NEXUS: A CAUSAL-AWARE FRAMEWORK FOR MULTIMODAL LONG-FORM MOVIE UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Understanding long-form videos, especially movies, requires jointly integrating visual, audio, and dialogue signals while persistently tracking character dynamics and event causality. Agentic models are increasingly used to tackle this challenge. However, these approaches either rely on subtitles as a stand-in for video, leaving other modalities underutilized, or reason directly over raw frames at a computational cost that scales sharply with video length and query complexity. Moreover, the field is largely evaluated through multiple-choice (MC) benchmarks, which do not always demand the depth of reasoning that free-form responses require. Under this lens, we introduce **NEXUS**, a framework for long-form video understanding built around three core components. First, we propose **NEXUS-MF**, an end-to-end multimodal understanding pipeline that processes long videos into a structured context representation capturing character dynamics and long-term causal dependencies. Second, we present **NEXUS-OEBench**, a large-scale, human-annotated, open-ended QA benchmark comprising **1,701** question–answer pairs spanning 150+ hours of video content, with questions demanding deep causal and multimodal reasoning. Third, we introduce **NEXUS-QA**, a video QA agent that causally reasons over NEXUS-MF's structured representation, actively seeking and verifying evidence to construct complete causal chains before forming a final response. NEXUS outperforms baselines on QA benchmarks with $3.4\times$ fewer tokens and $3.3\times$ lower costs, while maintaining stable computational overhead as video length increases. Additionally, NEXUS-MF's multimodal representation generalizes well on the context-retrieval task.

1 INTRODUCTION

Video understanding poses multiple challenges, including multimodal understanding and long-term causal dependencies over a storyline (Zhou et al., 2024; Nagrani et al., 2024). A video scene is jointly composed of the visual (what is seen), audio (what is heard), and dialogue (what is said) channels (Arandjelović & Zisserman, 2017; Roth et al., 2020); therefore, understanding a scene well requires integrating multiple modalities (Baltrušaitis et al., 2018). Consider a video of two people walking with their backs to the camera; the visual track alone cannot show whose lips are moving, and the dialogue track itself does not distinguish the character that is speaking. Existing work only partially addresses this challenge. Many work rely on closed captions (CC) to localize the context and reason over CC (Zhang et al., 2024; Ventura et al., 2025; Lin et al., 2026), while visual or audio signals still deliver crucial information about the scene, such as the personalized voices or who is on screen during a specific line of dialogue, not to mention information that is delivered only through a specific channel, such as text on a sign or on a computer screen.

Beyond integrating multiple modalities, another challenge of video understanding is to follow the storyline itself and understand the long-term causal dependencies between events and the interactions between characters, including who the characters are and how their relationships and roles change as events unfold (Vicol et al., 2018; Kukleva et al., 2020). Long-form video, e.g., a movie with hours of footage, amplifies this challenge even further, since the number of events is much higher, leading to more complicated dynamics between characters and a longer context length over which to trace their dependencies. Current benchmarks confirm the resulting gap empirically, showing models recover local, single-scene facts far more reliably than the character arcs and relationship changes that unfold over a story (Tian et al., 2026; Asgarov et al., 2026). In addition, specific to the

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

movie domain, the writer's or director's intent is a further challenge that might be captured to fully understand the video (Shinozaki-Conefrey et al., 2026).

Potential downstream tasks, such as video summary generation, event localization, or recap video generation, built on top of a video-understanding backbone model are less explored, with most work targeting one specific task, such as video QA (Zhang et al., 2024) or chapter-level summarization (Ventura et al., 2025), rather than the general causal dependencies or the multiple aspects of understanding as a reusable package that other downstream tasks could later draw on. Existing work such as WorldMM (Yeo et al., 2026) and MemDreamer (Chen et al., 2026) has taken steps in this direction, building a structured graph or memory that a downstream task could potentially query. However, their designs target and are evaluated only on the QA task specifically, leaving their reusability for other downstream purposes untested. This absence of a persistent representation also drives the cost problem, since agents need to re-derive evidence from raw video every time (Lin et al., 2026; Wang et al., 2024b), so cost scales sharply with video length and the number of queries.

Existing benchmarks for video QA are dominated by multiple-choice (MC) questions (Lei et al., 2018; Mangalam et al., 2023). Yet MC and open-ended question answering probe different aspects of understanding, even over the same content (Xiao et al., 2021). For instance, with candidate answers provided, a model tends treat each option as a hypothesis and search for evidence that supports or contradicts it (Tapaswi et al., 2016; Rawal et al., 2024; Goyal et al., 2017; Loginova et al., 2025), reaching a correct prediction from a single locally-matching cue without necessarily recovering the fuller causal chain behind it (Lei et al., 2023). Open-ended question answering removes this scaffold with no options to check against, a model is required to identify relevant evidence and construct rather than select its answer (Xiao et al., 2021), so a correct response better reflects the depth of its causal and multimodal understanding. Additionally, many video QA benchmarks are built from short clips rather than long-form videos (Xu et al., 2017; Jang et al., 2017; Xiao et al., 2021).

To resolve these gaps, we introduce **NEXUS**, a framework for long-form video understanding with three key components. First, **NEXUS-MF** (Multimodal Understanding Foundation) processes a long video *once* into a persistent, structured context representation that captures character profiles, inter-character relationships, and long-term causal dependencies between events, grounded jointly in vision, audio, and text. We refer to this representation as a *multimodal understanding package*, reusable across downstream tasks. Second, **NEXUS-OEBench** is a large-scale, human-annotated, open-ended QA benchmark comprising 1,701 question–answer pairs across 150+ hours of movie content, designed to demand deep causal and multimodal reasoning over long storylines. Third, **NEXUS-QA** is a video QA agent that reasons causally over NEXUS-MF's representation, reducing computational cost and time while improving answer quality. We further incorporate a Structural Causal Model (SCM) (Pearl, 2009) into NEXUS-QA to mitigate the influence of misleading evidence and guide the agent toward identifying true causal factors. We summarize our contributions as follows:

- We propose NEXUS, a framework for long-form video understanding, including a multimodal understanding foundation (NEXUS-MF), a large-scale human-annotated open-ended QA benchmark (NEXUS-OEBench), and a video QA agent (NEXUS-QA).
- We demonstrate the reusability of NEXUS-MF's structured representation across tasks. In particular, NEXUS-QA outperforms existing video QA agents on both NEXUS-OEBench and multiple-choice (MC) benchmarks while reducing computational cost.
- NEXUS paves the way for open-ended benchmarking beyond multiple-choice (MC) evaluation, with the flexibility to extend to a wider range of video understanding tasks.

2 RELATED WORK

Video Question-Answering Agents A growing number of agentic approaches tackle long-form video understanding, including VideoAgent (Wang et al., 2024b), VideoSeek (Lin et al., 2026), WorldMM (Yeo et al., 2026), MemDreamer (Chen et al., 2026), and OmniScript (Pu et al., 2026), where **VideoSeek** and **WorldMM** are the closest existing work to NEXUS-QA. **VideoSeek** (Lin et al., 2026) searches for evidence separately for each question, using a toolkit of *overview*, *skim*, and *focus* that sample frames at sparse, medium, and dense frequencies, respectively. A thinking model decides tool calls and judges when enough evidence has been gathered, while a separate

tool-execution model performs the toolkit. This design performs well on few-hour-long video but grows costly and slow as video length and the number of questions increase. **WorldMM** (Yeo et al., 2026) is a dynamic multimodal memory agent for long-video reasoning. WorldMM constructs three complementary memory stores (episodic memory, semantic memory, and visual memory), and an adaptive retrieval agent, with a separate retrieval model and thinking model, selects which memory source and temporal granularity to draw from for a given query. However, this design limits the flexibility in looking for query-based evidence or causal relations that are not in the memories.

Video Question-Answering Benchmarks A number of existing benchmarks evaluate video QA on short or moderate-length video. NExT-QA (Xiao et al., 2021), MovieChat-1K (Song et al., 2024), MLVU (Zhou et al., 2024), ActivityNet-QA (Yu et al., 2019), FriendsQA (Wu et al., 2025), and Neptune (Nagrani et al., 2024) each combine some subset of open-ended answering, human annotation, and narrative content, though none yet reach genuinely long-form video. Fewer benchmarks target genuinely long-form video: HLVU (Curtis et al., 2020), LVBench (Wang et al., 2024a), EgoLifeQA (Yang et al., 2025), MINERVA (Nagrani et al., 2025), and 1H-VideoQA (Gemini Team, 2024) each reach it by trading off some other dimension, such as scale, answer format, or narrative content. Appendix D.2 discusses every benchmark above, short- and long-form alike, in more detail.

3 NEXUS

In this section, we present the three core components of NEXUS in detail. **NEXUS-MF** (§3.1), a multi-step agent that processes a long video once into a persistent, structured representation of its characters, relationships, and causal dependencies to be reused in later steps and tasks; **NEXUS-OEBench** (§3.2), a large-scale, human-annotated, open-ended QA benchmark for long-form video understanding; and **NEXUS-QA** (§3.3), a video agent that reasons causally over NEXUS-MF's representation and assembles evidence into a complete causal chain to perform QA tasks with reduced costs. Figure 1 illustrates an overview of NEXUS.

3.1 NEXUS-MF

NEXUS-MF processes the raw multimodal input, including video V, audio track A, and closed captions C, into a persistent, structured understanding package $\mathcal{R}$ (Eq. 1), consisting of (1) character-attributed closed captions $\hat{C}$, in which most lines now carry a predicted character identity; (2) a sequence of K context-chunk summaries $\{X_k\}_{k=1}^{K}$ localizing content over the video's timeline; and (3) a relation graph G_{rel} and an event graph G_{event} capturing characters' interactions. NEXUS-MF builds $\mathcal{R}$ using three components: diarization and chunking, a multimodal understanding agent, and graph extraction.

$$\mathcal{R} = \left(\hat{C},\ \{X_k\}_{k=1}^{K},\ G_{\text{rel}},\ G_{\text{event}}\right). \tag{1}$$

Chunk summary $X_k = \left(X_k^{\text{overview}},\ X_k^{\text{detail}},\ X_k^{\text{script}}\right)$ itself has three parts of different length and focus: a short *overview* for high-level context, a medium-length *detail* focused on character actions and dialogue, and a long, fine-grained *script* that captures visual and on-screen text information.

3.1.1 DIARIZATION AND CHUNKING

Audio Diarization Closed captioning carries *what* is said but not *who* says it, while human viewers naturally identify speakers through voice nuances in the audio. To recover this speaker information, we pass each video's audio through NVIDIA NeMo speaker diarization model (Kuchaiev et al., 2019; Park et al., 2022), producing n speech segments S (Eq. 2), where segment i spans t_i^{start} to t_i^{end} and carries an unassigned speaker label s_i drawn from the set $\mathcal{S}$ of speakers detected by the model. The mapping from $\mathcal{S}$ to character identities remains many-to-many due model errors: a single character may be split across several labels, and a single label may merge multiple characters. Segments S are then used as inputs to the character-assignment step that follows (Eq. 3).

$$S = \left\{(t_i^{\text{start}}, t_i^{\text{end}}, s_i)\right\}_{i=1}^{n}, \qquad s_i \in \mathcal{S}. \tag{2}$$

Context Chunking Unlike prior work (Yeo et al., 2026; Chen et al., 2026; Pu et al., 2026) that relies on fixed-length chunks, we introduce context-aware chunks $\{X_k\}_{k=1}^{K}$ tailored to the video's

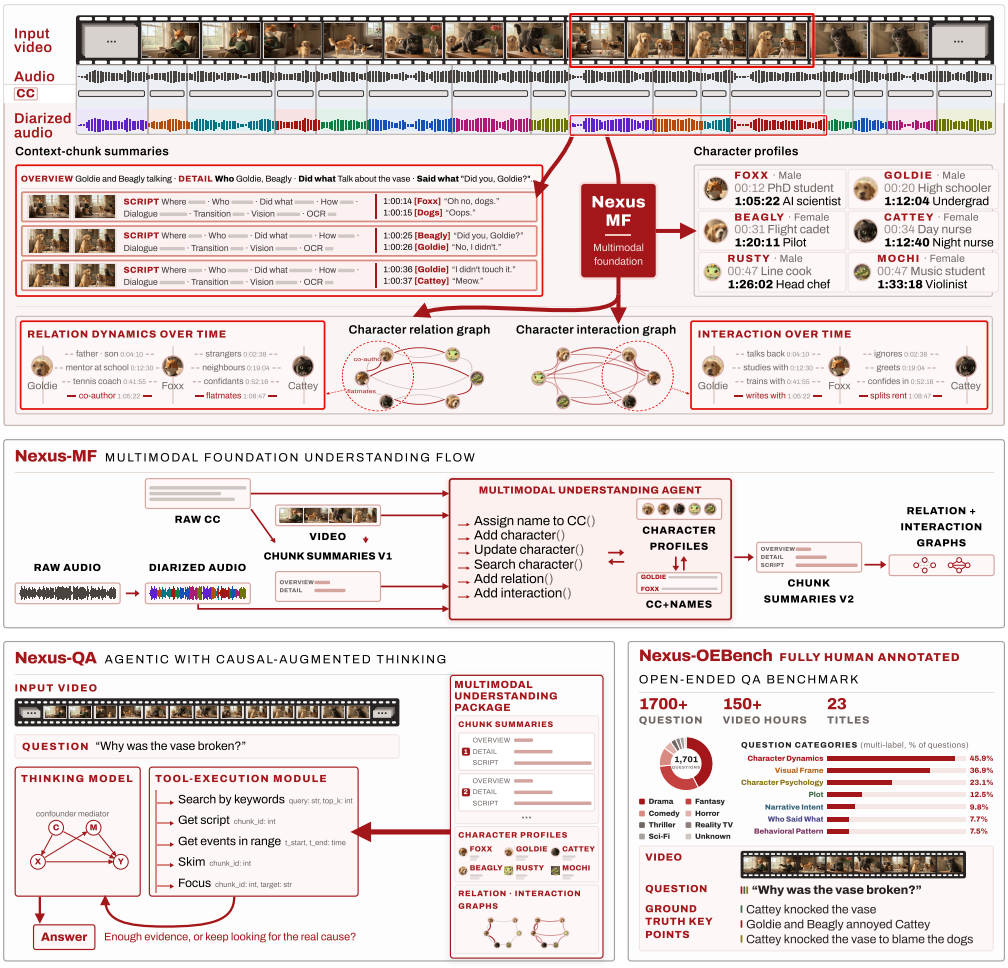


Figure 1: **NEXUS overview**. **NEXUS-MF** processes a video as a multimodal understanding package via a multi-step agent pipeline. **NEXUS-QA**, an agentic system that reasons over the package with a toolkit for searching evidence to answer questions. **NEXUS-OEBench** is a large-scale, open-ended, human-annotated QA benchmark.

semantic flow. Starting from 10-minute blocks, a context-chunking model g_θ (e.g., a VLM) generates sparse frame descriptions for video V sampled at 1 frame every 20 seconds, and divides each block into smaller chunks based on contextual shifts in these descriptions. Subsequently, g_θ constructs an *overview* X_k^{overview} (1–2 sentences) and a *detail* X_k^{detail} ($\sim$150 words) for each chunk k, addressing three core questions: *Who?*, *Did what?*, and *Said what?*.

3.1.2 MULTIMODAL UNDERSTANDING AGENT

Agent Workflow As illustrated in Figure 1, this stage pursues two intertwined targets: character-attributed closed captions $\hat{C}$, which resolve *who* speaks each line, and a character-profile database P, which tracks character dynamics (Appendix E). Taking each chunk's speech segments S_k, raw captions C_k, and detail summary X_k^{detail} as input, a multimodal understanding agent f_θ processes the K context chunks sequentially to jointly produce both targets (Eq. 3).

$$(P_k,\ \{\hat{c}_i\}_{i\in S_k}) = f_\theta\big(P_{k-1},\ X_k^{\text{detail}},\ S_k,\ C_k\big)\,, \qquad k = 1, \ldots, K. \tag{3}$$

Here, $P_0 = \emptyset$; S_k and C_k denote the speech segments and raw caption lines within chunk k; and $\hat{c}_i$ represents the resolved speaker for segment i. Each resolved label $\hat{c}_i$ is assigned to its caption line to build $\hat{C}$, while each profile $p \in P_K$ tracks character evolution, inter-character relationships, and interactions throughout the video. These two outputs mutually strengthen each

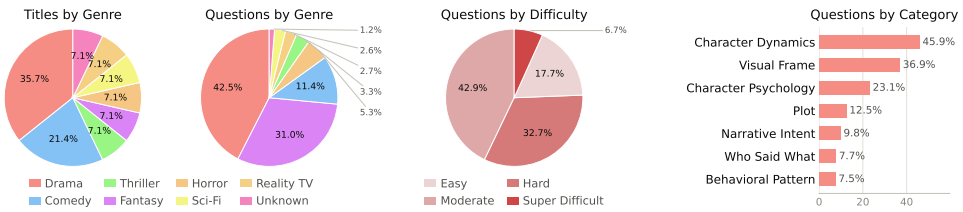


Figure 2: **NEXUS-OEBench distributions by genre, difficulty, and category.**

other: prior context in P_{k-1} helps speaker identification, and each newly resolved speaker folds into P_k as new evidence. Once $\hat{C}$ is established for the entire video, model g_θ constructs the third summary, *script* X_k^{script}, which aims to address an expanded set of fine-grained questions per moment: *Where?*, *Who?*, *Did what?*, *How?*, *Said what?*, *Any transition?*, *Any visual information?*, and *Any readable text?*. Each entry is derived from a frame pair, sampled one second apart, every 10 seconds, and capped at 50 words per pair. To perform these updates, f_θ utilizes five specialized tools: four memory-management tools (`add_character`, `update_profile`, `update_relation`, and `update_interaction`) to incrementally construct character states, and a retrieval tool (`cross_chunk_lookup`) to search neighboring chunks when local context in chunk k is insufficient.

3.1.3 GRAPH EXTRACTION

This stage consolidates all previously extracted context—character-attributed closed captions ($\hat{C}$), final character profiles (P_K), and chunk summaries ($\{X_k\}_{k=1}^K$)—into unified relation and event graphs via a single LLM call h_θ (Eq. 4):

$$G_{\text{rel}},\ G_{\text{event}} = h_\theta\Big(\hat{C},\ P_K,\ \{X_k\}_{k=1}^K\Big)\,. \tag{4}$$

The relation graph G_{rel} captures evidence-based relationships between characters, while the event graph G_{event} decomposes this narrative evidence into discrete, structured events to facilitate downstream querying (Appendix F).

3.2 NEXUS-OEBENCH

NEXUS-OEBench consists of **1,701** questions across 23 sources derived from 14 distinct films and TV shows. 11 of the 14 titles serve as individual sources; the remaining three (longer shows) are re-cut into multiple sources of varying video lengths (e.g., a 5-version title where we curate 5 sets of questions requiring context spanning 5, 10, 15, 20, and 25 episodes, and two 3-version titles following the same construction). Each question is manually paired with a set of 1–6 free-form keypoints, written by a group of annotators who have good knowledge of these titles, with sensitive content excluded. For titles with multiple context length versions, questions are accumulated progressively across lengths, such that each longer version subsumes all questions from its shorter counterparts.

NEXUS-OEBench is categorized across eight genres, four difficulty levels, and seven question types, labeled by annotators as detailed in Figure 2 and Appendix H.1. The benchmark diversifies question types to require high-level comprehension of character arcs, story structure, and multimodal context. Specifically, the seven question types are: plot (factual narrative events), character dynamics (evolving relationships over time), character psychology (character motivations and mental states), narrative intent (the creative purpose behind a scene), visual frame (details grounded solely in visual frames), behavioral pattern (implicit recurring actions across scenes), and who-said-what (speaker attribution for dialogue). Individual questions may carry multiple category tags (Appendix H.2). Finally, Table 1 compares NEXUS-OEBench against existing long-form video benchmarks, demonstrating that NEXUS-OEBench bridges the key gaps identified in Sections 1 and 2.

3.3 NEXUS-QA

NEXUS-QA is a video question-answering agent that reasons over NEXUS-MF's representation $\mathcal{R}$ (Eq. 1) to answer a question q. Inspired by VideoSeek (Lin et al., 2026), we adopt a think-act-observe trajectory formulation $\tau = (\langle z_1, a_1, o_1\rangle, \ldots, \langle z_t, a_t, o_t\rangle, \ldots)$, governed by a thinking model θ_{think}

Table 1: **Comparison with existing long-form video question-answering benchmarks.**

Benchmark	Mean Length	Max Length	Questions	Format
HLVU (Curtis et al., 2020)	∼1.5 hr	–	<1,000	KG-query + QA
LVBench (Wang et al., 2024a)	1.1 hr	2.3 hr	1,549	MC
EgoLifeQA (Yang et al., 2025)	∼50 hr / stream	–	500	MC
MINERVA (Nagrani et al., 2025)	0.2 hr	1.5 hr	1,515	MC
1H-VideoQA (Gemini Team, 2024)	–	1.75 hr	101–125	MC
NEXUS-OEBench (ours)	**6.80 hr**	**28.5 hr**	**1,701**	**Open-ended**

and a tool-execution model θ_{tool}. At turn t, θ_{think} conditions on the trajectory history $\tau_{<t}$, question q, and representation $\mathcal{R}$ to produce a reasoning trace z_t and an action a_t:

$$(z_t, a_t) = \theta_{\text{think}}(\tau_{<t}, q, \mathcal{R}), \qquad a_t \in \mathcal{T} \cup \{\texttt{answer}\}. \tag{5}$$

Action a_t is either a tool call selected from set $\mathcal{T}$ to gather further evidence or `answer` to generate the final output. When $a_t \in \mathcal{T}$, the execution model generates observation $o_t = \theta_{\text{tool}}(a_t, \mathcal{R})$, and step $\langle z_t, a_t, o_t \rangle$ is appended to τ. NEXUS-QA introduces three core modifications to this loop to leverage $\mathcal{R}$ efficiently without re-extracting evidence from raw video:

First, NEXUS-QA drops the full closed captions (CC) from the input prompt—significantly reducing token overhead—and instead uses NEXUS-MF's relation graph G_{rel} as a lightweight overview.

Second, we augment toolset $\mathcal{T}$ with three offline retrieval tools (Appendix P analyzes tool usage dynamics across video lengths):

- `search_script`: Accepts keywords from θ_{think} to query chunk scripts $\{X_k^{\text{script}}\}_{k=1}^{K}$, returning the matching chunk's time window and overview summary X_k^{overview}.
- `get_script`: Retrieves the complete script X_k^{script} for a specified chunk k.
- `get_event_range`: Accepts a temporal range $[t_{\text{start}}, t_{\text{end}}]$ and returns all corresponding events from G_{event}.

While `skim` and `focus` are retained from VideoSeek, we discard the baseline's *overview* tool due to its poor scalability on ultra-long videos.

Third, we propose **NEXUS+SCM**, which incorporates a Structural Causal Model (Pearl, 2009) (Appendix G) to prevent the agent from accepting confounding correlations as causal evidence. When evaluating whether $\tau_{<t}$ contains sufficient evidence to stop, θ_{think} is guided by a causal-role classifier $\rho_\theta : o \mapsto \{\text{cause}, \text{confounder}, \text{mediator}\}$. This classifier assesses whether gathered observations reflect a true cause (or intermediate mediator) rather than a mere confounder (events co-occurring with the outcome). If the evidence is identified as confounding or incomplete, the agent continues retrieving evidence in subsequent turns.

4 EXPERIMENTS ON VIDEO QUESTION ANSWERING

4.1 VIDEO QUESTION ANSWERING ON NEXUS-OEBENCH

Setup Given a question about a video, the QA model answer in free-form plain text with no required format or length constrains. To evaluate the model's answer against the ground-truth keypoints, we design a panel of three LLM evaluators (Claude Opus 5, Gemini 2.5 Pro, GPT-5.4-mini), each working independently to mark which keypoints the answer covers, and a final judge (o3) reconciles their verdicts. We report keypoints *recall* as the primary metric (see Appendix B for more details). We compare NEXUS against two most recent baselines, **VideoSeek** (Lin et al., 2026) and **WorldMM** (Yeo et al., 2026). VideoSeek and NEXUS share the same high-level agentic design with a thinking model θ_{think} and the tool-execution model θ_{tool}, with more details in Appendix A.

QA Performance Comparison As shown in Table 2, NEXUS outperforms both baselines by a large margin with markedly fewer tokens and costs per question than both baselines. Preprocessing costs are amortized across questions for both NEXUS and WorldMM, with NEXUS preprocessing $5\times$ faster than WorldMM, as shown in Figure 3. NEXUS also achieves lower wall-clock time than both baselines, as NEXUS-QA queries the persistent NEXUS-MF representation offline, which avoids latency and cost increases as the complexity of questions or video length scales (Appendix P analyzes

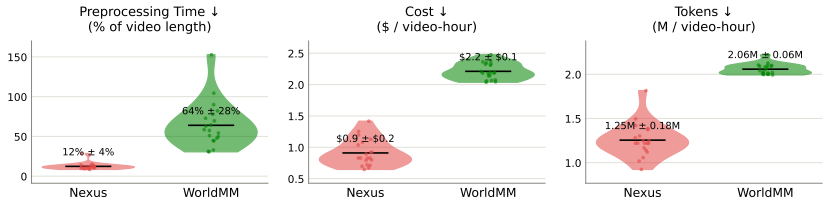


Figure 3: **Distribution of preprocessing time, cost, and token usage per video-hour, NEXUS vs. WorldMM.** NEXUS's preprocessing is cheaper, faster, and lighter than WorldMM's on all metrics.

Table 2: **QA results on NEXUS-OEBench.** Recall differences (Δ) and multipliers are relative to VideoSeek; Standard deviation is computed over per-title recall across context sources. Overall, NEXUS in both configurations outperforms both baselines on all metrics shown.

Engine	Recall (%) ↑	Tokens/Q ↓	Cost/Q ↓	Time/Q (s) ↓
WorldMM	40.0 ± 12.3	312.2K	$0.913	110.8
VideoSeek	51.9 ± 19.3	750.0K	$0.444	83.5
NEXUS	(Δ+12.3) **64.2** ± 9.8	(3.3× fewer) 225.7K	(3.1× cheaper) $0.141	(1.2× faster) 70.5
NEXUS+SCM	(Δ+13.0) **64.9** ± 10.5	(3.4× fewer) **217.6K**	(3.3× cheaper) **$0.135**	(1.4× faster) **59.2**

how NEXUS-QA's tool-call mix shifts with video length). Furthermore, NEXUS+SCM not only achieves higher recall than NEXUS but also runs 1.2× faster (Table 2), with the most pronounced gains on questions requiring higher-level reasoning, such as ruling out causal confounders, as illustrated in Table 12.

Effect of Context Length Figure 5 compares QA performance and computational cost across increasing context lengths for the three longer titles with 3 and 5 context length versions, respectively. On the 5-version title, VideoSeek's recall drops 55% and cost grows 3.6× over a 5× length increase (2.6 to 13.5 hours), ultimately failing to complete the run at 28.5 hours with exceeding the API token budget. NEXUS, in contrast, maintains stable QA performance as video length grows 11× over the full range (2.6 to 28.5 hours), with recall dropping only 34% and cost growing under 2×. On the 3-version titles, over a 2.1× length increase (6.5 to 13.8 hours), VideoSeek's recall drops 44%, and cost grows 1.8× while NEXUS's recall instead drops only 18%, with cost growing under 1.5× over the same range. This same trend holds across NEXUS-OEBench's full set of titles and versions (Figure 4): as video length grows, VideoSeek's recall falls below NEXUS's and NEXUS+SCM's while its tokens/question grow far larger. Overall, NEXUS is more robust to increasing video length as it searches NEXUS-MF's persistent representation $\mathcal{R}$ instead of re-deriving evidence every query.

Ablations Table 5 reports ablations on a subset of four NEXUS-OEBench titles. First, we run VideoSeek on character-attributed CC, which yields a 7% performance gain with significantly fewer tokens, verifying direct impact of including speaker label in CC. Second, we iteratively remove individual components from NEXUS-MF's representation, i.e., "OCR", "Where?", or "How?", to verify the importance of each. Results show dropping any individual component leads to a consistent performance drop, demonstrating that all are necessary. Third, we remove NEXUS-MF's relation graph G_{rel} from NEXUS-QA's input and disable the query to `get_event_range` to skip the event graph G_{event}, respectively. Removing either graph lowers recall substantially while raising token usage, as NEXUS-QA must fall back on searching raw chunks for evidence that the graph would otherwise have surfaced directly. Finally, we strengthen NEXUS-MF's character-assignment step with a larger token budget and a character name list extracted from video credits as additional input, yielding a 4.8% improvement in QA performance. For these titles, we manually annotate ground-truth character labels per CC line to compute assignment accuracy and coverage rate.

Web Search Baseline We also compare NEXUS against latest version of the models with web search access, such as GPT, Gemini and Claude. We run those models on NEXUS-OEBench questions by using only the title name and question text, without processing the video. Table 3 reports mean recall across the same four NEXUS-OEBench titles as Table 5. NEXUS+SCM and VideoSeek both outperform all three search-based tools, with NEXUS+SCM achieving a 24.2%–36.7% recall advantage, demonstrating that online information alone, such as movie-related discussions, is insufficient to solve the movie QA task, which requires deep multimodal understanding.

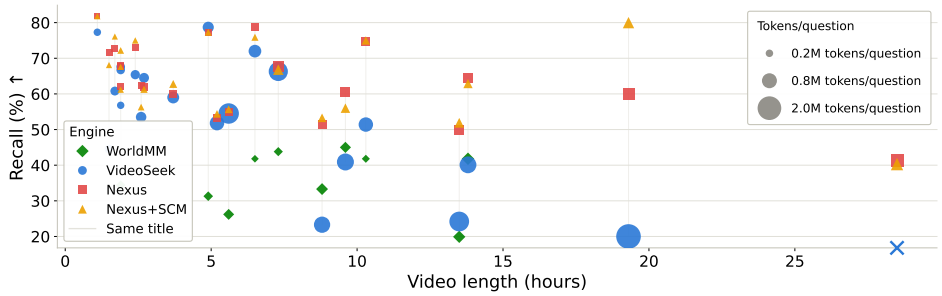


Figure 4: **QA performance comparison in NEXUS-OEBench**. Each thin vertical line marks one context source. The blue × on the x-axis marks an incomplete run. Overall, NEXUS and NEXUS+SCM achieve better QA performance on most context sources while keeping computation cost stably low, especially at longer context lengths

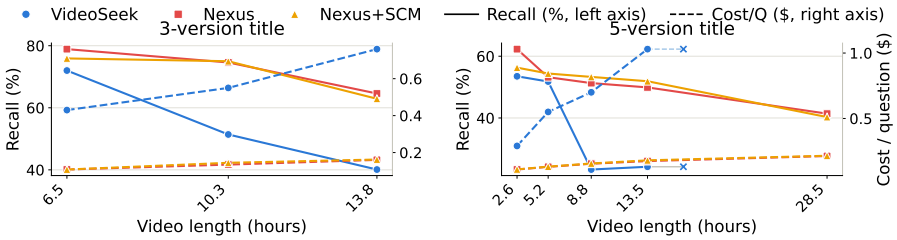


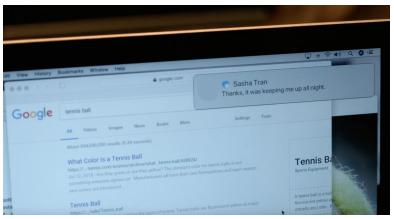
Figure 5: **Recall and cost per question vs. video length for the 3-version and 5-version titles**. As video length grows, VideoSeek's recall falls, and its cost climbs sharply on both titles, while both NEXUS configurations' recall degrades more gently and their costs stay relatively flat.

4.2 PUBLIC BENCHMARKS

We test whether NEXUS-QA's advantage generalizes beyond NEXUS-OEBench using two additional MC benchmarks scored by accuracy, both reported in Table 4 (Appendix L for details). First, on LVBench (Wang et al., 2024a), both NEXUS configurations significantly outperform VideoSeek while reducing computational cost, indicating that NEXUS also generalizes to domains beyond NEXUS-OEBench. Second, MINERVA-Ego (Nagrani et al., 2025) is an egocentric video benchmark; unlike NEXUS-OEBench and LVBench, it contains no dialogue, so NEXUS-MF's character-attributed CC $\hat{C}$ is empty for every title. NEXUS and NEXUS+SCM nonetheless substantially outperform VideoSeek on it, showing that NEXUS-MF's representation remains informative even with a modality absent from its full multimodal preprocessing pipeline. VideoSeek, by contrast, underperforms both NEXUS configurations on this benchmark.

5 EXPERIMENTS ON VISUAL CONTEXT RETRIEVAL

Setup We further evaluate NEXUS's visual grounding capabilities. For each title in NEXUS-OEBench, we sample 20 frames along with their corresponding ground-truth timestamps. We use



Question (Visual Frame, Who-said-what): Who said 'Thanks, it was keeping me up all night'?
Ground Truth: Sasha Tran.
VideoSeek: Could not be found in the movie. **0% recall**
WorldMM: Not mentioned in the movie. **0% recall**
NEXUS: Sasha Tran. **100% recall**
NEXUS+SCM: Sasha Tran. **100% recall**

Figure 6: **Qualitative result comparison.** The quoted line is spoken on screen but never appears in the CC, so answering it requires visual search rather than transcript search alone: VideoSeek and WorldMM are not able to find it, while NEXUS and NEXUS+SCM correctly ground it visually.

Table 3: **General-purpose models with web-search access**, candidates for the web-search baseline, compared against VideoSeek and NEXUS+SCM.

Model	Provider	Recall (%)	Tokens/Q ↓	Cost/Q ↓
Gemini	Google	37.2 ± 20.9	746	$0.0070
ChatGPT	OpenAI	41.1 ± 15.3	23.9K	$0.0311
Claude	Anthropic	28.6 ± 17.6	19.7K	$0.1130
VideoSeek	–	54.1 ± 6.9	1.18M	$0.283
NEXUS+SCM	–	**65.3 ± 8.8**	**342.6K**	**$0.109**

Table 4: **Accuracy on LVBench and MINERVA-Ego.** Both NEXUS configurations achieve higher accuracies than baselines.

Method	LVBench ↑	Tokens/Q ↓	MINERVA-Ego ↑	Tokens/Q ↓
WorldMM	61.9%†	–	–	–
VideoSeek‡	76.7%	–	–	–
VideoSeek	68.12%	588.3K	41.3%	256K
NEXUS	**83.83%**	**140.9K**	52.50%	129K
NEXUS+SCM	83.61%	142.6K	**56.05%**	117K

†WorldMM's own reported value. ‡VideoSeek's own reported value.

Table 5: **Ablations on a subset of four NEXUS-OEBench titles.** Ablation results prove the importance of each NEXUS component. In particular, eliminating the relation event graphs give the largest drops in recall, while significantly more tokens leading to much higher costs.

Variant	Recall (%) ↑	Tokens/Q ↓	Cost/Q ↓	Time (s) ↓	Accuracy (%) ↑	Assignment Rate (%) ↑
VideoSeek + Raw CC	54.1 ± 6.9	1.18M	$0.283	75.0	–	–
VideoSeek + Character-Attributed CC	**61.0** ± 4.2	692.8K	$0.203	63.3	91.0	72.3
NEXUS+SCM (full)	**65.3** ± 8.8	**342.6K**	**$0.109**	**49.7**		
– OCR	64.1 ± 9.2	345.8K	$0.114	51.6		
– Where	61.6 ± 11.3	323.2K	$0.105	45.6	91.0	72.3
– How	62.0 ± 5.9	326.4K	$0.107	47.1		
– Relation Graph	47.5 ± 18.0	543.0K	$0.175	46.9		
– Event Graph	54.5 ± 13.3	439.6K	$0.149	50.4		
NEXUS+SCM + Reinforced CC	70.1 ± 5.7	371.7K	$0.111	50.1	91.4	94.9

Gemini 2.5 Flash to generate an ∼100-word description per frame capturing the background and depicted activity, without mentioning character names or identities (see Appendix M). We then prompt the QA agents (NEXUS, NEXUS+SCM, VideoSeek, and WorldMM) to retrieve the timestamp in seconds corresponding to each description. We report two metrics: the time difference between predicted and ground-truth timestamps, and frame similarity computed via perceptual hashing (pHash) of the predicted and ground-truth frames.

Results Figure 7 shows both metric distributions. VideoSeek's and WorldMM's time differences skew heavily toward the larger buckets, with most retrieved timestamps landing far from the ground truth, while NEXUS and NEXUS+SCM concentrate in the smallest buckets, retrieving timestamps much closer to the ground truth. Frame similarity follows the same pattern: most frames retrieved by VideoSeek and WorldMM fall in the lowest similarity bucket, while NEXUS and NEXUS+SCM cluster toward the highest, retrieving visually similar frames. Overall, NEXUS-MF's persistent representation supports precise visual grounding beyond question answering, extending NEXUS-QA's advantage over both baselines to a distinct downstream task.

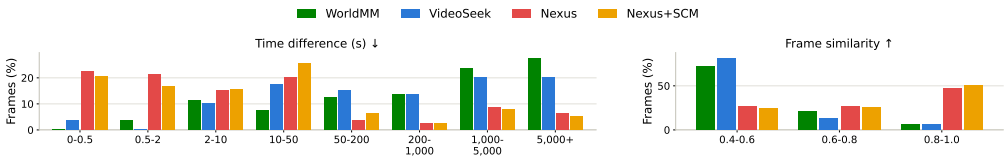


Figure 7: **Visual context-retrieval evaluation.** NEXUS and NEXUS+SCM concentrate in the smallest time-difference and highest similarity buckets, retrieving timestamps and frames much closer to the ground truth than either baseline.

6 CONCLUSION

We introduce NEXUS, a framework for long-form video understanding comprising three components: **NEXUS-MF**, which processes a video once into a persistent, causally structured representation of its characters and events, reusable across multiple downstream tasks; **NEXUS-OEBench**, a large-scale, open-ended benchmark demanding the depth of reasoning a human viewer would require; **NEXUS-QA**, which answers questions by reasoning over NEXUS-MF's representation rather than re-deriving evidence from raw video for every query. We hope NEXUS serves as a stepping stone toward more robust, open-ended, and rigorous evaluation of long-form video understanding.

AI USE STATEMENT

In this work, we used generative AI tools to help draft portions of this paper's text, to plot figures from raw data, and to generate a short illustrative video used in the main architecture figure. For NEXUS-OEBench, we additionally used generative AI tools to perform light grammar checking on annotator-written questions and answers, and to help filter out candidate questions and keypoints containing sensitive content. We did not use generative AI tools to write the benchmark's question-answer pairs or keypoints themselves, which were written manually by internal annotators (§3.2). We take responsibility for the final content of this work.

ETHICS STATEMENT

NEXUS-OEBench's questions and keypoints were intentionally filtered to exclude sensitive content, e.g., violence, sexual content, discrimination, self-harm, and harassment, among other similarly sensitive themes, as part of the annotation process described in §3.2.

NEXUS-MF's core capabilities – persistent identity tracking via audio-visual cues, relationship tracking, and event-causality extraction over long-form video – are general-purpose and not intrinsically tied to entertainment content; the same techniques could in principle be repurposed.

The dataset used in this research is an internal dataset, and proper permission was obtained for its use in research.

REFERENCES

Relja Arandjelović and Andrew Zisserman. Look, listen and learn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 609–617, 2017.

Ali Asgarov, Kaushik Narasimhan, Najibul Haque Sarker, Hani Alomari, Chia-Wei Tang, Anushka Sivakumar, Zaber Ibn Abdul Hakim, Shaurya Mallampati, and Chris Thomas. Nest: Narrative event structures in time for long video understanding. *arXiv preprint arXiv:2606.19706*, 2026.

Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2): 423–443, 2018.

Cong Chen, Guo Gan, Kaixiang Ji, ChaoYang Zhang, Zhen Yang, Guangming Yao, Hao Chen, Jingdong Chen, Yi Yuan, and Chunhua Shen. Memdreamer: Decoupling perception and reasoning for long video understanding via hierarchical graph memory and agentic retrieval mechanism. *arXiv preprint arXiv:2606.07512*, 2026.

Keith Curtis, George Awad, Shahzad Rajput, and Ian Soboroff. Hlvu: A new challenge to test deep understanding of movies the way humans do. In *Proceedings of the 2020 International Conference on Multimedia Retrieval (ICMR)*, 2020.

Gemini Team. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Yunseok Jang, Yale Song, Youngjae Yu, Youngjin Kim, and Gunhee Kim. Tgif-qa: Toward spatio-temporal reasoning in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Oleksii Kuchaiev, Jason Li, Huyen Nguyen, Oleksii Hrinchuk, Ryan Leary, Boris Ginsburg, Samuel Kriman, Stanislav Beliaev, Vitaly Lavrukhin, Jack Cook, Patrice Castonguay, Mariya Popova, Jocelyn Huang, and Jonathan M. Cohen. Nemo: a toolkit for building ai applications using neural modules. *arXiv preprint arXiv:1909.09577*, 2019.

Anna Kukleva, Makarand Tapaswi, and Ivan Laptev. Learning interactions and relationships between movie characters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9846–9855, 2020.

Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L. Berg. Tvqa: Localized, compositional video question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1369–1379, 2018.

Jie Lei, Tamara Berg, and Mohit Bansal. Revealing single frame bias for video-and-language learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, 2023.

Jingyang Lin, Jialian Wu, Jiang Liu, Ximeng Sun, Ze Wang, Xiaodong Yu, Jiebo Luo, Zicheng Liu, and Emad Barsoum. Videoseek: Long-horizon video agent with tool-guided seeking. *arXiv preprint arXiv:2603.20185*, 2026.

Olga Loginova, Oleksandr Bezrukov, Ravi Shekhar, and Alexey Kravets. Addressing blind guessing: Calibration of selection bias in multiple-choice question answering by video language models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025.

Karttikeya Mangalam, Raiymbek Akshulakov, and Jitendra Malik. Egoschema: A diagnostic benchmark for very long-form video language understanding. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023.

Arsha Nagrani, Mingda Zhang, Ramin Mehran, Rachel Hornung, Nitesh Bharadwaj Gundavarapu, Nilpa Jha, Austin Myers, Xingyi Zhou, Boqing Gong, Cordelia Schmid, Mikhail Sirotenko, Yukun Zhu, and Tobias Weyand. Neptune: The long orbit to benchmarking long video understanding. *arXiv preprint arXiv:2412.09582*, 2024.

Arsha Nagrani, Sachit Menon, Ahmet Iscen, Shyamal Buch, Ramin Mehran, Nilpa Jha, Anja Hauth, Yukun Zhu, Carl Vondrick, Mikhail Sirotenko, Cordelia Schmid, and Tobias Weyand. Minerva: Evaluating complex video reasoning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Tae Jin Park, Nithin Rao Koluguri, Jagadeesh Balam, and Boris Ginsburg. Multi-scale speaker diarization with dynamic scale weighting. In *Proceedings of Interspeech 2022*, pp. 5080–5084, 2022.

Judea Pearl. *Causality: Models, Reasoning and Inference*. Cambridge University Press, 2nd edition, 2009.

Junfu Pu, Yuxin Chen, Teng Wang, and Ying Shan. Omniscript: Towards audio-visual script generation for long-form cinematic video. *arXiv preprint arXiv:2604.11102*, 2026.

Rawal, Saifullah, Farré, Basri, Jacobs, Somepalli, and Goldstein. Cinepile: A long video question answering dataset and benchmark. *arXiv preprint arXiv:2405.08813*, 2024.

Joseph Roth, Sourish Chaudhuri, Ondrej Klejch, Radhika Marvin, Andrew Gallagher, Liat Kaver, Sharadh Ramaswamy, Arkadiusz Stopczynski, Cordelia Schmid, Zhonghua Xi, and Caroline Pantofaru. Ava active speaker: An audio-visual dataset for active speaker detection. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4492–4496, 2020.

Kaishuu Shinozaki-Conefrey, Olivier Pascaud, Robin Courant, Xi Wang, Dimitris Samaras, and Vicky Kalogeiton. Take 85: Testing audiovisual filmmaker's intent across 85 hours of film. *arXiv preprint arXiv:2608.30068*, 2026.

Enxin Song, Wenhao Chai, Guanhong Wang, Yucheng Zhang, Haoyang Zhou, Feiyang Wu, Haozhe Chi, Xun Guo, Tian Ye, Yanting Zhang, Yan Lu, Jenq-Neng Hwang, and Gaoang Wang. Moviechat: From dense token to sparse memory for long video understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18221–18232, 2024.

Tapaswi, Zhu, et al. Movieqa: Understanding stories in movies through question-answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Qiuyu Tian, Yiding Li, Fengyi Chen, Zequn Liu, Youyong Kong, Fan Guo, Yuyao Li, Jinjing Shen, Zhijing Xie, Yiyun Luo, and Xin Zhang. Stage: A full-screenplay benchmark for reasoning over evolving stories. *arXiv preprint arXiv:2601.08510*, 2026.

Lucas Ventura, Antoine Yang, Cordelia Schmid, and Gül Varol. Chapter-llama: Efficient chaptering in hour-long videos with llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18947–18958, 2025.

Paul Vicol, Makarand Tapaswi, Lluís Castrejón, and Sanja Fidler. Moviegraphs: Towards understanding human-centric situations from videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8581–8590, 2018.

Weihan Wang, Zehai He, Wenyi Hong, Yean Cheng, Xiaohan Zhang, Ji Qi, Xiaotao Gu, Shiyu Huang, Bin Xu, Yuxiao Dong, Ming Ding, and Jie Tang. Lvbench: An extreme long video understanding benchmark. *arXiv preprint arXiv:2406.08035*, 2024a.

Xiaohan Wang, Yuhui Zhang, Orr Zohar, and Serena Yeung-Levy. Videoagent: Long-form video understanding with large language model as agent. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2024b.

Zhengqian Wu, Ruizhe Li, Zijun Xu, Zhongyuan Wang, Chunxia Xiao, and Chao Liang. Friendsqa: A new large-scale deep video understanding dataset with fine-grained topic categorization for story videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

Junbin Xiao, Xindi Shang, Angela Yao, and Tat-Seng Chua. Next-qa: Next phase of question-answering to explaining temporal actions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Dejing Xu, Zhou Zhao, Jun Xiao, Fei Wu, Hanwang Zhang, Xiangnan He, and Yueting Zhuang. Video question answering via gradually refined attention over appearance and motion. In *Proceedings of the 25th ACM International Conference on Multimedia (ACM MM)*, 2017.

Jingkang Yang, Shuai Liu, Hongming Guo, Yuhao Dong, Xiamengwei Zhang, Sicheng Zhang, Pengyun Wang, Zitang Zhou, Binzhu Xie, Ziyue Wang, Bei Ouyang, Zhengyu Lin, Marco Cominelli, Zhongang Cai, Yuanhan Zhang, Peiyuan Zhang, Fangzhou Hong, Joerg Widmer, Francesco Gringoli, Lei Yang, Bo Li, and Ziwei Liu. Egolife: Towards egocentric life assistant. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 28885–28900, 2025.

Woongyeong Yeo, Kangsan Kim, Jaehong Yoon, and Sung Ju Hwang. Worldmm: Dynamic multimodal memory agent for long video reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25599–25609, June 2026.

Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. Activitynet-qa: A dataset for understanding complex web videos via question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 9127–9134, 2019.

Ce Zhang, Taixi Lu, Md Mohaiminul Islam, Ziyang Wang, Shoubin Yu, Mohit Bansal, and Gedas Bertasius. A simple llm framework for long-range video question-answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2024.

Junjie Zhou, Yan Shu, Bo Zhao, Boya Wu, Shitao Xiao, Xi Yang, Yongping Xiong, Bo Zhang, Tiejun Huang, and Zheng Liu. Mlvu: A comprehensive benchmark for multi-task long video understanding. *arXiv preprint arXiv:2406.04264*, 2024.

A REPRODUCIBILITY

Table 6 lists the model and reasoning effort used for each component of NEXUS-MF and NEXUS-QA.

Table 6: **Reproducibility details.** Model and reasoning effort used for each component of NEXUS.

Component	Model	Reasoning Effort
g_θ (chunking & captioning)	GPT-5.4-mini	low
f_θ (character assignment)	GPT-5.4-mini	low
h_θ (graph extraction)	GPT-5.4-mini	low
θ_{think} (QA thinking model)	Gemini 3.7 Flash	medium
θ_{tool} (QA tool-execution model)	Gemini 2.5 Flash	low

g_θ samples frames at 1 frame per 20 seconds for the *overview* and *detail* chunk summaries, and at frame pairs one second apart, sampled every 10 seconds, for the *script* summary.

VideoSeek and NEXUS-QA are both capped at 10 reasoning rounds (think-act-observe turns, Eq. 5) before being forced to answer.

The NEXUS-OEBench will be released upon acceptance.

B LIMITATIONS

We presented NEXUS as a causal-aware platform for long-form video understanding, with the ability to build a persistent representation of a movie once and reuse it across downstream tasks such as QA and context retrieval; however, we see the following limitations that open room for improvement:

- NEXUS-OEBench's recall metric relies on a panel of LLM evaluators and a reconciling judge (§4) rather than exhaustive human scoring of every answer.
- This recall metric credits a response for the ground-truth keypoints it substantiates, but does not measure precision: the LLM judges score the response against the keypoint list itself, not the video, so they cannot verify whether a claim in the response beyond the annotated keypoints is correct or incorrect.
- NEXUS-MF's preprocessing, while amortized across every downstream query on a video, is still a nontrivial one-time cost, which may not pay off for a use case that only ever asks a single question of a given video.
- NEXUS-OEBench spans eight genres but skews toward Drama and Fantasy (Appendix H).
- NEXUS-MF's chunk captioning (§3.1) is free-form and produced once, independent of any specific question; a summary written without foreknowledge of what will later be asked may omit a detail that a query-targeted pass over the same footage would catch.

C FUTURE WORK

NEXUS-MF's persistent understanding package $\mathcal{R}$ (Eq. 1) is built once per video and reusable across tasks; this work evaluates it on QA and the visual context-retrieval task alone. We see several natural directions for extending it further:

- Other downstream tasks mentioned in §1 but not yet evaluated – query-based summary generation, event or context localization, and recap video generation – would further test $\mathcal{R}$'s reusability.
- $\mathcal{R}$ is currently built once, offline, for a complete video; for a title released incrementally (§3.2's multi-version titles, e.g. a series adding new episodes over time), updating $\mathcal{R}$ incrementally as new content arrives, rather than reprocessing the full video from scratch, would avoid redundant computation.

- NEXUS+SCM's causal-role classifier ρ_θ (§3.3) currently guides evidence search at answer time; applying the same causal frame earlier, during NEXUS-MF's graph-extraction step (§3.1) itself, could produce a relation/event graph that is causally structured from the start, rather than causally reasoned over only afterward.

D ADDITIONAL RELATED WORK

D.1 QA AGENTS

VideoAgent (Wang et al., 2024b) uses a large language model as a central agent that iteratively identifies what information it still needs and calls vision-language foundation models as tools to retrieve and translate the relevant visual evidence, emphasizing interactive reasoning and planning over processing dense, uniformly-sampled frames. Its strength is efficiency at inference time: by retrieving only the visual evidence it judges necessary at each step, it avoids densely processing a video throughout. However, like VideoSeek (below), VideoAgent builds no persistent representation across queries, so each question restarts this retrieval loop from scratch rather than amortizing work across queries the way NEXUS-MF's representation does; and it is evaluated on EgoSchema and NExT-QA, both moderate-length video benchmarks rather than the multi-hour, narrative-driven video NEXUS-OEBench targets.

VideoSeek (Lin et al., 2026) reasons directly over sampled frames and CC for every query via a thinking/tool-performing model pair (§2), rather than committing to any precomputed representation. Its main strength is generality: with no upfront processing pass, it adapts to arbitrary queries without being constrained by what an earlier indexing step chose to keep. This comes at a structural cost, though: every question re-derives its evidence from raw video, so both token cost and wall-clock latency scale with how much of the video a question's evidence requires, and, per §4, with video length itself. On NEXUS-OEBench (Table 2, Figure 5), this shows up directly: VideoSeek's cost per question grows 2.0–3.4$\times$ as a title is cut into longer versions, versus only 1.3–1.7$\times$ for NEXUS, and its recall declines more steeply over the same range (e.g. 48.4 to 22.5 across one title's five versions). Nothing in VideoSeek's design amortizes this repeated re-derivation across questions asked of the same video, which is precisely the gap NEXUS-QA's persistent representation is built to close.

WorldMM (Yeo et al., 2026) instead builds three complementary memory stores (episodic, semantic, and visual; §2), and an adaptive retrieval agent selects which to draw from per query. Its strength is preserving visual detail that text-only memory summaries lose, while letting a query pull from whichever store and temporal granularity suits it rather than committing to one fixed representation.

MemDreamer (Chen et al., 2026) builds a structured memory graph from fixed-length video chunks, incrementally linking entities and events into a queryable graph that a downstream task can draw on (§2). Its strength is amortizing preprocessing across queries, similar in spirit to NEXUS-MF's persistent representation. However, MemDreamer's design and evaluation target question answering specifically, leaving its graph's reuse for other downstream tasks, such as NEXUS's context-retrieval task (§4), untested; and its fixed-length chunking (§3.1) risks splitting a single event or relationship-defining moment across a chunk boundary, a failure mode NEXUS-MF's context-adapted chunking is designed to avoid.

OmniScript (Pu et al., 2026) converts a video into a single, comprehensive script-like textual representation, fixed-length chunk by fixed-length chunk, capturing dialogue, action, and scene description in one pass, which downstream tasks then reason over as text. Its strength is simplicity: reducing retrieval to standard text search over one linear script. However, collapsing everything into one flat script sacrifices the character-level structure, relationship tracking, and event causality that NEXUS-MF's representation ($\hat{C}$, P_K, G_{rel}, G_{event}; Eq. 1) builds explicitly, information NEXUS-QA's toolkit (§3.3) depends on directly.

D.2 BENCHMARKS

NExT-QA (Xiao et al., 2021) uniquely poses the same questions in both MC and open-ended form, targeting causal and temporal reasoning, but its videos are brief clips of everyday activities rather than narrative content. MovieChat-1K (Song et al., 2024) leans further toward long-form and is

mostly open-ended, evaluating both whole-video and moment-specific understanding through separate global and breakpoint modes, though its videos remain far shorter than a full film or season. MLVU (Zhou et al., 2024) spans a wide range of video lengths and mixes MC with free-form generation tasks such as summarization and captioning, but its open-ended portion is a small slice of a broader task suite, and its overall scale is modest relative to the length range it targets. ActivityNet-QA (Yu et al., 2019) is large-scale, fully human-annotated, and open-ended, but its web videos are centered on everyday actions rather than characters or storylines. FriendsQA (Wu et al., 2025) draws on a real narrative series and organizes questions around fine-grained story topics such as character, action, and location, but its questions are generated by an LLM pipeline rather than human-annotated. Neptune (Nagrani et al., 2024) is one of the few benchmarks to combine MC and genuinely open-ended evaluation, with its own model-based metric for scoring free-form answers, but its videos still average only a few minutes despite an extended cap. Each long-form benchmark reaches long-form video at the cost of some other dimension. HLVU (Curtis et al., 2020) is closest in spirit to NEXUS-OEBench (full movie-length video with a knowledge-graph query format that probes character relationships directly rather than through option-selection) but its scale is under 1,000 questions across a handful of movies. LVBench (Wang et al., 2024a) reaches genuine hour-long video with a sizable, general-domain question set, but is MC only. EgoLifeQA (Yang et al., 2025) pushes furthest on video length, spanning tens of hours per participant stream, but its 500 questions are MC and framed around an egocentric personal assistant (recalling objects, habits, routines) rather than narrative or character understanding. MINERVA (Nagrani et al., 2025) pairs MC questions with hand-crafted reasoning traces that expose how a model arrives at an answer, but its videos average only 12 minutes despite an upper range of 1.5 hours. 1H-VideoQA (Gemini Team, 2024) was built specifically because prior benchmarks like EgoSchema top out at a few minutes per video, but its own scale (roughly 100 questions over 21 videos) makes it more an illustrative diagnostic than a large evaluation resource.

E NEXUS-MF CHARACTER-PROFILE DATABASE SCHEMA

Each profile $p \in P_K$ (§3.1, Eq. 3) is a record for one character with an `id`, `name`, current `roles` (job/occupation, one entry per context, e.g. a workplace), and current `relations` to other characters (`of_who`, and a free-text `relation` such as "mom" or "manager"). Two timelines track how the profile changes over the film: `role_timeline` logs every occupation change (e.g., hired, promoted, fired) and `relation_timeline` logs every change in a relation to another character (e.g., friend to estranged), each entry timestamped; `key_events` logs other narrative turning points (betrayed, injured, reconciled) that are not themselves a role or relation change.

The example below is a sanitized illustration (character names, roles, and timestamps are invented, not drawn from any NEXUS-OEBench title), showing one character's own role changing over time alongside their relation to a second character.

```
{
  "id": "char_1",
  "name": "John Doe",
  "sex": "male",
  "roles": [
    {"context": "org:unspecified", "role": "baking contest finalist",
     "uncertain": false}
  ],
  "relations": [
    {"of_who": "char_2", "relation": "teammate", "uncertain": false}
  ],
  "role_timeline": [
    {"context": "org:unspecified", "role": "contest newcomer",
     "timestamp": "00:05:10", "uncertain": false},
    {"context": "org:unspecified", "role": "baking contest finalist",
     "timestamp": "00:42:30", "uncertain": false}
  ],
  "relation_timeline": [
    {"of_who": "char_2", "relation": "rival",
     "timestamp": "00:06:45", "uncertain": false},
    {"of_who": "char_2", "relation": "teammate",
     "timestamp": "00:38:15", "uncertain": false}
  ],
  "key_events": [
    {"event": "Advances to the final round after a surprise dessert.",
     "timestamp": "00:42:30"},
    {"event": "Agrees to team up with Jane Doe for the final challenge.",
     "timestamp": "00:38:15"}
  ]
}
```

F NEXUS-MF RELATION AND EVENT GRAPH SCHEMA

Both G_{rel} and G_{event} (§3.1, Eq. 4) share the same nodes, one per resolved character (an `id` and display `name`). They differ in what an edge represents.

Each G_{rel} edge is a *relation* between two characters: a `type` (one of `family`, `romantic`, `professional`, `social`, `antagonistic`, `power`, `other`), a free-text `detail`, a `polarity` (`positive`/`negative`/`neutral`/`mixed`), a `confidence` level, whether it is `reciprocal`, an `evidence` list of timestamped, quoted CC lines supporting it, and a `dynamics` list recording each time the relationship's phase meaningfully shifts (e.g., colleagues to rivals), each phase itself backed by its own evidence. G_{event} shares the same character nodes but flattens every evidence entry, from every relation and every phase in its `dynamics`, into its own timestamped edge between the same two characters; where a G_{rel} edge summarizes a relationship, its G_{event} counterparts are the individual, chronologically ordered interactions it was built from.

The example below is a sanitized illustration (character names, dialogue, and timestamps are invented, not drawn from any NEXUS-OEBench title) showing one relation.

```
{
  "characters": [
    {"id": "char_1", "name": "John Doe"},
    {"id": "char_2", "name": "Jane Doe"}
  ],
  "relations": [
    {
      "from": "char_1", "to": "char_2",
      "type": "social",
      "detail": "rival bakers who team up for the final round",
      "polarity": "positive",
      "reciprocal": true,
      "confidence": "high",
      "evidence": [
        {"start": 2295.0, "end": 2301.5,
         "quote": "Let's just combine our recipes and split the win.",
         "source_cue_indices": [402]}
      ],
      "dynamics": [
        {
          "phase": "rivalry to teamwork",
          "start": 2290.0, "end": 2550.0,
          "summary": "John Doe and Jane Doe agree to team up
                      for the final challenge.",
          "evidence": [
            {"start": 2295.0, "end": 2301.5,
             "quote": "Let's just combine our recipes and split the win.",
             "source_cue_indices": [402]}
          ]
        }]
    }]
}
```

This same evidence entry also becomes a G_{event} edge: {`from`: `char_1`, `to`: `char_2`, `type`: `social`, `time`: 2295.0, `quote`: "Let's just combine our recipes and split the win.", `phase`: "rivalry to teamwork"}.

G STRUCTURAL CAUSAL MODELS

A Structural Causal Model (SCM) (Pearl, 2009) represents a system as a set of variables linked by directed edges, where an edge $X \to Y$ encodes that X is a direct cause of Y. A central lesson of causal inference is that two variables can be strongly correlated, or one can appear to explain the other, without one directly causing the other; Figure 8 illustrates the three structures underlying this distinction.

Direct cause. $X \to Y$: X directly causes Y.

Confounder. A confounder C causes both X and Y ($C \to X$ and $C \to Y$). X and Y are then correlated purely through their shared cause C, even if X has no causal effect on Y at all.

Mediator. A mediator M lies on the causal path from X to Y ($X \to M \to Y$): X does cause Y, but only indirectly, through M.

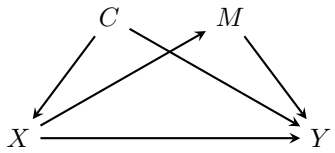


Figure 8: The three causal structures an SCM distinguishes between two variables X and Y: a direct cause ($X \to Y$), a confounder C that causes both ($C \to X$, $C \to Y$), and a mediator M on the causal path between them ($X \to M \to Y$).

Table 7: Genre composition and per-title question statistics of NEXUS-OEBench.

Genre	Titles	Context Sets	Total Qs	Mean Qs / Title	Mean Qs / Hour	Median Qs / Hour
Drama	5	12	893	178.6	13.47	12.90
Comedy	3	3	111	37.0	22.67	26.00
Fantasy	1	3	551	551.0	17.12	15.15
Horror	1	1	51	51.0	6.95	6.95
Thriller	1	1	32	32.0	17.05	17.05
Reality TV	1	1	26	26.0	2.71	2.71
Sci-Fi	1	1	25	25.0	5.12	5.12
Unknown	1	1	12	12.0	7.35	7.35
Overall	**14**	**23**	**1,701**	**121.5**	**13.92**	**13.54**

A confounder and a mediator can both produce an observed association between X and Y that is indistinguishable, at a glance, from a direct cause: correlation or co-occurrence alone cannot tell the three structures apart. Distinguishing them requires additional evidence, e.g., an intervention, a temporal ordering, or independent knowledge of the underlying mechanism, that rules out the alternative explanations.

The causal-role classification above is implemented as an instruction injected into NEXUS+SCM's thinking-model prompt at each turn, reproduced verbatim below.

```
## Causal Evidence Classification (SCM)
For every question, when gathering evidence for a conclusion Y,
classify each surrounding context you have found (p1, p2, p3,
... from search/tool results) by its causal role before
treating it as support for Y:
- Direct cause: p is itself the mechanism that produced Y (a
  stated action/reason immediately preceding and explaining Y).
- Mediator: p sits between an earlier cause and Y (X -> p -> Y)
  rather than being the root cause itself. Don't stop at p --
  look for X, the earlier trigger that operates through p, to
  confirm the full chain before citing p alone as sufficient.
- Confounder: p is a common cause of Y and of other things too,
  not specific evidence for Y's cause. A confounder that merely
  co-occurs with Y is not evidence for it -- do not cite it as
  if it were.
- Narrative-intent evidence: actively look for a planted
  detail, visual motif, or callback that signals the narrative
  was built toward Y -- this is likely one of the actual causes
  itself (a story often plants the very thing it later uses as
  the cause), not merely a writer's-intent signal separate from
  the real mechanism. Confirm it actually connects to Y through
  an action/statement in the story before citing it as the
  cause, but don't discount a candidate just because you first
  noticed it as a "setup" moment rather than an immediate
  trigger.
This classification organizes what you have gathered and points
to what to search for next (e.g. realizing p is a mediator
tells you to go find X).
```

H NEXUS-OEBENCH COMPOSITION

H.1 DIFFICULTY ANNOTATION

Difficulty is annotated by the same annotator who wrote the question, using two criteria. First, a question that must be answered by drawing on multiple question categories (Appendix H.2) at once – for instance, requiring both a *visual frame* detail and a *character dynamics* judgment – is rated more difficult than a single-category question. Second, a question that requires tracing a long-term causal dependency or a pattern across many scenes, rather than a fact available at one point in the video, is likewise rated more difficult.

H.2 QUESTION CATEGORIES

Each question is also tagged with one or more of seven categories describing the kind of understanding it requires. To ground all seven in one running example, consider the following fully invented scenario, not drawn from any NEXUS-OEBench title: across three seasons of a family-restaurant drama, John Doe inherits his late father's failing restaurant and secretly borrows money from a rival restaurateur, Richard Roe, to keep it open; he hides this loan from his sister and co-owner, Jane Doe, for two seasons, quietly selling off pieces of the business to cover repayments, until she discovers it in the Season 3 finale and confronts him, recontextualizing his guarded behavior across the entire series.

- *Plot* questions ask about factual story events rather than their interpretation, e.g., "What arrangement does John Doe make to keep the restaurant open after his father's death?". This is challenging when, as here, the show never states the event outright in one line: the loan surfaces only in fragments (a phone call, a signature, a delivery of cash) scattered across separate scenes that a model must piece together itself.
- *Character dynamics* questions concern how relationships between characters evolve or shift over the course of the film, e.g., "How does the relationship between John Doe and Jane Doe change from Season 1, when she trusts him to run the restaurant alone, to the Season 3 finale?". Answering this requires tracking their relationship across the entire series, not just one scene, since the loan and its consequences unfold gradually over three seasons before Jane learns of it, and the shift itself is never announced but must be inferred from many small changes in how the two interact.
- *Character psychology* questions ask what a character thinks, feels, or is motivated by at a given moment, e.g., "Why does John Doe avoid discussing the restaurant's finances with Jane Doe throughout Season 2?". This is challenging because the true motive (guilt over the hidden loan) is never spoken aloud; a model must infer it from indirect behavior and could easily latch onto a more obvious but wrong surface explanation, such as him simply being a private person.
- *Narrative intent* questions ask how to correctly interpret a scene (the writer's or director's purpose), e.g., "Why does the show repeatedly frame John Doe alone in the empty dining room after closing, across multiple seasons?". This requires recognizing a visual motif repeated across many episodes and connecting it to a symbolic meaning (his isolation and guilt) that is never mentioned in dialogue or captions at all, so no amount of transcript-only reasoning can recover it.
- *Visual frame* questions target a specific detail visible in a particular frame, whose answer lives in the image rather than the dialogue or plot, e.g., "What is visible inside the office desk drawer that John Doe quickly shuts when Jane walks in?". This is challenging because the detail must be found by *searching the video visually*, not by searching text: no CC or line of dialogue ever mentions it, so a model that only reasons over captions or transcripts has no signal to work with at all, and must instead look at the right frame itself.
- *Behavioral pattern* questions ask about an implicit pattern across scenes that must be inferred, or an action whose meaning only becomes clear later, e.g., "What does John Doe always do right before Richard Roe comes to the restaurant?". This is challenging because the answer is built from small, implicit signals in John's actions and emotions that never look directly relevant on their own: every time Richard is about to visit, John first

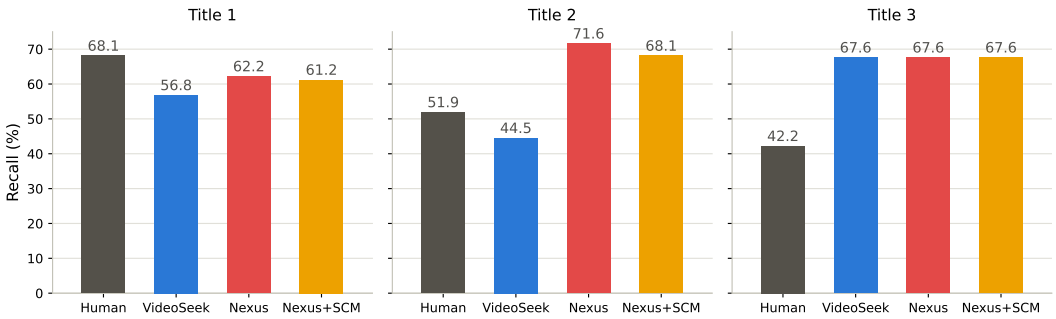


Figure 9: **Human evaluation on NEXUS-OEBench and comparison.**

sends Jane out on an invented errand, then grows visibly tense and short-tempered with the kitchen staff until Richard leaves. Any single occurrence reads as ordinary moodiness or an unremarkable scheduling coincidence; only after noticing the same combination of action and emotion repeat across a dozen separate episodes, and only once the Season 3 finale reveals the loan, does the pattern resolve into evidence that John has been hiding Richard's visits from Jane the entire time.

- *Who-said-what* questions ask which character said a given line, attributed from dialogue or audio, e.g., "Who says 'I didn't want you to lose the one thing Dad left us' during the Season 3 finale confrontation?". This can be challenging for multiple distinct reasons: the speaker may be off-screen at the moment the line is said, with no face on screen to match the voice to, or, conversely, John, Jane, and other family members may all be on screen at once talking over each other during the confrontation, so a model must correctly ground the line in the right voice or visual cue rather than guessing from whoever is simply on screen.

Categories are multi-label, so a single question may be tagged with more than one; the character-dynamics question above, for instance, could equally be tagged plot.

H.3 CAUSAL ("WHY") QUESTION TAXONOMY

To quantify how much of NEXUS-OEBench probes explicitly causal, "why"-style reasoning as opposed to factual recall, we tag each question by whether its text begins with "Why". 14.2% of questions (241 of 1,701) are why-questions; 83.0% of these also carry the *character psychology* and/or *narrative intent* category, the two categories that ask a model to infer a character's motive or a scene's interpretive intent rather than recall a fact, confirming that most explicitly causal questions target the deeper, motive- or intent-level reasoning that NEXUS-QA's SCM-guided evidence search (§3.3) is designed for. The remaining why-questions (17.0% of the 241) are causal in phrasing only, asking why a purely factual event occurred and tagged solely under categories such as plot.

I HUMAN EVALUATION

We additionally collect human answers for a subset of NEXUS-OEBench's titles and questions, scored with the same LLM-judge recall protocol used throughout §4.

J ADDITIONAL ABLATION RESULTS

Table 8 repeats the script-component and graph ablations from Table 5 with plain NEXUS in place of NEXUS+SCM, on the same subset of titles. The same pattern holds for the individual components: removing any single one lowers recall relative to the full script, and removing either graph lowers it further still.

Table 8: **Individual component and graph ablation on the same title subset as Table 5, with plain NEXUS.** The top group removes one individual component or graph at a time; the bottom row strengthens character-attributed CC, mirroring the pattern seen with NEXUS+SCM. Where shown, Standard deviation is computed over per-title recall across this subset's titles.

Variant	Recall (%) ↑	Tokens/Q ↓	Cost/Q ↓	Time (s) ↓
NEXUS (full)	**65.4** ± 8.4	**336.1K**	**$0.106**	**55.4**
– OCR	64.1 ± 8.5	342.6K	$0.111	50.4
– Where	62.8 ± 4.8	320.0K	$0.105	45.8
– How	64.4 ± 9.5	329.7K	$0.106	46.6
– Relation Graph	49.6 ± 16.9	539.8K	$0.174	45.8
– Event Graph	56.1 ± 11.1	433.1K	$0.147	48.8
NEXUS+ Reinforced CC	67.8 ± 6.2	381.4K	$0.112	52.8

K ALTERNATIVE BACKBONE: KIMI K3 THINKING MODEL

To test whether NEXUS-QA's advantage holds under a different thinking-model backbone, we replace θ_{think} (Gemini 3.7 Flash; Table 6) with Kimi K3, keeping the tool-execution model θ_{tool} (Gemini 2.5 Flash) unchanged, and re-run VideoSeek and both NEXUS configurations on the same four-title subset as Table 5. Table 9 shows the same ordering as with the default backbone: NEXUS+SCM achieves the highest recall and lowest token usage, followed by NEXUS, with VideoSeek trailing both by a wide margin.

Table 9: **Alternative backbone: Kimi K3 as the thinking model** θ_{think}, in place of Gemini 3.7 Flash, on the same four-title subset as Table 5. Standard deviation is computed over per-title values across this subset's titles. Overall, NEXUS in both configurations continues to outperform VideoSeek here as well, confirming that NEXUS's advantage over the baseline is consistent across backbone models.

Model	Recall (%) ↑	Tokens/Q (M) ↓
VideoSeek	55.4 ± 10.1	1.68 ± 1.76
NEXUS	64.5 ± 6.9	0.33 ± 0.10
NEXUS+SCM	**67.8** ± 6.4	**0.26** ± 0.09

L PUBLIC BENCHMARK EVALUATION SETUP

VideoSeek and both NEXUS configurations are each run once over the full set of included videos for LVBench and MINERVA-Ego (Table 4), using the same backbone models as NEXUS-OEBench (§4): Gemini 3.7 Flash as θ_{think} and Gemini 2.5 Flash as θ_{tool}.

For LVBench, we evaluate on 82 of the benchmark's 103 published videos; the remaining 21 are excluded because they are no longer available to download.

NEXUS+SCM costs slightly more tokens/Q than plain NEXUS on LVBench, unlike on NEXUS-OEBench and MINERVA-Ego: LVBench's questions demand comparatively little causal reasoning, so NEXUS+SCM's causal-role classifier rarely finds a confounder worth searching past, and its extra per-turn reasoning is not offset by the evidence-search savings seen on the more causally-demanding benchmarks.

M VISUAL CONTEXT-RETRIEVAL: FULL RESULTS

Table 10 gives the per-title mean/std summary behind Figure 7's distributions.

Table 10: **Visual context-retrieval evaluation.** Given a detailed description of a single frame, each engine retrieves that frame's timestamp, scored by time difference and frame similarity. Standard deviation is computed across titles.

Method	Time Difference (s) ↓	Frame Similarity ↑
WorldMM	1379.3 ± 538.1	0.578 ± 0.044
VideoSeek	957.0 ± 32.2	0.567 ± 0.012
NEXUS	432.5 ± 259.0	**0.799** ± 0.059
NEXUS+SCM	**401.7** ± 256.5	0.781 ± 0.091

```
You are building a visual retrieval benchmark. Given a single
frame from a movie/show, write one detailed description of
exactly this frame, about 100 words, covering: (1) the
setting/background, (2) who is visible -- describe each person
generically by appearance (e.g. "a middle-aged man in a dark
suit"); do not name anyone, since you have no source for names
other than this one image, and (3) what is happening / the
activity in the frame. Base the description ONLY on what is
visible in this frame -- no outside knowledge of the show/movie.
Do not mention timestamps or frame numbers. Respond as JSON:
{"description": "..."}
```

With the frame image attached and a fixed user turn, "Describe this frame." This description is then substituted into a fixed question template (the only templated field is the description itself) and posed to each engine under evaluation:

```
Here is a detailed description of a single frame from this video:

"<description>"

At what timestamp does this frame occur? Search the video and
give your single best estimate. End your answer with your
estimate on its own line in exactly this format:
TIMESTAMP: <seconds>
(e.g. TIMESTAMP: 842.5)
```

N RUN VARIANCE

Table 11: **Run-to-run variance on the 4-title ablation subset.** Mean and standard deviation across repeated runs of VideoSeek's `+Raw CC` configuration and NEXUS/NEXUS+SCM's `full` configuration (same subset and configurations as Tables 5–8), demonstrating low variance across runs and justifying the single-run numbers reported elsewhere in the paper.

Engine	Recall (%)	Tokens/Q	Cost/Q	Time/Q (s)
VideoSeek	53.9 ± 3.5	1.21M ± 0.63M	$0.301 ± 0.007	79.0 ± 4.6
NEXUS	65.1 ± 3.7	346.1K ± 12.5K	$0.113 ± 0.002	58.4 ± 3.2
NEXUS+SCM	65.4 ± 4.3	337.6K ± 29.7K	$0.105 ± 0.002	55.7 ± 3.2

O QUALITATIVE RESULT COMPARISON

The two examples below are drawn from an actual NEXUS-OEBench title, with only character names changed for anonymity: side characters repeatedly get the protagonist Alice's name wrong throughout the film.

Question (Plot, Character Dynamics): What are names that Alice was miscalled as?

Ground Truth: Alicia, Allison, Alyssa, Alina, Alexis, and Alys.

WorldMM's Answer: Alicia and Allison. **33% recall**

VideoSeek's Answer: Alicia, Alyssa, and Alina. **50% recall**

NEXUS's Answer: Alicia, Allison, Alyssa, Alina, Alexis, and Alys. **100% recall**

NEXUS+SCM's Answer and Causal Flow: Alicia, Allison, Alyssa, Alina, Alexis, and Alys. **100% recall** NEXUS+SCM first calls `search_script` with keywords describing Alice being called the wrong name, surfacing one early chunk where a coworker calls her "Alicia"; recognizing a single hit as an incomplete answer to an enumerative question, it issues further `search_script` and `get_event_range` calls across other chunks, surfacing "Allison," "Alyssa," "Alina," "Alexis," and "Alys" from separate scenes before answering.

A second example.

Question (Character Dynamics, Behavioral Pattern): Does Kate have any assistant in the office?

Ground Truth: Yes, Alice.

WorldMM's Answer: No. **0% recall**

VideoSeek's Answer: Yes, Thomas. **0% recall**

NEXUS's Answer: Yes, Alice. **100% recall**

NEXUS+SCM's Answer and Causal Flow: Yes, Alice. **100% recall** NEXUS+SCM's causal-role classifier first surfaces Thomas, who appears alongside Kate in one scene fetching coffee, but flags him as a confounder rather than her actual assistant since he never recurs in that role; it continues searching and finds Alice consistently handling Kate's scheduling and paperwork across multiple scenes, the pattern of an actual assistant, before answering.

Table 12: **Computation cost for the "Does Kate have any assistant in the office?" example.** Percentage tags on the NEXUS+SCM row are relative to plain NEXUS, ruling out the Thomas confounder without further wasted search reduces both time and total tokens.

Engine	Time (s)	Tokens In	Tokens Out	Tokens Total	Cost
VideoSeek	54.2	375.4K	3.4K	378.8K	$0.2632
NEXUS	43.1	221.1K	2.8K	223.9K	$0.1661
NEXUS+SCM	(-22.3%) **33.5**	(-11.5%) **195.7K**	(-25.0%) **2.1K**	(-11.7%) **197.8K**	(-10.4%) **$0.1489**

P TOOL-CALL DISTRIBUTION OVER VIDEO LENGTH

Figures 10 and 11 break down NEXUS-QA's tool calls by share (%) of each tool's use per question, against video length, across all 23 NEXUS-OEBench context sources, for plain NEXUS and NEXUS+SCM respectively. `get_script` and `search_script` dominate throughout, regardless of video length or configuration. As video length grows, NEXUS-QA leans increasingly on `search_script` to pinpoint evidence directly, while its reliance on broad `get_event_range` lookups recedes; `skim` and `focus` stay comparatively flat. This shift toward targeted search and away from broad scanning as context grows helps explain why NEXUS-QA's cost and latency grow only modestly with video length (§4), despite searching a persistent representation that itself grows with the video. The same trend holds within each multi-version title (thin connected lines in Figures 10 and 11): as later versions extend the same title to greater length, `search_script`'s share rises and `get_event_range`'s share falls, confirming the trend is driven by video length rather than by differences across titles.

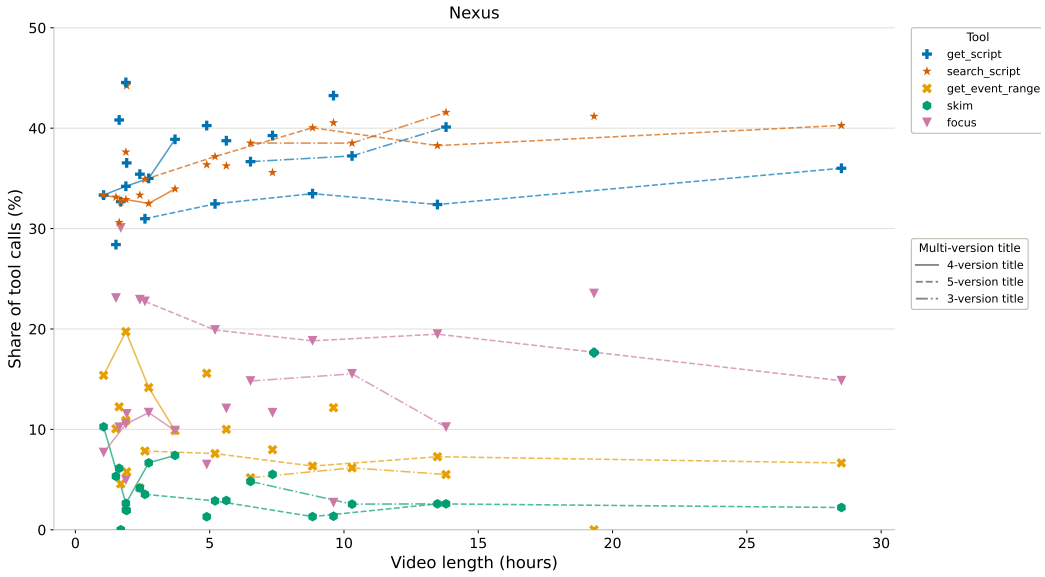


Figure 10: **Share of NEXUS-QA's tool calls (%) vs. video length, plain NEXUS.** Thin lines connect the versions of the same multi-version title. `get_script` and `search_script` dominate throughout, and `search_script`'s share rises with video length while `get_event_range`'s share falls.

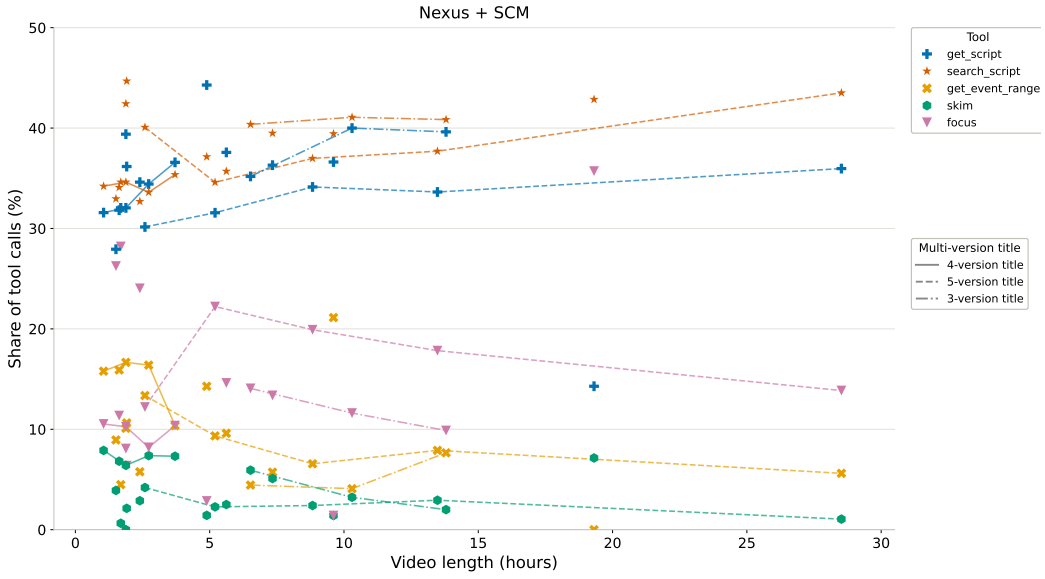


Figure 11: **Share of NEXUS-QA's tool calls (%) vs. video length, NEXUS+SCM.** Thin lines connect the versions of the same multi-version title. The same pattern as plain NEXUS (Figure 10) holds: `get_script` and `search_script` dominate, and `search_script`'s share rises with video length while `get_event_range`'s share falls.