al. Llava-scissor: Token compression with semantic connected components for video llms. *arXiv preprint arXiv:2506.21862*, 2025.

[12] Shen, L., G. Gong, T. He, et al. Fastvid: Dynamic density pruning for fast video large language models. *arXiv preprint arXiv:2503.11187*, 2025.

[13] Li, Y., H. Gui, Z. Fan, et al. Less is more, but where? dynamic token compression via llm-guided keyframe prior. *arXiv preprint arXiv:2512.06866*, 2025.

[14] Cho, J., J. Lee, M. Hayat, et al. Floc: Facility location-based efficient visual token compression for long video understanding. *arXiv preprint arXiv:2511.00141*, 2025.

[15] Liu, X., Y. Wang, J. Ma, et al. Video compression commander: Plug-and-play inference acceleration for video large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 1910–1924. 2025.

[16] Fan, Z., K. Chen, R. Xing, et al. Flashvid: Efficient video large language models via training-free tree-based spatiotemporal token merging. *arXiv preprint arXiv:2602.08024*, 2026.

[17] Liu, H., C. Li, Y. Li, et al. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26296–26306. 2024.

[18] Alayrac, J.-B., J. Donahue, P. Luc, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736, 2022.

[19] Liu, H., C. Li, Y. Li, et al. Llavanext: Improved reasoning, ocr, and world knowledge, 2024.

[20] Liu, H., C. Li, Q. Wu, et al. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023.

[21] Li, J., D. Li, S. Savarese, et al. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023.

[22] Team, G., R. Anil, S. Borgeaud, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[23] Liu, A., B. Feng, B. Wang, et al. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*, 2024.

[24] Chen, Z., W. Wang, Y. Cao, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint arXiv:2412.05271*, 2024.

[25] Lin, B., Y. Ye, B. Zhu, et al. Video-llava: Learning united visual representation by alignment before projection. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pages 5971–5984. 2024.

[26] Wang, P., S. Bai, S. Tan, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[27] Shang, Y., M. Cai, B. Xu, et al. Llava-prumerge: Adaptive token reduction for efficient large multimodal models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22857–22867. 2025.

[28] Qin, J., X. Zou, D. Lu, et al. Sharp eyes and memory for videollms: Information-aware visual token pruning for efficient and reliable videollm reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, pages 8538–8546. 2026.

[29] Huang, X., H. Zhou, K. Han. Prunevid: Visual token pruning for efficient video large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 19959–19973. 2025.

[30] Tao, K., C. Qin, H. You, et al. Dycoke: Dynamic compression of tokens for fast video large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 18992–19001. 2025.

[31] Lin, X., X. Liu, Y. Wang, et al. V-cast: Video curvature-aware spatio-temporal pruning for efficient video large language models. *arXiv preprint arXiv:2603.27650*, 2026.

[32] Cai, K., J. Zhang, Y. Fan, et al. Racot: Plug-and-play contrastive example generation mechanism for enhanced llm reasoning reliability. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, pages 30112–30120. 2026.

[33] Fan, Y., J. Zhang, K. Cai, et al. Cost-effective communication: An auction-based method for language agent interaction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, pages 29412–29420. 2026.

[34] Lu, P., S. Mishra, T. Xia, et al. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in neural information processing systems*, 35:2507–2521, 2022.

[35] Fan, Y., J. Zhang, K. Cai, et al. 3dalign-daer: Dynamic attention policy and efficient retrieval strategy for fine-grained 3d-text alignment at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 40, pages 265–273. 2026.

[36] —. Ccg: Rare-label prediction via neural sem–driven causal game. *Findings of the Association for Computational Linguistics: EMNLP*, 2025:6243–6256, 2025.

[37] Radford, A., J. W. Kim, C. Hallacy, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

[38] Cai, K., J. Zhang, J. Yang, et al. Flashvlm: Text-guided visual token selection for large multimodal models. *arXiv preprint arXiv:2512.20561*, 2025.

[39] Yu, H., W. Li, X. Qu, et al. Visiontrim: Unified vision token compression for training-free mllm acceleration. *arXiv preprint arXiv:2601.22674*, 2026.

[40] Li, S., H. Wu, J. Shao, et al. Seeing the forest and the trees: Query-aware tokenizer for long-video multimodal language models. *arXiv preprint arXiv:2511.11910*, 2025.

[41] Zhang, S., J. Yang, J. Yin, et al. Q-frame: Query-aware frame selection and multi-resolution adaptation for video-llms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22056–22065. 2025.

[42] Tang, X., J. Qiu, L. Xie, et al. Adaptive keyframe sampling for long video understanding. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 29118–29128. IEEE, 2025.

[43] Shen, X., Y. Xiong, C. Zhao, et al. Longvu: Spatiotemporal adaptive compression for long video-language understanding. *arXiv preprint arXiv:2410.17434*, 2024.

[44] Zhang, K., Z. Yang, B. Wang, et al. Querystream: Advancing streaming video understanding with query-aware pruning and proactive response. In *The Fourteenth International Conference on Learning Representations*. 2026.

[45] Fu, C., Y. Dai, Y. Luo, et al. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24108–24118. 2025.

[46] Wu, H., D. Li, B. Chen, et al. Longvideobench: A benchmark for long-context interleaved video-language understanding. *Advances in Neural Information Processing Systems*, 37:28828–28857, 2024.

[47] Li, K., Y. Wang, Y. He, et al. Mvbench: A comprehensive multi-modal video understanding benchmark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22195–22206. 2024.

[48] Zhou, J., Y. Shu, B. Zhao, et al. Mlvu: A comprehensive benchmark for multi-task long video understanding. *arXiv preprint arXiv:2406.04264*, 2(5):6, 2024.

[49] Zhang, K., B. Li, P. Zhang, et al. Lmms-eval: Reality check on the evaluation of large multimodal models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 881–916. 2025.

[50] Bai, S., Y. Cai, R. Chen, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

A Appendix

In the appendix, we provide more experimental results A.1, benchmark and model details A.2, algorithmic details of Q-Foucs A.3 and more visualizations of Q-Foucs A.4.

A.1 More experimental results.

Additional experiments on LLava-Video. In Tab. 10, we present the experimental results of Q-Focus integrated with state-of-the-art compression baselines on LLaVA-Video under retention ratios $R \in \{25\%, 15\%\}$. Notably, integrating Q-Focus consistently improves performance across all benchmarks and baseline strategies.

Table 10: **Performance of Q-Focus integrated into state-of-the-art methods on LLaVA-OneVision and LLaVA-Video.** "Avg." shows the mean performance across different benchmarks. Δ represents the percentage improvement compared to the baseline.

Method	Retention Ratio R	VideoMME				MLVU	LongVideo Bench	MVBench	Avg.		Δ
		Short	Medium	Long	Overall				Score	Rel. Acc (%)	
LLaVA-Video											
Vanilla	100%	77.0	62.1	53.3	64.2	70.3	59.5	60.4	63.6	100.0	
VidCom^2		72.3	60.1	51.8	61.4	58.7	56.6	56.9	58.4	91.8	
+ Q-Focus		**74.2**	**61.0**	**52.7**	**62.6**	**60.1**	**58.1**	**58.1**	**59.7**	**93.9**	**+2.1**
FlashVID	25%	74.7	60.4	52.2	62.4	64.8	59.1	60.4	61.7	97.0	
+ Q-Focus		**75.4**	**61.5**	**52.8**	**63.2**	**65.6**	**59.9**	**60.8**	**62.4**	**98.1**	**+1.1**
VidCom^2		66.3	58.5	50.3	58.4	56.5	52.2	53.3	55.1	86.6	
+ Q-Focus		**73.4**	**61.9**	**51.1**	**62.1**	**58.7**	**55.8**	**57.1**	**58.4**	**91.9**	**+5.3**
FlashVID	15%	73.0	61.5	50.8	61.8	64.2	58.3	60.4	61.2	96.2	
+ Q-Focus		**73.5**	**62.1**	**52.3**	**62.6**	**65.9**	**59.3**	**60.7**	**62.1**	**97.7**	**+1.5**

A.2 Implementation details.

Benchmarks detials.

VideoMME. VideoMME is a comprehensive multi-modal benchmark for evaluating video understanding capabilities of VLLMs. It comprises 900 high-quality videos spanning 6 diverse domains and 30 fine-grained subcategories, with durations ranging from 11 seconds to 1 hour. Each video is annotated with human-verified multiple-choice questions (2,700 in total), systematically categorized into short, medium, and long subsets to enable multi-scale temporal analysis. The benchmark further integrates audio tracks and subtitles, requiring models to perform holistic perception across visual, auditory, and textual modalities.

MLVU. MLVU is designed to challenge VLLMs on long-video understanding with videos ranging from 3 minutes to 2 hours. It features 3,102 multiple-choice questions across 9 diverse tasks, including topic reasoning, anomaly detection, action counting, temporal ordering, and plot-based question answering. The benchmark jointly evaluates fine-grained spatial recognition and long-range temporal reasoning, requiring models to maintain coherent context over extended durations while retrieving event-level details and summarizing global narratives.

LongVideoBench. LongVideoBench targets long-context, interleaved video-language understanding with 3,763 videos (8 seconds to 1 hour) and 6,678 human-annotated multiple-choice questions. Its core innovation is a novel "referring-reasoning" task, where models must precisely retrieve and reason over specific multimodal contexts referenced in the questions. The benchmark spans 17 fine-grained categories across perception and relation levels, and incorporates original or transcribed subtitles to evaluate robustness to interleaved textual inputs.

MVBench. MVBench systematically evaluates temporal reasoning in video understanding through 20 distinct tasks generated via a static-to-dynamic transformation pipeline. By extending image-based benchmarks with temporal dynamics, it requires models to perform complex reasoning beyond single-frame perception, covering perspectives from first-person to third-person views. The benchmark emphasizes challenges such as action sequencing, causal inference, and state tracking, providing

Algorithm 1 Q-Focus: Question-Guided Token Compression

Require: Visual tokens $\mathbf{X}^v$, Query tokens $\mathbf{X}^t$, Target retention ratio R, Hyperparameters α, β, k, σ
Ensure: Compressed tokens $\hat{\mathbf{X}}^v_{\text{final}}$

// Stage 1: Global Compression

$\hat{\mathbf{X}}^v, \mathbf{s}^{\text{visual}} \leftarrow \text{GlobalCompression}(\mathbf{X}^v, R)$ ▷ e.g., VidCom^2 or FlashVID

// Stage 2: Question Focusing

$\mathbf{q}^+ \leftarrow \text{Normalize}(\text{Mean}(\mathbf{X}^t))$

$s_i^+ \leftarrow \langle \mathbf{v}_i, \mathbf{q}^+ \rangle$ for all $\mathbf{v}_i \in \hat{\mathbf{X}}^v$

$\tilde{s}_i \leftarrow \text{Z-Score}(s_i^+)$

$\mathbf{q}^- \leftarrow \sum_i \text{Softmax}(-\tilde{s}_i/\tau) \cdot \mathbf{v}_i$ ▷ Adaptive negative sample

$s_i^{\text{contrast}} \leftarrow \langle \mathbf{v}_i, \mathbf{q}^+ \rangle - \langle \mathbf{v}_i, \mathbf{q}^- \rangle$ ▷ ACM

$a_{ij} \leftarrow \langle \mathbf{v}_i, \mathbf{v}_j \rangle$ for all i, j ▷ Affinity matrix

for each token i **do**

  $\mathcal{N}(i) \leftarrow \text{TopK}(\{a_{ij}\}_j, k)$

  $\alpha_{ij} \leftarrow \frac{\exp(a_{ij}/\sigma)}{\sum_{l \in \mathcal{N}(i)} \exp(a_{il}/\sigma)}$ for $j \in \mathcal{N}(i)$

  $s_i^{\text{content}} \leftarrow \sum_{j \in \mathcal{N}(i)} \alpha_{ij} \cdot s_j^{\text{contrast}}$ ▷ QCM

end for

$s_i^{\text{final}} \leftarrow s_i^{\text{visual}} + \alpha \cdot s_i^{\text{contrast}} + \beta \cdot s_i^{\text{content}}$ ▷ Multi-source fusion

$\hat{\mathbf{X}}^v_{\text{final}} \leftarrow \text{TopK}(\hat{\mathbf{X}}^v, \mathbf{s}^{\text{final}}, \lceil R \cdot |\hat{\mathbf{X}}^v| \rceil)$

return $\hat{\mathbf{X}}^v_{\text{final}}$

a rigorous testbed for assessing a model's ability to comprehend and reason over evolving video content.

Model details.

LLaVA-OneVision. LLaVA-OneVision introduces a unified architecture capable of handling single-image, multi-image, and video inputs within a single framework. By encoding videos as interleaved sequences of visual tokens—consistent with its image representation—the model enables seamless knowledge transfer from static visual pre-training to dynamic temporal understanding. This design eliminates the need for task-specific heads and facilitates robust zero-shot generalization across diverse video benchmarks, establishing a versatile foundation for multi-modal reasoning.

LLaVA-Video. Building upon the image-stage checkpoints of LLaVA-OneVision, LLaVA-Video is further optimized through instruction tuning on the large-scale synthetic corpus LLaVA-Video-178K. This dataset encompasses detailed video captioning, open-ended dialogue, and multiple-choice reasoning tasks. Architecturally, the model integrates the SigLIP vision encoder with the Qwen2 language backbone, yielding significant improvements in temporal comprehension and achieving state-of-the-art performance across multiple video understanding benchmarks.

Qwen3-VL. Qwen3-VL represents the latest iteration in the Qwen-VL series, featuring substantial enhancements in visual perception and reasoning efficiency. The model supports high-resolution input processing and incorporates advanced token compression strategies to handle long video sequences with reduced computational overhead. With an upgraded LLM backbone optimized for complex logical reasoning and multilingual interaction, Qwen3-VL excels at fine-grained visual grounding and following intricate instructions within extended video contexts.

A.3 Algorithm details of Q-Focus.

Algorithm 1 outlines the complete inference pipeline of the proposed Q-Focus framework. The procedure is explicitly decoupled into two sequential stages to emulate human coarse-to-fine cognition. Stage 1 leverages an off-the-shelf global compression method to eliminate spatial-temporal redundancy, yielding a compact token set $\hat{\mathbf{X}}^v$ and initial importance scores $\mathbf{s}^{\text{visual}}$. Stage 2 injects question semantics through the Adaptive Contrastive Module (ACM) and Question Context Module (QCM). ACM dynamically computes contrastive scores $\mathbf{s}^{\text{contrast}}$ by aligning visual tokens with a normalized query embedding and subtracting an adaptively constructed negative sample to suppress semantic noise. Subsequently, QCM propagates these scores to semantically related neighbors via a sparse

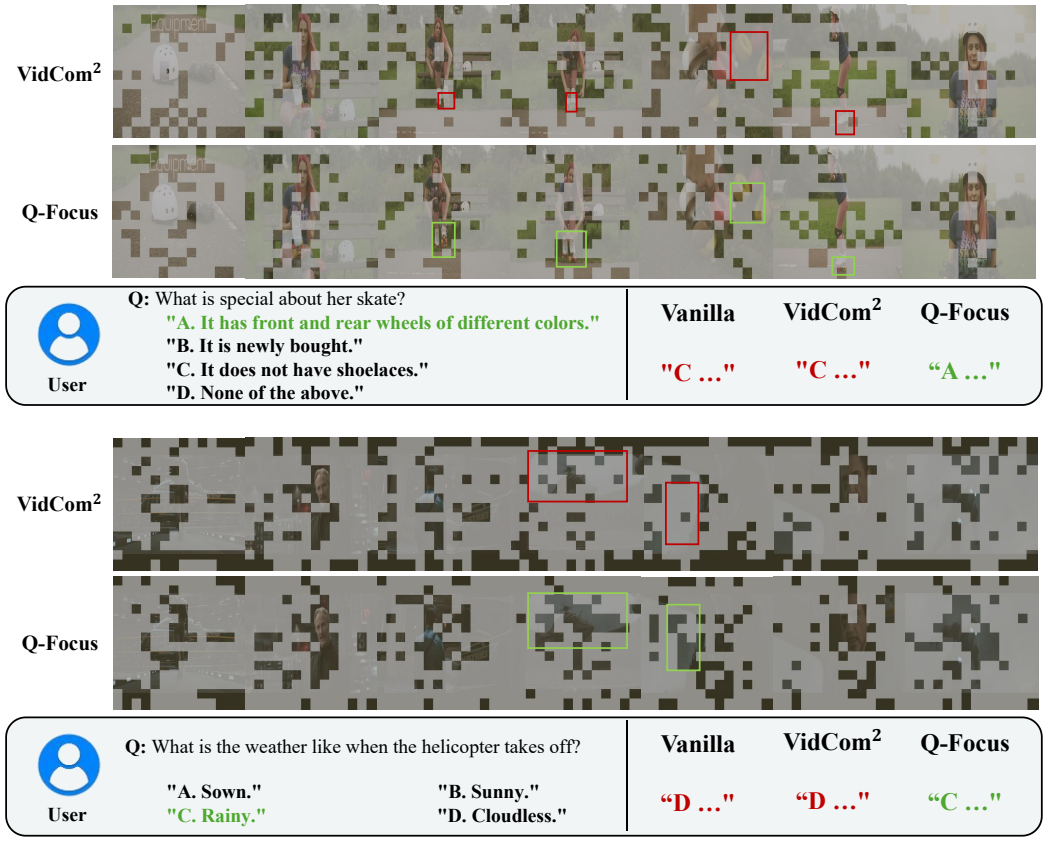

Figure 4: **Qualitative visualization of token retention.** Gray regions denote discarded tokens, while clear regions represent retained visual information.

affinity graph, producing contextual scores $s^{content}$. Finally, a multi-source fusion strategy (Eq. 11) aggregates all three signals to determine the optimal token subset for efficient and question-aware LLM inference.

A.4 More visualizations.

Query-guide token select.

Fig. 4 presents qualitative visualization comparing the token compress method of VidCom^2 on LLaVA-OneVision. The results demonstrate that Q-Focus effectively concentrates on question-relevant details by allocating higher attention weights to semantically aligned tokens. This targeted focus ensures that answer-critical contextual cues are preserved, enabling the model to maintain robust reasoning capabilities and generate precise responses, whereas the baseline tends to disperse attention across redundant or irrelevant regions.

More qualitative analysis.

Figure 5 presents further qualitative comparisons among Q-Focus, representative token compression baselines, and the original uncompressed VideoLLM. The evaluated cases span diverse and challenging video understanding scenarios, encompassing rapid action transitions, subtle semantic shifts, and complex long-horizon reasoning. As demonstrated, Q-Focus dynamically allocates the token budget to question-relevant regions, thereby more effectively isolating and preserving tokens that are critical for accurate response generation. Notably, in intricate scenarios where conventional compression methods and even the baseline model occasionally overlook pivotal visual cues, Q-Focus consistently focus the necessary context to yield correct predictions.

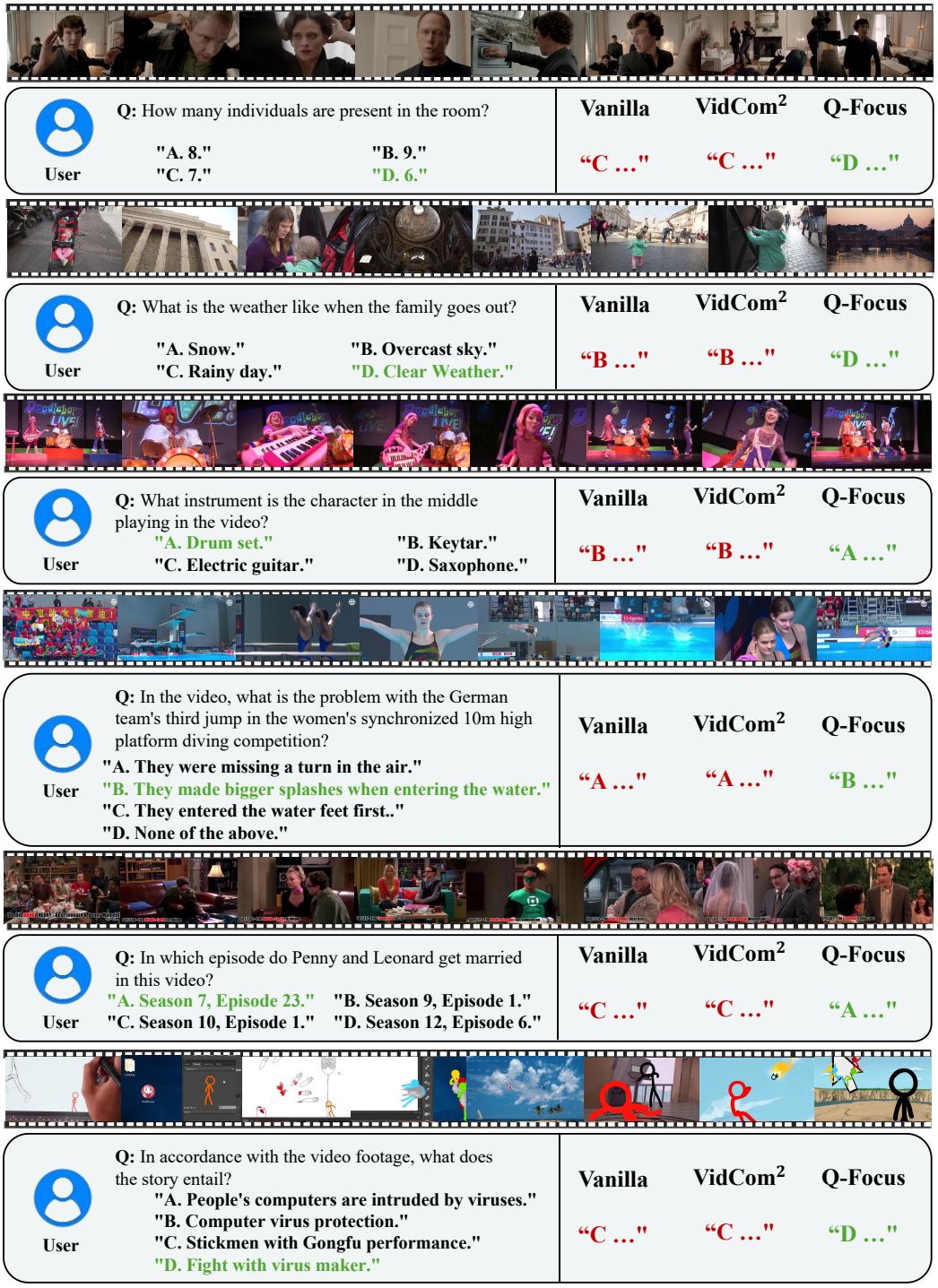

Figure 5: **Additional qualitative comparisons.** Q-Focus preserves task-critical moments and correctly answers challenging cases where baseline compression methods.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The paper's contributions and scope are clearly claim in the abstract and summarized at the end of the introduction.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The discussion addresses the slight speed overhead incurred by the proposed method.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: The Methods section contains a complete and correct proof.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The experimental section fully discloses all information necessary to reproduce its key experimental results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Due to company policy, we will open-source the code in the future; in the supplementary materials, we have provided pseudocode.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: We present the training and testing details in detail in the experimental section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We have presented all test details in detail in the experimental section.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We have provided sufficient information.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Complies with the NeurIPS Code of Ethics in all respects.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Will not result in negative social impact.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Token compression tasks do not involve such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: This item has been properly handled.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: We will open-source the code in the future.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: the paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: the paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: the core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.