

StoryTeller: Training-Free Narrative Grounding for Long-Form Audio Description

Seung Hyun Hahm, Minh T. Dinh, and SouYoung Jin

Dartmouth College, USA

{Seung.Hyun.Hahm.GR, Minh.T.Dinh.GR, SouYoung.Jin}@dartmouth.edu

Abstract. Long-form audio description (AD) requires more than describing visible actions: it must preserve characters, events, relationships, and story context across scenes so that blind and low-vision (BLV) audiences can follow a film. Modern video-language models (VLMs) are effective on short clips, but they often treat each moment independently, producing descriptions that miss who characters are, why events matter, and how the current scene connects to earlier narrative context. We propose StoryTeller, a training-free framework for story-aware long-form AD. Instead of relying only on local visual cues, StoryTeller maintains a verified narrative memory that carries forward story-relevant information across scenes, enabling later descriptions to remain coherent, grounded, and contextually informative. Given only raw video and a movie title, StoryTeller can optionally retrieve public movie metadata to resolve names and story context, while accepting only facts that are supported by the video through semantic filtering and VLM verification. The method requires no subtitles, scripts, AD transcripts, aligned captions, character banks, precomputed face identities, or task-specific fine-tuning. To evaluate whether generated AD preserves narrative information, we introduce StoryAD-QA¹, a question-answering benchmark that tests whether a language model can answer story-context questions using only the generated descriptions. Experiments on standard AD benchmarks and diverse long-form videos show that StoryTeller consistently improves narrative coherence, factual grounding, and story comprehension over strong baselines in automatic, QA-based, and human evaluations.

Keywords: audio description · long-form video understanding · retrieval-augmented generation · narrative grounding · accessibility

1 Introduction

Audio description (AD) enables blind and low-vision (BLV) audiences to access visual media by narrating important visual information between dialogue and other sounds. Effective AD goes beyond describing objects or actions in the current frame: it identifies who is present, explains what has just happened,

¹ **StoryAD-QA benchmark dataset and evaluation code:**
https://github.com/SEE-AI-Lab/ECCV2026_StoryTeller_StoryAD_QA

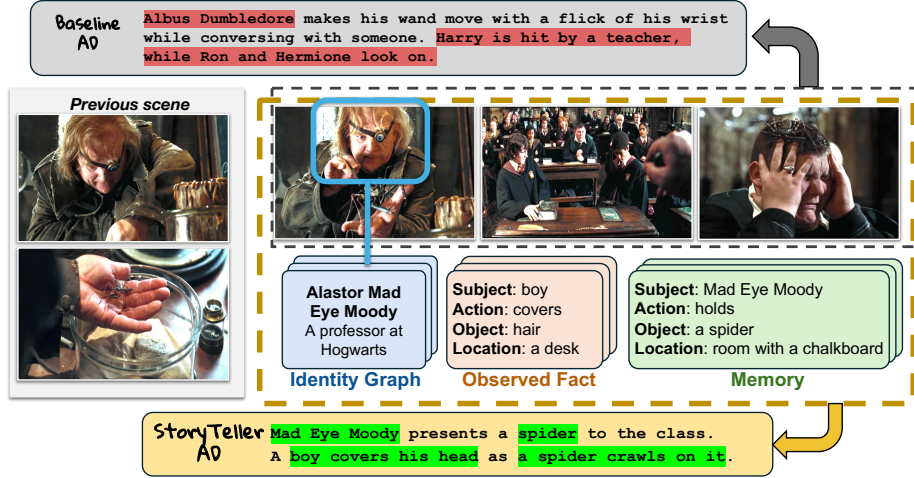


Fig. 1: Story-level audio description requires narrative memory. Existing audio description (AD) systems typically generate descriptions using only local visual context or static character banks, which often fails to preserve long-range narrative relationships. Our StoryTeller instead maintains a persistent narrative state consisting of an identity graph that tracks characters across scenes and a salience-weighted memory that accumulates narrative facts over time. This enables consistent character references and long-range story reasoning across an entire film. The baseline example is the output of AutoAD-Zero.

conveys why a moment matters, and connects the scene to the broader narrative. Professional AD writers are able to provide this narrative grounding because they watch the film and write with the whole story in mind. However, producing high-quality AD remains costly and time-intensive, and copyright restrictions make large publicly distributable AD datasets difficult to obtain.

Recent advances in video-language models (VLMs) [2,19,30,36] and large language models have motivated research on automatic AD generation. State-of-the-art identity-aware AD systems [8,9,27] are typically trained as task-specific models on curated movie datasets such as LSMDC [28] and MAD [31]. These pipelines incorporate identity-aware components built from face-recognition backbones trained on labeled data. While effective in benchmark settings, this reliance on copyrighted training data, curated movie annotations, and specialized identity modules limits their scalability, reproducibility, and applicability to new films.

Training-free pipelines, including AutoAD-Zero [38] and MM_Narrator [40], have recently emerged as an alternative to supervised AD generation. These methods remove the need for task-specific model training, but they often still rely on resources prepared before AD generation, including character banks, aligned subtitles, or existing AD transcripts. Moreover, they typically propagate context in shallow or static forms, such as concatenating previous descriptions or reusing fixed identity features. Although these strategies can improve local consistency across nearby clips, they do not explicitly model how narrative in-

formation evolves over time: which characters, relationships, and events should remain salient, and which details should fade as the film progresses.

Human viewers do not understand a film by retaining a complete list of previous frames. Instead, they maintain an evolving representation of the story, where central characters, unresolved conflicts, and recurring relationships remain salient, while incidental visual details gradually lose importance. This perspective suggests that long-form AD should preserve identity continuity, causal structure, and evolving character interactions across a film, without relying on external resources such as subtitles, scripts, AD transcripts, character banks, or precomputed face identities.

We introduce **StoryTeller**, a training-free framework for long-form AD that maintains an explicit story state as it processes clips in chronological order. Given a sequence of video clips $\{v_1, \dots, v_T\}$, the system keeps two forms of memory: an identity graph for recurring characters and a salience-weighted memory for verified story facts. Each new clip updates this state through three steps: track visible identities, extract candidate facts with optional public movie metadata, and verify those facts against the video before updating memory. The memory then reinforces facts that remain relevant and decays facts that no longer matter.

StoryTeller takes as input only the raw video and a movie title. The title may be used to retrieve public IMDb plot summaries, character lists, and short character descriptions. This retrieved text is only a source of possible names or context; it is not copied into the AD, stored directly as memory, or treated as ground truth. A fact enters memory only if it is supported by the current clip and accepted by semantic filtering and VLM verification.

Beyond generation, we also address evaluation. Standard metrics such as BLEU [25], CIDEr [34], and SPICE [1] measure lexical overlap between generated descriptions and reference captions, but they do not evaluate whether a listener can reconstruct the narrative from the AD alone. To address this limitation, we introduce **StoryAD-QA**, a narrative-comprehension question-answering benchmark. For each scene, we generate questions whose answers depend on both the current clip and preceding narrative context. During evaluation, a language model receives only the generated audio descriptions and must answer these questions without access to the video. This protocol directly measures whether the description preserves characters, events, and causal relationships necessary for story understanding rather than merely describing visual details in isolation.

Contributions.

- We propose StoryTeller, a fully training-free framework for long-form audio description that uses raw video and public title-keyed metadata without subtitles, AD transcripts, manually curated visual character banks, or task-specific fine-tuning.
- We introduce an explicit narrative memory mechanism with reinforcement-decay dynamics that approximates how human viewers track evolving story salience across scenes.

- We present StoryAD-QA, a narrative-comprehension QA benchmark that evaluates whether generated audio descriptions support story understanding beyond lexical similarity.

2 Related Work

Benchmarking Audio Description Generation. Movie understanding has often been studied as question answering over long videos [13, 32, 33, 35, 39]. For example, MovieQA [33] uses questions grounded in plot summaries, scripts, and subtitles, while [35] and [13] annotate characters, interactions, and story structure. MovieChat [32] emphasizes on long-video QA with sparse memory and introduces MovieChat-1K. These datasets are crucial for evaluating video reasoning, but they do not directly ask whether an AD gives a listener enough information to understand the story without seeing the video.

Datasets designed specifically for AD instead focus on localized narration. LSMDC [28] and MAD [31] provide large-scale movie captioning resources aligned to short clips, and recent AD generation systems [8–10, 26, 27, 38, 40] build on these datasets with identity-aware modules or curated character banks. Despite their usefulness, evaluation in these works often relies on direct string comparison with professional AD using metrics such as CIDEr [34], SPICE [1], and ROUGE [21], which are highly sensitive to temporal alignment. Addressing this limitation, AutoAD II [8] introduces an evaluation protocol that measures whether generated descriptions are semantically aligned with nearby ground-truth annotations rather than requiring exact sentence matches. Building on this direction, AutoAD III [10] further proposes CRITIC for evaluating character retrieval and LLM-AD-Eval for sentence-level semantic scoring with LLMs.

More recently, question-answering evaluation has been explored to assess the usefulness of AD. ADQA [15] evaluates whether generated AD supports visual appreciation and narrative understanding in a certain video segment via questions generated from reference descriptions. Instead, our benchmark focuses on long-form narrative grounding, evaluating whether ADs preserve information required for reasoning over story context that spans multiple scenes.

Identity Modeling in Video Narratives. Maintaining consistent character identity across scenes is essential for long-form video understanding and AD. Several recent AD systems explicitly rely on *character banks*—precomputed repositories of character-specific visual features, e.g., face embeddings or reference images used to recognize recurring characters throughout a movie [8–10]. In contrast, StoryTeller requires no character bank or precomputed face identities. Instead, it discovers recurring identities directly from repeated face observations, allowing character identities to emerge dynamically as narrative evidence accumulates.

Retrieval-Augmented Generation. Retrieval-augmented generation (RAG) improves grounding by incorporating external evidence during generation [5, 18]. However, naive retrieval can introduce irrelevant or contradictory information and may amplify hallucinations in multimodal models [12, 29]. Prior work improves retrieval quality through dense retrievers and structured fusion

mechanisms [7, 14, 16], but these approaches focus primarily on factual knowledge grounding rather than narrative coherence. Our approach instead performs narrative-aware retrieval restricted to movie-scoped sources and triggered only when scene observations indicate potentially story-relevant events.

Memory for Long-Horizon Reasoning. Memory mechanisms have been widely explored for long-horizon multimodal reasoning. Recent systems such as VideoAgent [4], Optimus-1 [20], and M3-Agent [23] maintain explicit temporal memories to support extended video reasoning. Other long-video systems employ episodic memories for VideoQA [37], retrieval-augmented memories for long-video comprehension [24], or global audio-visual character representations for dense video description [11]. These methods primarily target long-video QA, embodied reasoning, video comprehension, or dense video description. In contrast, StoryTeller is designed for accessibility-oriented audio description, where generated descriptions are intended to complement the original movie audio rather than summarize the entire film. Instead of maintaining a general-purpose memory, StoryTeller stores a lightweight narrative memory of visually verified story facts with evolving salience weights, allowing important characters and events to persist across scenes while transient details gradually fade.

3 StoryTeller

Figure 2 illustrates the overall pipeline of StoryTeller. The input is a movie divided into chronological *clips* $\{v_1, \dots, v_T\}$. StoryTeller processes the clips sequentially while carrying a persistent *narrative state* that summarizes story information accumulated from previously processed clips. This state enables the system to track recurring characters, extract scene-level events, and preserve long-range narrative context throughout the movie.

Before processing clip v_t , the system maintains the narrative state

$$\mathcal{S}_{t-1} = (\mathcal{G}_{t-1}, \mathcal{M}_{t-1}), \quad (1)$$

where $\mathcal{G}_{t-1}$ is the identity graph storing visual representations of recurring characters, and $\mathcal{M}_{t-1}$ is the narrative memory storing verified story facts together with their salience weights.

Processing clip v_t updates the narrative state according to

$$\mathcal{S}_t = \Phi(v_t, \mathcal{S}_{t-1}), \quad (2)$$

where Φ denotes the overall state update operator.

For each new clip, StoryTeller updates its narrative state through three modules. First, the *identity graph update* links visible faces to existing character nodes when there is sufficient visual evidence, allowing the system to maintain consistent references to recurring characters across clips. Second, *grounded fact induction* proposes structured facts about the current scene, optionally using public movie metadata as hints for names or story context. These candidate facts are then filtered to ensure that they are semantically relevant to the scene

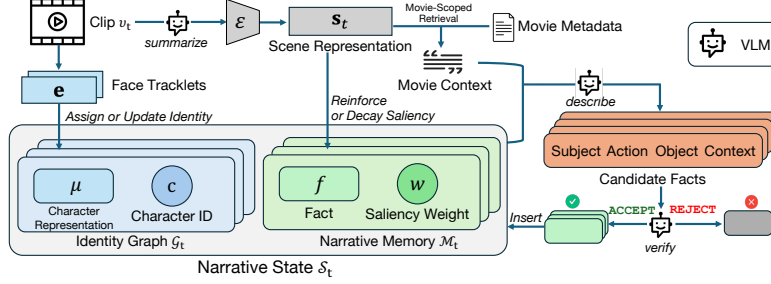


Fig. 2: Overview of StoryTeller. Before processing clip v_t , the system maintains the narrative state $S_{t-1} = (\mathcal{G}_{t-1}, \mathcal{M}_{t-1})$, where $\mathcal{G}_{t-1}$ is the identity graph and $\mathcal{M}_{t-1}$ is the narrative memory. Given v_t , StoryTeller (1) updates the identity graph to maintain character continuity, (2) extracts structured factual candidates using the updated identity graph, optional public movie metadata, and the previous narrative memory, and (3) updates the narrative memory using semantic filtering, VLM verification, and reinforcement-decay saliency dynamics. Public movie metadata may suggest character names or plot context, but only visually verified facts are stored in narrative memory.

and supported by the video. Third, the *saliency-based narrative memory* updates the importance of stored facts: facts that remain relevant are reinforced, while facts that no longer matter gradually decay. This allows important characters, events, and relationships to persist across scenes without carrying forward every transient detail.

Besides the visual clips, the movie title is used to retrieve optional public movie metadata. The retrieved metadata may suggest character names or plot context, but it cannot by itself create a memory entry or directly influence the generated narration. A fact is added to the narrative memory only if it is supported by the current clip and accepted by the semantic filtering and VLM verification steps.

3.1 Identity Graph Update

The identity graph $\mathcal{G}_{t-1}$ stores character representations accumulated from previously processed clips. Its role is to associate recurring appearances of the same person across clips, fostering consistent character references throughout the movie.

Unlike conventional character banks in the AD literature [10, 38], which store static character representations, our identity graph is updated continuously as new visual observations and verified narrative evidence become available. This allows recurring characters to be tracked consistently across scenes, while novel character identities can be established progressively as sufficient supporting evidence accumulates.

Tracklets and Embeddings. For each clip v_t , faces are detected in each frame and associated into short temporal tracklets $\mathcal{T}_k$, where each tracklet represents a single face observed over consecutive frames. A pretrained ArcFace encoder ϕ [3] extracts an embedding $\mathbf{v}_{t,i} = \phi(x_{t,i})$ for each face crop $x_{t,i}$. The embedding of

tracklet $\mathcal{T}_k$ is obtained by averaging the embeddings of all face observations in the tracklet:

$$\mathbf{e}_k = \frac{1}{|\mathcal{T}_k|} \sum_{i \in \mathcal{T}_k} \mathbf{v}_{t,i}. \quad (3)$$

Identity Graph Representation. The identity graph at time t is defined as

$$\mathcal{G}_t = \{(c_j, \boldsymbol{\mu}_j)\}_{j=1}^{N_t}, \quad (4)$$

where c_j denotes a character identity and $\boldsymbol{\mu}_j$ is the mean embedding of all tracklets assigned to that character. The graph therefore maintains a compact representation of characters observed up to time t .

Identity Assignment. For each new tracklet embedding $\mathbf{e}_k$, we compute its cosine similarity to every existing identity embedding:

$$j^* = \arg \max_j \cos(\mathbf{e}_k, \boldsymbol{\mu}_j). \quad (5)$$

If $\cos(\mathbf{e}_k, \boldsymbol{\mu}_{j^*}) > \tau$, the tracklet is assigned to identity c_{j^*} . Otherwise, a new identity node is created and the tracklet remains unnamed until sufficient evidence is accumulated to associate it with a character.

Identity Reseeding. Characters often appear before their names are revealed. Accordingly, every face tracklet is initially represented as an anonymous identity. The VLM may consult optional public movie metadata when proposing a character name, but the current clip must provide sufficient visual and narrative evidence before the association is accepted. Once verified, the corresponding anonymous identity is assigned the character name, and the identity graph is updated by creating a new named node or updating an existing one. Previously observed or future tracklets that are visually similar can then be associated with the same identity. Thus, the similarity threshold τ serves only as a candidate matching criterion rather than proof of identity. Tracklets that lack sufficient visual or narrative evidence remain anonymous until additional evidence becomes available.

3.2 Grounded Fact Induction

Given the current clip v_t , the updated identity graph $\mathcal{G}_t$ containing recurring character identities, and the narrative memory $\mathcal{M}_{t-1}$ summarizing previously accumulated story information, StoryTeller then inducts facts, i.e., extracts structured factual candidates and filters them before they are inserted into memory. The objective is to identify facts that are both visually supported and consistent with the evolving narrative context to achieve a new narrative memory $\mathcal{M}_t$.

Scene Representation. We first summarize the visual content of the clip. A VLM generates a concise scene summary s_t describing the primary event in v_t . The summary is then embedded as $\mathbf{s}_t = \varepsilon(s_t)$, where $\varepsilon(\cdot)$ denotes the text embedding function.

Public Movie Metadata Retrieval. Before fact induction, StoryTeller uses the movie title to retrieve optional public movie metadata, including plot summaries, character lists, and short character descriptions from IMDb. Each paragraph is embedded, and the paragraphs most similar to the current scene summary $\mathbf{s}_t$ are retrieved. The retrieved metadata serves only as auxiliary context: it may suggest character names or plot context, but it is never copied into the narration or directly stored in the narrative memory. All stored facts must be supported by the current clip and accepted by the semantic filtering and VLM verification steps.

If little or no metadata is available, retrieval simply returns fewer or no passages. The remainder of the pipeline is unchanged: facts are still extracted and verified, the identity graph and narrative memory are updated from visual evidence, and unnamed characters remain anonymous until sufficient evidence supports an identity assignment.

Structured Fact Extraction. Given clip v_t , the VLM proposes structured factual candidates conditioned on three sources of context: (i) recurring character identities from the identity graph $\mathcal{G}_t$, (ii) optional public movie metadata, and (iii) relevant narrative context retrieved from the memory $\mathcal{M}_{t-1}$. Each candidate is represented as

$$f = (\text{subject}, \text{action}, \text{object}, \text{context}),$$

where the subject may correspond to either a named character or an anonymous identity. The extracted candidates describe potential narrative updates, including character actions, interactions, object states, and scene context.

This stage is intentionally permissive: it generates hypotheses rather than committing facts to memory. Candidate facts, including proposed character names inferred from optional public movie metadata, are accepted only after the semantic filtering and VLM verification.

Semantic Filtering. The extracted candidates may include facts that are only weakly related to the current clip or inconsistent with the evolving narrative. Before applying computationally expensive VLM verification, we use a lightweight semantic filter to retain candidates that are either semantically related to the current scene or consistent with previously verified facts.

Each candidate fact f is embedded as $\mathbf{z}_f = \varepsilon(f)$. We compute

$$g_f = \max \left(\cos(\mathbf{z}_f, \mathbf{s}_t), \max_m \cos(\mathbf{z}_f, \varepsilon(f_m)) \right), \quad (6)$$

where f_m denotes the m -th fact stored in the narrative memory $\mathcal{M}_{t-1}$. The first term measures semantic similarity to the current scene summary, while the second measures consistency with previously verified facts. Candidate facts with $g_f > \sigma$ are forwarded to VLM verification. We use a conservative high-recall threshold for σ to discard only clearly unrelated candidates while retaining plausible facts for subsequent verification.

Verification Stage. Candidates that pass semantic filtering are evaluated by a dedicated VLM verifier using the current clip. The verifier outputs **ACCEPT** or **REJECT**. Only accepted facts are inserted into the narrative memory or used to

update named identities. This stage prevents semantic consistency from being mistaken for visual evidence: a candidate may agree with retrieved metadata or previously verified memory, but it cannot update the identity graph or narrative memory unless the video supports it.

3.3 Saliency-Based Narrative Memory

After verification, accepted facts are inserted into the narrative memory $\mathcal{M}_t$. Each memory entry consists of a validated fact and an associated saliency weight that determines its influence on future narration.

Formally, the memory at time t is

$$\mathcal{M}_t = \{(f_m, w_m^{(t)})\}, \quad (7)$$

where f_m is the m -th validated fact and $w_m^{(t)}$ is its saliency weight at time t . Accepted facts are inserted into the narrative memory with an initial saliency weight $w_m^{(0)} = 0.25$.

Relevance Computation. For the current scene summary s_t , we compute the semantic similarity between the scene embedding $\mathbf{s}_t$ and each stored fact embedding $\mathbf{z}_m = \varepsilon(f_m)$:

$$r_m^{(t)} = \cos(\mathbf{s}_t, \mathbf{z}_m). \quad (8)$$

Reinforcement–Decay Dynamics. The relevance score $r_m^{(t)}$ determines how strongly fact f_m should be reinforced. Memory weights evolve through reinforcement and decay: at each scene, every fact decays slightly, and facts relevant to the current scene are reinforced in proportion to their relevance:

$$w_m^{(t)} = \lambda w_m^{(t-1)} + \alpha r_m^{(t)}, \quad \lambda \in (0, 1). \quad (9)$$

Here λ is the decay factor, applied to the previous weight, and α is the reinforcement strength. A fact that stays relevant across scenes keeps gaining weight and remains salient, while a fact that stops being relevant ($r_m^{(t)} = 0$) loses a fixed fraction of its weight each scene and gradually fades. Supplementary Sec. S5 visualizes persistent and transient memory dynamics.

4 StoryAD-QA: Evaluating Narrative Comprehension

Existing captioning metrics such as CIDEr [34], SPICE [1], ROUGE [21], and BLEU [25] measure lexical or semantic similarity between generated and reference descriptions. While effective for evaluating caption quality at the scene level, they do not assess whether a sequence of audio descriptions preserves the information needed to understand a story. Narrative comprehension requires tracking characters, relationships, events, and causal developments across scenes, yet ADs are essentially concise and may omit important context. As a result, high captioning scores do not necessarily indicate strong story-level understanding. To

address this limitation, we introduce StoryAD-QA, a multiple-choice benchmark that evaluates whether generated audio descriptions retain sufficient information for downstream narrative reasoning. Rather than comparing generated descriptions against reference text, StoryAD-QA measures whether questions about the story can be answered using only the generated AD.

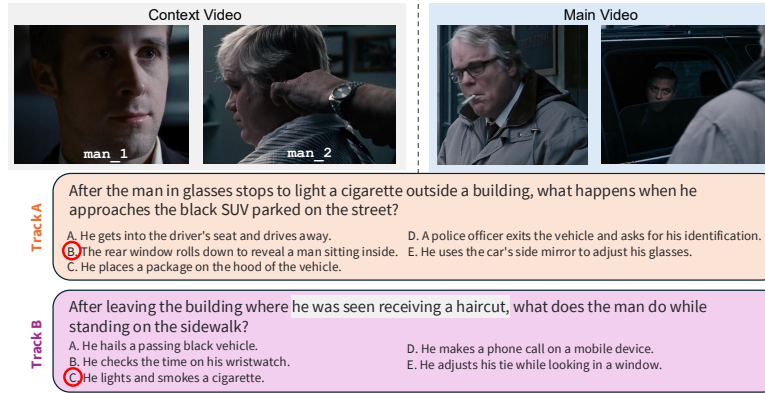


Fig. 3: Example of questions from two tracks in StoryAD-QA. Grey panels show frames from the preceding context window used in Track B for question generation. The context reveals that the man was receiving a haircut inside the building, enabling the question to reference this earlier event. Blue panels show the main clip, where the man stands outside the building and lights a cigarette, which determines the correct answer. During evaluation, only the AD of the main clip is provided, so the description must preserve or reintroduce the earlier context.

4.1 Benchmark Construction

StoryAD-QA is constructed from movie clips derived from the MAD-Eval [31] dataset. In total, the benchmark contains **2,574 questions** across two evaluation tracks: 1,611 in Track A (segment-only QA) and 963 in Track B (context-conditioned QA). See Supplementary Table S6 for additional dataset statistics.

Each example in StoryAD-QA consists of a video segment and, following prior work [15, 17], a multiple-choice question with five answer options, exactly one of which is correct. Questions are generated automatically using a vision-language model (VLM) that observes the original video along with short movie-level metadata describing the overall story context. The VLM is instructed to focus on clearly observable events or outcomes and to avoid relying on dialogue, external knowledge, or subjective interpretation. This ensures that answers are grounded in visual evidence rather than speculation. All answer choices are randomly shuffled before evaluation to reduce positional bias. During evaluation, the answering model receives only the AD text for the evaluated segment and the answer choices; it receives no video, retrieved IMDb/Wikipedia metadata, movie title, or dialogue transcript.

4.2 Evaluation Protocol

To evaluate whether generated AD preserves narrative information beyond individual frames, StoryAD-QA uses multiple-choice question answering over continuous movie segments. Questions are generated from video segments, but during evaluation an *answerer* (a Language Model) receives only the generated AD text, together with the question and answer choices, without access to the original video. Accuracy is the fraction of questions answered correctly from the AD text alone. We report two complementary evaluation tracks (See Table 3).

Track A: Segment-only QA. In Track A, questions are generated from a single continuous video segment of 30, 60, 120, or 240 seconds. During evaluation, the answering model receives the generated ADs for that same segment and selects the correct option. This track asks whether the AD captures the key events, entities, and relationships within the observed segment.

Track B: Context-conditioned QA. In Track B, each question is constructed from a fixed 30-second main clip together with its preceding context. The question generator observes the main clip plus $c \in 30, 60, 90$ seconds of video immediately before it. During evaluation, however, the answerer receives only the AD generated for the 30-second main clip, along with the question and answer choices; it does not receive the preceding context or the original video. This setting tests whether the main-clip AD preserves or reintroduces contextual information from earlier scenes, such as character identities, ongoing actions, and narrative relationships.

All questions retained in StoryAD-QA are manually verified before inclusion. Human annotators check that each question is visually grounded, has plausible distractors, and has a correct answer. Low-quality questions are regenerated and re-verified. Additional details are provided in Supplementary Sections S8.2–S8.3.

5 Experiments

We evaluate StoryTeller along three dimensions: (i) captioning quality on MAD-Eval, (ii) narrative comprehension with the proposed StoryAD-QA benchmark, and (iii) the contribution of individual pipeline components through ablation studies. All experiments are conducted without task-specific training, subtitles, scripts, AD transcripts, character banks, or precomputed face identities.

5.1 Experimental Setup

Datasets. We evaluate our method on MAD-Eval [31], which consists of 10 movies from the LSMDC [28] dataset. LSMDC contains 202 movies paired with professionally authored audio descriptions aligned to the corresponding video clips. To evaluate long-range narrative understanding, we further introduce and evaluate on StoryAD-QA, a question-answering benchmark constructed from non-overlapping segments of the MAD-Eval movies.

Implementation Details. We use Qwen3-VL-2B-Instruct [19] for scene summarization and structured fact extraction. We use Qwen3-VL-8B-Thinking for

fact verification, which outputs an ACCEPT/REJECT decision for each candidate fact. Text embeddings are computed with Qwen3-VL-Embedding-2B [19], and face embeddings are extracted with ArcFace [3]. Gemini3-Flash [6] is used only for StoryAD-QA question generation and answering. For question generation, Gemini3-Flash receives the video and a short Wikipedia overview. For answering, it receives only the AD text, question, and answer choices. IMDb retrieval is used only by the AD-generation pipeline and is defined in Section 3.2. Additional implementation details are provided in Supplementary Sec. S4, and the full prompts are provided in Supplementary Sec. S10.

Threshold Calibration. Identity assignment uses a fixed cosine similarity threshold of $\tau = 0.58$ for ArcFace embeddings. The semantic grounding threshold is set to $\sigma = 0.20$, calibrated from grounding-score statistics collected on separate calibration runs to provide a high-recall semantic filter that removes only clearly unrelated candidates before VLM verification. Both thresholds are fixed across all experiments and are not tuned on the evaluation data. Additional calibration statistics and threshold analyses are provided in Supplementary Sec. S4.2.

Runtime. On one A100-SXM4-80GB GPU, StoryTeller processes a feature-length movie in approximately 3–4 hours. The semantic filter is approximately $3,000\times$ faster than VLM verification, substantially reducing unnecessary verifier calls.

Memory Parameters. Narrative memory uses the reinforcement–decay mechanism described in Sec.3.3. All hyperparameters are fixed across experiments; complete values are provided in Supplementary Table S4.3.

5.2 MAD-Eval Benchmark Comparison

Table 1 compares StoryTeller with prior audio description systems while separating required resources from captioning performance. We additionally evaluate a variant without public movie metadata retrieval in the ablation study (Table 2).

We additionally evaluate StoryTeller on MovieChat [32] using 9 videos and 27 questions. Results are reported in Supplementary Sec. S7.

5.3 Ablation Study

To evaluate the contribution of each component in StoryTeller, we construct controlled ablation variants on MAD-Eval (Table 2). Each variant removes one module while keeping the remaining pipeline unchanged. All experiments use the Qwen3-VL backbone to isolate the contribution of the proposed components.

A1 (VLM-only narration). Direct VLM narration without structured fact extraction, identity tracking, verification, or narrative memory.

A2 (No schema). Replaces structured facts with scene summaries while preserving the sequential pipeline.

A3 (No memory). Removes cross-clip narrative memory while retaining per-clip fact extraction and verification.

A4 (No identity). Removes persistent identity tracking, leaving facts in generic form without consistent character grounding.

Table 1: Comparison on MAD-Eval. *Train* denotes whether task-specific training is required. *Curated Res.* includes subtitles, scripts, AD transcripts, aligned captions, character banks, or precomputed face identities. *Public Meta.* means public movie text retrieved by title, such as IMDb plot summaries, character lists, and short character descriptions. *VLM* indicates the underlying vision-language model used by each method. Our method requires neither additional training nor curated or precomputed movie resources; the default setting uses public movie metadata.

Model	Method Setup				Metrics		
	Train	Curated Res.	Public Meta.	VLM	CIDEr	SPICE	ROUGE-L
AutoAD-I [9]	✓	✓	×	-	14.3	4.4	11.9
AutoAD-II [8]	✓	✓	×	-	19.2	-	13.4
AutoAD-III [10]	✓	✓	×	-	24.0	-	-
MM-Vid [22]	×	✓	✓	GPT-4V	6.1	6.1	9.8
MM-Narrator [40]	×	✓	✓	GPT 4	13.9	5.2	13.4
AutoAD-Zero [38]	×	✓	✓	VideoLLaMA2	22.4	7.3	14.4
StoryTeller	×	×	✓	VideoLLaMA2	19.1	9.0	16.0
StoryTeller	×	×	✓	Qwen3-VL	21.4	6.7	15.3

Table 2: Ablation results on MAD-Eval. A1–A4 remove one internal module while keeping the remaining modules unchanged. A5 disables public IMDb metadata only; identity, schema, and memory remain active.

Variant	Public Meta.	Identity	Schema	Memory	CIDEr	SPICE	ROUGE-L
StoryTeller (Full)	✓	✓	✓	✓	21.4	6.7	15.3
A1 (VLM only)	×	×	×	×	11.4	4.3	9.6
A2 (No schema)	✓	✓	×	✓	15.4	4.4	12.5
A3 (No memory)	✓	✓	✓	×	18.0	6.0	13.2
A4 (No identity)	✓	×	✓	✓	17.9	5.6	12.3
A5 (No IMDb)	×	✓	✓	✓	17.2	4.8	12.2

A5 (No IMDb metadata). Disables public movie metadata retrieval while retaining identity tracking, structured facts, verification, and narrative memory.

Table 2 shows that each proposed component contributes to StoryTeller, with every ablation lowering MAD-Eval scores. Removing identity grounding (A4) leads to relative drops of 16.4% in CIDEr, 16.4% in SPICE, and 19.6% in ROUGE-L, even though public movie metadata remains available. This suggests that external metadata alone is insufficient for maintaining narrative consistency, and that persistent character resolution provides complementary information for coherent storytelling. A5 further isolates IMDb metadata: disabling it lowers CIDEr/SPICE/ROUGE-L by 19.6%/28.4%/20.3%, showing that public metadata helps but is not the sole driver of performance.

5.4 StoryAD-QA Narrative Evaluation

Table 3 reports results on the StoryAD-QA benchmark. In Track A, which evaluates understanding within a single video segment, StoryTeller consistently outperforms AutoAD-Zero across all segment lengths. The gains remain clear

Table 3: StoryAD-QA QA benchmark (Accuracy). (a) Segment-only QA. (b) Context-conditioned QA with c seconds of prior context plus a 30 s main clip; only the 30 s main-clip ADs are provided during evaluation. Reference AD denotes professional human-written AD text.

(a) Track A: Segment-only QA					(b) Track B: Context-conditioned QA			
Model	30	60	120	240	Model	30+30	60+30	90+30
Reference AD	0.953	0.963	0.995	0.981	Reference AD	0.828	0.825	0.806
AutoAD-Zero	0.840	0.830	0.896	0.917	AutoAD-Zero	0.662	0.654	0.673
StoryTeller	0.892	0.928	0.953	0.972	StoryTeller	0.754	0.716	0.788

for longer segments, suggesting that narrative-aware descriptions better capture story-level information beyond local clip content.

Track B evaluates contextual reasoning by generating questions using preceding context while providing only the ADs of the 30-second main clip during evaluation. Across all settings, StoryTeller achieves higher accuracy than AutoAD-Zero, with the largest improvement of +11.5 points in the 90+30 setting. These results indicate that the generated narration more effectively preserves contextual cues such as character identities and ongoing events, enabling the answering model to recover information originating from earlier clips.

The ‘‘Reference AD’’ row uses the professional AD text supplied with MAD-Eval as the answerer’s input. Because professional AD is scene-limited and may omit details needed for some generated questions, its accuracy is not necessarily 100%.

Table 4 presents component ablations on StoryAD-QA. Several simplified variants remain competitive on short or local settings, but the complete system achieves the best performance on the longest and most context-dependent settings: 240s in Track A and 90+30 in Track B. Removing narrative memory (A3), the identity graph (A4), or IMDb metadata (A5) reduces performance in these long-range settings, highlighting the role of each component. Notably, even without IMDb metadata, A5 remains competitive with or above AutoAD-Zero across all settings, indicating that the gains are not solely attributable to public movie information.

5.5 Qualitative Results

Figure 4 compares professional reference AD, AutoAD-Zero, and StoryTeller, illustrating differences in identity consistency and narrative recall across clips. Additional qualitative examples are provided in Supplementary Fig. S1 and Secs. S2–S3; human evaluation and preference results are reported in Sec. S9.

6 Conclusion

We introduced StoryTeller, a training-free framework for long-form audio description that uses narrative memory to capture evolving story context across

Table 4: Component ablations on StoryAD-QA accuracy. A1: VLM only; A2: no structured schema; A3: no narrative memory; A4: no identity graph; A5: no IMDb metadata.

Variant	Track A				Track B		
	30s	60s	120s	240s	30+30	60+30	90+30
Full	0.892	0.928	0.953	0.972	0.754	0.716	0.788
A1	0.908	0.904	0.928	0.911	0.747	0.750	0.751
A2	0.905	0.909	0.932	0.931	0.796	0.709	0.751
A3	0.899	0.907	0.918	0.891	0.756	0.712	0.779
A4	0.909	0.904	0.908	0.921	0.794	0.750	0.774
A5	0.892	0.909	0.908	0.901	0.761	0.716	0.770

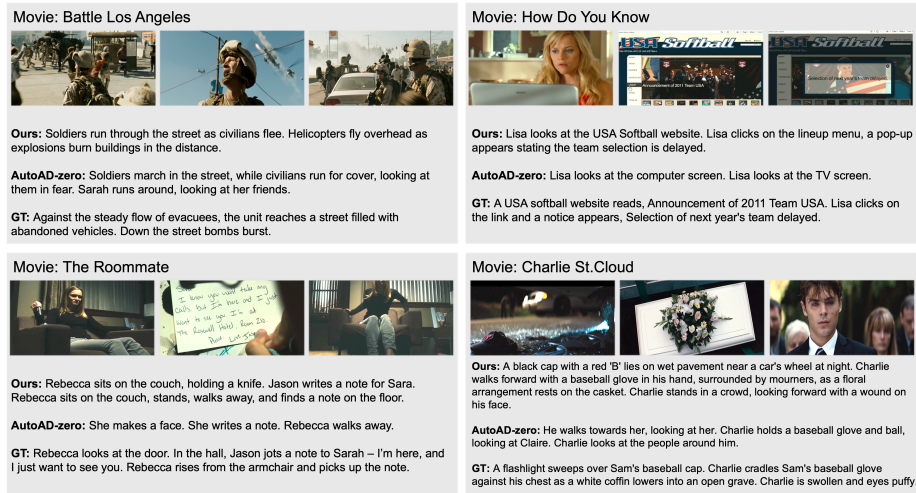


Fig. 4: Qualitative comparison of StoryTeller, AutoAD-Zero [38], and professional reference AD. (GT).

scenes. By maintaining an identity graph and structured narrative memory, the system generates coherent descriptions without task-specific training or curated movie resources; public movie metadata can suggest context, but only video-supported facts enter memory. We also introduced StoryAD-QA, a benchmark for evaluating whether generated AD supports narrative understanding. Experiments show that narrative-aware modeling improves contextual grounding and story comprehension in long-form video description.

Acknowledgments

This work was supported by startup funds provided by Dartmouth College. The authors also acknowledge support from the National Science Foundation under CAREER Award No. 2541968.

References

1. Anderson, P., Fernando, B., Johnson, M., Gould, S.: Spice: Semantic propositional image caption evaluation. In: European Conference on Computer Vision (ECCV). Springer (2016)
2. Cheng, Z., Leng, S., Zhang, H., Xin, Y., Li, X., Chen, G., Zhu, Y., Zhang, W., Luo, Z., Zhao, D., Bing, L.: Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms (2024)
3. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4690–4699 (2019)
4. Fan, Y., Ma, X., Wu, R., Du, Y., Li, J., Gao, Z., Li, Q.: Videoagent: A memory-augmented multimodal agent for video understanding. In: European Conference on Computer Vision (ECCV). pp. 75–92. Springer (2024)
5. Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., Wang, H.: Retrieval-augmented generation for large language models: A survey. arXiv preprint arXiv:2312.10997 (2023)
6. Google: Gemini 3 Flash Preview [large language model] (2026), <https://ai.google.dev/gemini-api/docs/models/gemini-3-flash-preview>, accessed: June 29, 2026
7. Guu, K., Lee, K., Tung, Z., Pasupat, P., Chang, M.: Retrieval augmented language model pre-training. In: Proceedings of the International Conference on Machine Learning (ICML). vol. 119, pp. 3929–3938. PMLR (2020), <https://proceedings.mlr.press/v119/guu20a.html>
8. Han, T., Bain, M., Nagrani, A., Varol, G., Xie, W., Zisserman, A.: AutoAD II: The Sequel - who, when, and what in movie audio description. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2023)
9. Han, T., Bain, M., Nagrani, A., Varol, G., Xie, W., Zisserman, A.: AutoAD: Movie description in context. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
10. Han, T., Bain, M., Nagrani, A., Varol, G., Xie, W., Zisserman, A.: Autoad iii: The prequel - back to the pixels. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 18164–18174 (June 2024)
11. He, Y., Lin, Y., Wu, J., Zhang, H., Zhang, Y., Le, R.: Storyteller: Improving long video description through global audio-visual character identification. arXiv preprint arXiv:2411.07076 (2024). <https://doi.org/10.48550/arXiv.2411.07076>
12. Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., Liu, T.: A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. ACM Transactions on Information Systems **1**(1) (January 2024). <https://doi.org/10.1145/3703155>
13. Huang, Q., Xiong, Y., Rao, A., Wang, J., Lin, D.: Movienet: A holistic dataset for movie understanding. In: European Conference on Computer Vision (ECCV) (2020)
14. Izacard, G., Grave, E.: Leveraging passage retrieval with generative models for open domain question answering. In: Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics (EACL). pp. 874–880 (2021)
15. Kala, D., Khandelwal, E., Tapaswi, M.: What you see is what you ask: Evaluating audio descriptions. In: Christodoulopoulos, C., Chakraborty, T., Rose, C., Peng, V. (eds.) Proceedings of the Conference on Empirical Methods in Natural Language

- Processing (EMNLP). pp. 23496–23518. Association for Computational Linguistics, Suzhou, China (Nov 2025). <https://doi.org/10.18653/v1/2025.emnlp-main.1199>, <https://aclanthology.org/2025.emnlp-main.1199/>
16. Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., Yih, W.t.: Dense passage retrieval for open-domain question answering. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 6769–6781. Association for Computational Linguistics (2020). <https://doi.org/10.18653/v1/2020.emnlp-main.550>, <https://aclanthology.org/2020.emnlp-main.550/>
 17. Lei, J., Yu, L., Bansal, M., Berg, T.: Tvqa: Localized, compositional video question answering. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1369–1379 (2018)
 18. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.t., Rocktäschel, T., Riedel, S., Kiela, D.: Retrieval-augmented generation for knowledge-intensive NLP tasks. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 33, pp. 9459–9474 (2020)
 19. Li, M., Zhang, Y., Long, D., Chen, K., Song, S., Bai, S., Yang, Z., Xie, P., Yang, A., Liu, D., Zhou, J., Lin, J.: Qwen3-vl-embedding and qwen3-vl-reranker: A unified framework for state-of-the-art multimodal retrieval and ranking. arXiv preprint arXiv:2601.04720 (2026)
 20. Li, Z., Xie, Y., Shao, R., Chen, G., Jiang, D., Nie, L.: Optimus-1: Hybrid multimodal memory empowered agents excel in long-horizon tasks. Advances in Neural Information Processing Systems (NeurIPS) **37**, 49881–49913 (2024)
 21. Lin, C.Y.: ROUGE: A package for automatic evaluation of summaries. In: Text Summarization Branches Out. pp. 74–81. Association for Computational Linguistics, Barcelona, Spain (Jul 2004), <https://aclanthology.org/W04-1013/>
 22. Lin, K., Ahmed, F., Li, L., Lin, C.C., Azarnasab, E., Yang, Z., Wang, J., Liang, L., Liu, Z., Lu, Y., Liu, C., Wang, L.: MM-VID: Advancing video understanding with gpt-4v(ision). arXiv preprint arXiv:2310.19773 (2023)
 23. Long, L., He, Y., Ye, W., Pan, Y., Lin, Y., Li, H., Zhao, J., Li, W.: Seeing, listening, remembering, and reasoning: A multimodal agent with long-term memory. arXiv preprint arXiv:2508.09736 (2025). <https://doi.org/10.48550/arXiv.2508.09736>
 24. Luo, Y., Zheng, X., Li, G., Yin, S., Lin, H., Fu, C., Huang, J., Ji, J., Chao, F., Luo, J., Ji, R.: Video-rag: Visually-aligned retrieval-augmented long video comprehension. Advances in Neural Information Processing Systems (NeurIPS) **38**, 168008–168033 (2026)
 25. Papineni, K., Roukos, S., Ward, T., Zhu, W.J.: Bleu: a method for automatic evaluation of machine translation. In: Isabelle, P., Charniak, E., Lin, D. (eds.) Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. pp. 311–318. Association for Computational Linguistics, Philadelphia, Pennsylvania, USA (Jul 2002). <https://doi.org/10.3115/1073083.1073135>, <https://aclanthology.org/P02-1040/>
 26. Park, J., Ye, J., Lee, S., Ka, H.W., Han, D.: Narrad: Automatic generation of audio descriptions for movies with rich narrative context. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 409–419. IEEE (2025)
 27. Raajesh, H., Desanur, N.R., Khan, Z., Tapaswi, M.: Micap: A unified model for identity-aware movie descriptions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)

28. Rohrbach, A., Torabi, A., Rohrbach, M., Tandon, N., Pal, C., Larochelle, H., Courville, A., Schiele, B.: Movie description. *International Journal of Computer Vision (IJCV)* **123**(1), 94–120 (2017). <https://doi.org/10.1007/s11263-016-0987-1>
29. Shao, H., Qian, S., Xiao, H., Song, G., Zong, Z., Wang, L., Liu, Y., Li, H.: Visual cot: Advancing multi-modal language models with a comprehensive dataset and benchmark for chain-of-thought reasoning. *Advances in Neural Information Processing Systems (NeurIPS)* **37**, 8612–8642 (2024)
30. Singh, A., Fry, A., Perelman, A., Tart, A., Ganesh, A., El-Kishky, A., McLaughlin, A., Low, A., Ostrow, A., Ananthram, A., Nathan, A., Luo, A., Helyar, A., Madry, A., Efremov, A., Spyra, A., Baker-Whitcomb, A., Beutel, A., Karpenko, A., Makelov, A., Neitz, A., Wei, A., Barr, A., Kirchmeyer, A., Ivanov, A., Christakis, A., Gillespie, A., Tam, A., Bennett, A., Wan, A., Huang, A., Sandjideh, A.M., Yang, A., Kumar, A., Saraiva, A., Vallone, A., Gheorghe, A., Garcia, A.G., Braunstein, A., Liu, A., Schmidt, A., Mereskin, A., Mishchenko, A., Applebaum, A., Rogerson, A., Rajan, A., Wei, A., Kotha, A., Srivastava, A., Agrawal, A., Vijayvergiya, A., Tyra, A., Nair, A., Nayak, A., Eggers, B., Ji, B., Hoover, B., Chen, B., Chen, B., Barak, B., Minaiev, B., Hao, B., Baker, B., Lightcap, B., McKinzie, B., Wang, B., Quinn, B., Fioca, B., Hsu, B., Yang, B., Yu, B., Zhang, B., Brenner, B., Zetino, C.R., Raymond, C., Lugaresi, C., Paz, C., Hudson, C., Whitney, C., Li, C., Chen, C., Cole, C., Voss, C., Ding, C., Shen, C., Huang, C., Colby, C., Hallacy, C., Koch, C., Lu, C., Kaplan, C., Kim, C., Minott-Henriques, C., Frey, C., Yu, C., Czarnecki, C., Reid, C., Wei, C., Decareaux, C., Scheau, C., Zhang, C., Forbes, C., Tang, D., Goldberg, D., Roberts, D., Palmie, D., Kappler, D., Levine, D., Wright, D., Leo, D., Lin, D., Robinson, D., Grabb, D., Chen, D., Lim, D., Salama, D., Bhattacharjee, D., Tsipras, D., Li, D., Yu, D., Strouse, D., Williams, D., Hunn, D., Bayes, E., Arbus, E., Akyurek, E., Le, E.Y., Widmann, E., Yani, E., Proehl, E., Sert, E., Cheung, E., Schwartz, E., Han, E., Jiang, E., Mitchell, E., Sigler, E., Wallace, E., Ritter, E., Kavanaugh, E., Mays, E., Nikishin, E., Li, F., Such, F.P., de Avila Belbute Peres, F., Raso, F., Bekerman, F., Tsimpourlas, F., Chantzis, F., Song, F., Zhang, F., Raila, G., McGrath, G., Briggs, G., Yang, G., Parascandolo, G., Chabot, G., Kim, G., Zhao, G., Valiant, G., Leclerc, G., Salman, H., Wang, H., Sheng, H., Jiang, H., Wang, H., Jin, H., Sikchi, H., Schmidt, H., Aspegren, H., Chen, H., Qiu, H., Lightman, H., Covert, I., Kivlichan, I., Silber, I., Sohl, I., Hammoud, I., Clavera, I., Lan, I., Akkaya, I., Kostrikov, I., Kofman, I., Etinger, I., Singal, I., Hehir, J., Huh, J., Pan, J., Wilczynski, J., Pachocki, J., Lee, J., Quinn, J., Kiros, J., Kalra, J., Samaroo, J., Wang, J., Wolfe, J., Chen, J., Wang, J., Harb, J., Han, J., Wang, J., Zhao, J., Chen, J., Yang, J., Tworek, J., Chand, J., Landon, J., Liang, J., Lin, J., Liu, J., Wang, J., Tang, J., Yin, J., Jang, J., Morris, J., Flynn, J., Ferstad, J., Heidecke, J., Fishbein, J., Hallman, J., Grant, J., Chien, J., Gordon, J., Park, J., Liss, J., Kraaijeveld, J., Guay, J., Mo, J., Lawson, J., McGrath, J., Vendrow, J., Jiao, J., Lee, J., Steele, J., Wang, J., Mao, J., Chen, K., Hayashi, K., Xiao, K., Salahi, K., Wu, K., Sekhri, K., Sharma, K., Singhal, K., Li, K., Nguyen, K., Gu-Lemberg, K., King, K., Liu, K., Stone, K., Yu, K., Ying, K., Georgiev, K., Lim, K., Tirumala, K., Miller, K., Ahmad, L., Lv, L., Clare, L., Fauconnet, L., Itow, L., Yang, L., Romaniuk, L., Anise, L., Byron, L., Pathak, L., Maksin, L., Lo, L., Ho, L., Jing, L., Wu, L., Xiong, L., Mamitsuka, L., Yang, L., McCallum, L., Held, L., Bourgeois, L., Engstrom, L., Kuhn, L., Feuvrier, L., Zhang, L., Switzer, L., Kondraciuk, L., Kaiser, L., Joglekar, M., Singh, M., Shah, M., Stratta, M., Williams, M., Chen, M., Sun, M., Cayton, M., Li, M., Zhang, M., Aljube, M.,

- Nichols, M., Haines, M., Schwarzer, M., Gupta, M., Shah, M., Guan, M.Y., Huang, M., Dong, M., Wang, M., Glaese, M., Carroll, M., Lampe, M., Malek, M., Sharmman, M., Zhang, M., Wang, M., Pokrass, M., Florian, M., Pavlov, M., Wang, M., Chen, M., Wang, M., Feng, M., Bavarian, M., Lin, M., Abdool, M., Rohaninejad, M., Soto, N., Staudacher, N., LaFontaine, N., Marwell, N., Liu, N., Preston, N., Turley, N., Ansmann, N., Blades, N., Pancha, N., Mikhaylin, N., Felix, N., Handa, N., Rai, N., Keskar, N., Brown, N., Nachum, O., Boiko, O., Murk, O., Watkins, O., Gleeson, O., Mishkin, P., Lesiewicz, P., Baltescu, P., Belov, P., Zhokhov, P., Pronin, P., Guo, P., Thacker, P., Liu, Q., Yuan, Q., Liu, Q., Dias, R., Puckett, R., Arora, R., Mullapudi, R.T., Gaon, R., Miyara, R., Song, R., Aggarwal, R., Marsan, R., Yemiru, R., Xiong, R., Kshirsagar, R., Nuttall, R., Tsiupa, R., Eldan, R., Wang, R., James, R., Ziv, R., Shu, R., Nigmatullin, R., Jain, S., Talaie, S., Altman, S., Arnesen, S., Toizer, S., Toyer, S., Miserendino, S., Agarwal, S., Yoo, S., Heon, S., Ethersmith, S., Grove, S., Taylor, S., Bubeck, S., Banerjee, S., Amdo, S., Zhao, S., Wu, S., Santurkar, S., Zhao, S., Chaudhuri, S.R., Krishnaswamy, S., Shuaiqi, Xia, Cheng, S., Anadkat, S., Fishman, S.P., Tobin, S., Fu, S., Jain, S., Mei, S., Egoian, S., Kim, S., Golden, S., Mah, S., Lin, S., Imm, S., Sharpe, S., Yadlowsky, S., Choudhry, S., Eum, S., Sanjeev, S., Khan, T., Stramer, T., Wang, T., Xin, T., Gogineni, T., Christianson, T., Sanders, T., Patwardhan, T., Degry, T., Shadwell, T., Fu, T., Gao, T., Garipov, T., Sriskandarajah, T., Sherbakov, T., Korbak, T., Kaftan, T., Hiratsuka, T., Wang, T., Song, T., Zhao, T., Peterson, T., Kharitonov, V., Chernova, V., Kosaraju, V., Kuo, V., Pong, V., Verma, V., Petrov, V., Jiang, W., Zhang, W., Zhou, W., Xie, W., Zhan, W., McCabe, W., DePue, W., Ellsworth, W., Bain, W., Thompson, W., Chen, X., Qi, X., Xiang, X., Shi, X., Dubois, Y., Yu, Y., Khakbaz, Y., Wu, Y., Qian, Y., Lee, Y.T., Chen, Y., Zhang, Y., Xiong, Y., Tian, Y., Cha, Y., Bai, Y., Yang, Y., Yuan, Y., Li, Y., Zhang, Y., Yang, Y., Jin, Y., Jiang, Y., Wang, Y., Wang, Y., Liu, Y., Stubenvoll, Z., Dou, Z., Wu, Z., Wang, Z.: Openai gpt-5 system card. arXiv preprint arXiv:2601.03267 (2025)
31. Soldan, M., Pardo, A., Alcázar, J.L., Caba, F., Zhao, C., Giancola, S., Ghanem, B.: Mad: A scalable dataset for language grounding in videos from movie audio descriptions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5026–5035 (June 2022)
 32. Song, E., Chai, W., Wang, G., Zhang, Y., Zhou, H., Wu, F., Chi, H., Guo, X., Ye, T., Zhang, Y., Lu, Y., Hwang, J.N., Wang, G.: 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 (June 2024)
 33. Tapaswi, M., Zhu, Y., Stiefel, R., Torralba, A., Urtasun, R., Fidler, S.: Movieqa: Understanding stories in movies through question-answering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4631–4640 (2016)
 34. Vedantam, R., Zitnick, C.L., Parikh, D.: CIDEr: Consensus-based image description evaluation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4566–4575 (2015)
 35. Vicol, P., Tapaswi, M., Castrejon, L., Fidler, S.: Moviegraphs: Towards understanding human-centric situations from videos. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2018). <https://doi.org/10.1109/CVPR.2018.00895>

36. Wang, W., Gao, Z., Gu, L., Pu, H., Cui, L., Wei, X., Liu, Z., Jing, L., Ye, S., Shao, J., Wang, Z., Chen, Z., Zhang, H., Yang, G., Wang, H., Wei, Q., Yin, J., Li, W., Cui, E., Chen, G., Ding, Z., Tian, C., Wu, Z., Xie, J., Li, Z., Yang, B., Duan, Y., Wang, X., Hou, Z., Hao, H., Zhang, T., Li, S., Zhao, X., Duan, H., Deng, N., Fu, B., He, Y., Wang, Y., He, C., Shi, B., He, J., Xiong, Y., Lv, H., Wu, L., Shao, W., Zhang, K., Deng, H., Qi, B., Ge, J., Guo, Q., Zhang, W., Zhang, S., Cao, M., Lin, J., Tang, K., Gao, J., Huang, H., Gu, Y., Lyu, C., Tang, H., Wang, R., Lv, H., Ouyang, W., Wang, L., Dou, M., Zhu, X., Lu, T., Lin, D., Dai, J., Su, W., Zhou, B., Chen, K., Qiao, Y., Wang, W., Luo, G.: Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. arXiv preprint arXiv:2508.18265 (2025)
37. Wang, Y., Zhang, L., Liu, J., Yan, J., Zhang, Z., Zheng, J., Ma, A., Ling, R., Yang, X., Wu, D., Chen, X., Li, X.: Video-em: Event-centric episodic memory for long-form video understanding. arXiv preprint arXiv:2508.09486 (2025). <https://doi.org/10.48550/arXiv.2508.09486>
38. Xie, J., Han, T., Bain, M., Nagrani, A., Varol, G., Xie, W., Zisserman, A.: AutoAD-Zero: A training-free framework for zero-shot audio description. In: Proceedings of the Asian Conference on Computer Vision (ACCV). pp. 2265–2281 (2024)
39. Yue, Z., Zhang, Q., Hu, A., Zhang, L., Wang, Z., Jin, Q.: Movie101: A new movie understanding benchmark. In: Proceedings of the Association for Computational Linguistics (ACL). pp. 4669–4684 (2023)
40. Zhang, C., Lin, K., Yang, Z., Wang, J., Li, L., Lin, C.C., Liu, Z., Wang, L.: Mm-narrator: Narrating long-form videos with multimodal in-context learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 13647–13657 (2024)

StoryTeller: Training-Free Narrative Grounding for Long-Form Audio Description –Supplementary Material–

Seung Hyun Hahm, Minh T. Dinh, and SouYoung Jin

Dartmouth College

S1 Overview

This supplementary document provides additional details and experimental results supporting the main paper. In particular, we include:

- Additional qualitative comparisons between StoryTeller, AutoAD-Zero, and professional reference AD.
- Failure cases of baseline methods and discussion of how StoryTeller improves narrative consistency.
- Analysis of narrative memory dynamics, including visualization of memory weights over time.
- A backbone comparison explaining how the VideoLLaMA2 and Qwen3-VL configurations differ across MAD-Eval and StoryAD-QA results.
- Further details and validation of the StoryAD-QA benchmark.
- Human evaluation of generated audio descriptions.
- Implementation details and hyperparameters used in the proposed framework.
- Full prompt templates used in the StoryTeller pipeline.

These materials aim to improve transparency and reproducibility of the proposed framework.

S2 Additional Qualitative Results

This section supports the qualitative analysis in Section 5.5 of the main paper by providing more examples beyond the main-paper figure.

Figure S1 presents additional qualitative comparisons between StoryTeller, AutoAD-Zero, and professional reference AD across four movies: *Harry Potter and the Goblet of Fire*, *The Ides of March*, *Roommate*, and *Legion*. Each example shows three representative frames together with the generated audio descriptions.

Across these examples, StoryTeller generally produces descriptions that include more concrete scene context visible in the frames, while AutoAD-Zero often focuses on shorter action-oriented phrases or more generic descriptions. The differences can be observed directly in the examples.



Fig. S1: Additional qualitative comparisons between StoryTeller, AutoAD-Zero, and professional reference AD across four movies. StoryTeller captures broader narrative context and scene semantics, while AutoAD-Zero often focuses on isolated actions or produces generic descriptions.

In *Harry Potter and the Goblet of Fire*, the frames show Harry fighting with the chained dragon. StoryTeller describes Harry leaping over a rocky cliff while being pursued by the dragon, capturing the spatial interaction between the character and the environment. AutoAD-Zero instead focuses on Harry casting a spell with his wand. Both descriptions refer to the same moment, but they emphasize different aspects of the scene.

The example from *The Ides of March* illustrates a scene with visible campaign signage and buildings in the background. StoryTeller describes the police vehicle and the political signs visible in the frames. In contrast, AutoAD-Zero produces a more generic description about officers walking toward a building, which does not explicitly mention the visual text present in the scene.

In the *Roommate* example, the first frame shows a character lying in bed, followed by outdoor campus scenes. StoryTeller describes both the indoor and outdoor settings shown across the frames. AutoAD-Zero produces a shorter description focused mainly on the sleeping character and nearby people walking.

Finally, the *Legion* example shows a car driving along a desert road while a dust storm forms in the distance. StoryTeller describes the car moving toward the storm and also mentions a small insect visible on the car vent in the final frame. AutoAD-Zero instead produces a more general description of the car driving and a person looking toward a dust cloud.

These examples highlight how the generated descriptions differ in content emphasis across methods. StoryTeller tends to include environmental details visible in the frames, while AutoAD-Zero often produces shorter descriptions centered on basic actions.

S3 Comparative Error Analysis

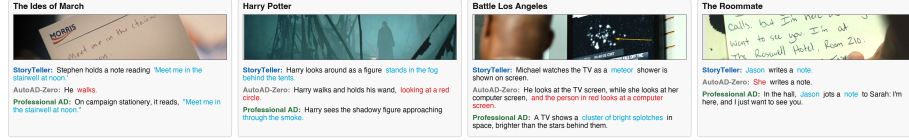


Fig. S2: Qualitative comparison of audio descriptions. Examples comparing StoryTeller, AutoAD-Zero, and professional reference AD across several movies. Text highlighted in blue corresponds to information that is visually grounded in the frames and appears in both StoryTeller and the professional reference AD. Text highlighted in red marks content that is less consistent with the visual evidence in the scene. These examples illustrate common challenges in automatic audio description, including grounding textual cues, interpreting scene entities, and maintaining character identity across scenes.

Figure S2 presents representative qualitative examples comparing StoryTeller, AutoAD-Zero [4], and professional reference AD across several movies. These examples highlight common challenges in automatic audio description, including difficulties in grounding textual cues, identifying scene entities, and maintaining character identity across scenes.

In the figure, text highlighted in blue corresponds to information that is consistent with the visual frames and appears in both StoryTeller and the professional reference AD. Text highlighted in red marks content that is less consistent with the visual evidence in the scene.

Missing textual or symbolic cues. In the *The Ides of March* example, the frame clearly shows a note containing the message “Meet me in the stairwell at noon.” StoryTeller and the professional reference AD both capture this textual information, while the baseline description instead focuses on a generic action (“He walks”), overlooking the written content visible in the scene.

Object grounding ambiguity. In the *Harry Potter and the Goblet of Fire* example, the scene shows a figure approaching through fog behind the tents. StoryTeller and the professional reference AD describe the approaching figure, whereas the baseline description refers to a “red circle,” which does not clearly correspond to any object visible in the scene.

Scene interpretation differences. The *Battle: Los Angeles* example illustrates a case where the baseline description interprets the television display as people interacting with computer screens. In contrast, StoryTeller and the professional reference AD identify the meteor shower shown on the display.

Character identity ambiguity. In the final example, the scene shows a character writing a note. StoryTeller correctly identifies the character as Jason and describes the action accordingly, while the baseline description refers to the character as “she,” which loses the identity information present in the narrative context.

These examples highlight common challenges in grounding audio descriptions to visual evidence and maintaining narrative consistency across scenes. By incorporating structured narrative facts and an evolving memory mechanism, StoryTeller is able to produce descriptions that better preserve scene context and character identity.

S4 Implementation Details

S4.1 Model Configuration

We use the following models in the StoryTeller pipeline:

- **VLM for scene summarization and fact extraction:** Qwen3-VL-2B-Instruct
- **VLM for fact verification:** Qwen3-VL-8B-Thinking
- **Text embedding model:** Qwen3-VL-Embedding-2B
- **VLM for StoryAD-QA generation and answering:** Gemini3-Flash for initial question generation and answering; Gemini 3.5 Flash is used only to regenerate questions that fail the first manual verification pass.
- **Face recognition encoder:** ArcFace [1]

Within each reported configuration, the same generation backbone is used consistently unless a table explicitly compares backbones. Scene-level observations are first summarized into structured facts, which are then verified and stored in the narrative memory. During generation, StoryTeller retrieves the most relevant verified facts and identity-consistent context to produce the final audio description. The AD pipeline may also retrieve IMDb plot summaries, character lists, and short character descriptions using the movie title. These texts can suggest possible names or plot context, but are not copied into narration or treated as professional reference AD; accepted facts must be supported by the clip.

Except for Gemini, all models are executed locally on a cluster comprising 8 NVIDIA A100-SXM4-80GB GPUs and 8 NVIDIA RTX 6000 Ada Generation GPUs. On one allocated A100-SXM4-80GB GPU, generating audio descriptions for a single film requires approximately 3–4 hours, corresponding to roughly 650 clips per movie. Across the 10 full movie runs, the pipeline processes 6,520 clips and 13,974 candidate facts; verification takes 3.68 ± 1.73 seconds per fact block, uses 473K tokens in total, and accepts 94.7% of candidate facts. The coarse semantic gate is approximately $3,000\times$ faster than VLM verification. Generating StoryAD-QA questions for the 10-movie evaluation set takes approximately 10–15 minutes per track.

In practice, generating the complete StoryAD-QA benchmark (2,574 questions across 10 movies) incurs an API cost of under \$20, while evaluating the complete benchmark costs less than \$5. These values are approximate and may vary depending on API pricing and the model’s internal reasoning-token usage.

Table S1: Hyperparameters used in StoryTeller.

Parameter	Value
Identity threshold τ	0.58
Grounding threshold σ	0.20
Reinforcement strength α	0.35
Decay factor λ	0.92
Initial memory weight $w_m^{(0)}$	0.25

Table S2: Effect of different semantic grounding thresholds. “Verifier calls saved” is measured relative to $\sigma = 0.20$.

σ	Calls saved	Accepted facts
0.20	0.0%	100.0%
0.40	1.4%	98.7%
0.45	6.1%	94.3%
0.50	19.3%	81.6%
0.55	43.2%	57.9%

For StoryAD-QA question generation only, Gemini3-Flash receives the video and a short Wikipedia movie overview. During question answering, it receives only the generated AD text, question, and answer choices—never the source video, movie title, IMDb or Wikipedia text, subtitles, or other retrieved text.

S4.2 Hyperparameters

Table S1 summarizes the hyperparameters used in StoryTeller. These values are fixed globally across all experiments and are not tuned on the evaluation datasets.

Grounding Threshold (σ). The semantic grounding threshold determines which candidate facts are forwarded to the computationally expensive VLM verification stage. To calibrate this threshold, we executed the complete StoryTeller pipeline on a separate set of movies and recorded the grounding score of every extracted candidate fact, producing approximately 2,700 candidate facts.

Among candidate facts ultimately accepted by the verifier, the minimum observed similarity was approximately 0.21, the 5th percentile was 0.36, and the median was 0.55. Based on these statistics, we selected $\sigma = 0.20$, slightly below the minimum observed similarity, so that the semantic filter functions as a high-recall gate. Its purpose is not to decide factual correctness, but to remove clearly unrelated candidates before VLM verification.

Table S2 reports the trade-off between computational cost and recall under alternative threshold values. Higher thresholds reduce the number of verifier calls but also discard visually supported facts. We therefore adopt the conservative setting $\sigma = 0.20$ to prioritize narrative recall over computational savings.

Table S3: Hyperparameter robustness on the test split. Performance remains stable under moderate variation. StoryAD-QA results are reported as accuracy (%) for Track A. The best performance for each metric is bolded.

Setting	CIDEr	StoryAD-QA Accuracy (%)			
		30s	60s	120s	240s
Default	21.4	89.2	92.8	95.3	97.2
α -20%	20.2	89.4	94.4	96.2	97.2
α +20%	21.0	90.5	94.4	97.2	98.2
λ -5%	21.9	89.7	94.2	95.3	97.2
λ +5%	21.8	90.1	94.0	94.8	96.3

Identity Threshold (τ). Identity assignment uses a fixed cosine similarity threshold of $\tau = 0.58$ for ArcFace embeddings. This value was selected empirically to balance missed associations and incorrect identity merges, and is kept fixed across all experiments.

Hyperparameter Robustness. Although StoryTeller introduces several global hyperparameters (identity threshold τ , grounding threshold σ , reinforcement strength α , and decay rate λ), these are fixed once and not tuned on evaluation data. To assess robustness, we vary α within $\pm 20\%$ and λ within $\pm 5\%$ while keeping the remaining parameters fixed. Performance on the test split remains stable, with StoryAD-QA accuracy varying by less than 3 percentage points and CIDEr by less than 6 percentage points. Results are summarized in Table S3.

S4.3 Memory Overhead of Identity Reseeding

Identity reseeding maintains a small backlog of unresolved (anonymous) tracklets until sufficient evidence is available for identity assignment. Each unresolved tracklet stores a single 512-dimensional ArcFace embedding (approximately 2KB). Across all movies in our experiments, retaining this anonymous backlog required less than 3.5MB per movie, indicating that delayed identity assignment introduces negligible memory overhead.

S4.4 Limitations and Discussion

StoryTeller maintains consistent character references using the identity graph described in Section 3.1 of the main paper which associates face tracklets across clips through clustering and similarity-based assignment. In rare cases, clustering errors may temporarily associate a tracklet with the wrong identity. When this occurs, the generated description may mention an incorrect character name for the corresponding scene.

Because StoryTeller maintains a persistent narrative state, such errors can often be corrected when stronger visual or narrative evidence becomes available.

In particular, the reseeding mechanism described in Section 3.1 of the main paper allows identities to be updated once additional observations clarify the character assignment. However, if incorrect associations persist across several clips, the resulting descriptions may temporarily reduce narrative accuracy.

Interestingly, this behavior also highlights the importance of maintaining an explicit narrative memory rather than relying solely on frame-level captioning. By storing and updating structured narrative facts over time, the system can revise earlier assumptions and maintain consistent story context as additional evidence appears. Future work may further improve robustness by integrating stronger identity verification or multimodal reasoning mechanisms.

S5 Analysis of Narrative Memory Dynamics

To better understand how StoryTeller maintains narrative context across long video sequences, we analyze the evolution of memory weights associated with verified narrative facts. Each fact f_m stored in the narrative memory is associated with a salience weight $w_m^{(t)}$ that reflects its current narrative importance. When later scenes provide visual or semantic evidence supporting f_m , the memory module reinforces its salience by increasing $w_m^{(t)}$. Conversely, if f_m is not observed again, its salience gradually decreases through decay.

Figure S3 visualizes representative trajectories from three movies: *Signs*, *Battle: Los Angeles*, and *Harry Potter and the Goblet of Fire*. For each movie we plot two facts: a persistent narrative cue (solid line) and a transient observation (dashed line). These trajectories illustrate how StoryTeller selectively retains story-relevant information while allowing short-lived visual details to fade from memory.

S5.1 Memory Weight Evolution

Figure S3 highlights two characteristic patterns of the memory update mechanism. Solid curves correspond to **persistent narrative facts** whose weights increase when similar visual evidence appears in later scenes. Dashed curves correspond to **transient observations** that occur only once and therefore decay when they are not reinforced. Star markers indicate scene indices selected for the qualitative examples shown in Fig. S4.

Signs. In *Signs*, the persistent fact corresponds to the interaction between the children and the family dog near the farmhouse porch. Because similar configurations recur across several scenes, the memory module repeatedly reinforces this fact, resulting in a steadily increasing weight. In contrast, a transient observation involving Bo grilling food outside appears only briefly and is never revisited, causing the associated weight to decay rapidly.

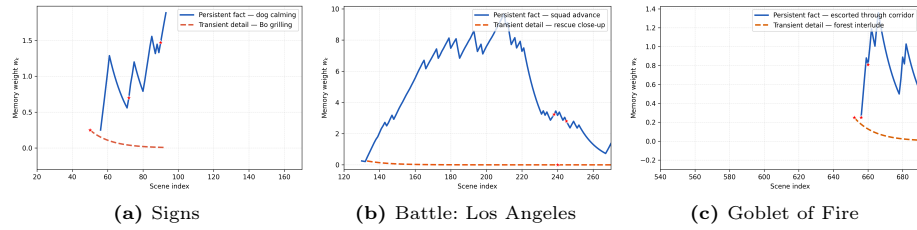


Fig. S3: Memory-weight trajectories $w_m^{(t)}$ for representative facts f_m extracted from *Signs*, *Battle: Los Angeles*, and *Harry Potter and the Goblet of Fire*. Solid curves denote **persistent narrative cues** that reappear across multiple scenes and therefore receive repeated reinforcement by the memory update mechanism. Dashed curves denote **transient observations** that occur only once and gradually decay when they are not reinforced. Star markers indicate the scene indices used in the qualitative case studies shown in Fig. S4, where the corresponding video frames and StoryTeller-generated audio descriptions are displayed.

Battle: Los Angeles. For *Battle: Los Angeles*, the persistent fact captures the squad advancing through a dark, smoke-filled environment while protecting civilians. Multiple neighboring scenes depict variations of this tactical situation, leading to repeated reinforcement and a sustained memory weight. By comparison, a brief close-up of a wounded soldier appears only once during the moment and quickly fades from memory.

Harry Potter and the Goblet of Fire. The third example contrasts a brief forest insert with a repeated corridor sequence. The persistent fact corresponds to Harry being escorted through a stone corridor by Moody. Each return to the hallway configuration reinforces the stored fact and increases its weight. The forest insert, however, functions only as a short atmospheric transition and therefore decays quickly.

S5.2 Case Study

While the weight trajectories reveal how narrative facts evolve in memory, the qualitative examples in Fig. S4 illustrate how this salience affects the generated audio descriptions.

Each panel presents representative frames together with the narration generated by StoryTeller. Words highlighted in red correspond to persistent narrative cues whose memory weights remain high across multiple scenes. Because these facts remain salient, StoryTeller continues referencing the same narrative context even when the camera framing shifts or visual details change.

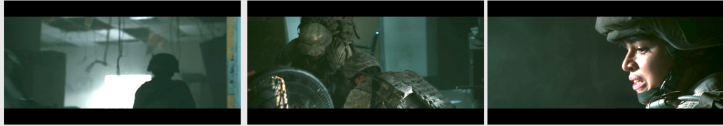
For example, in *Signs*, the generated description consistently refers to the interaction between the children and the dog despite changes in viewpoint. Similarly, in *Battle: Los Angeles*, the narration preserves the context of soldiers advancing through smoke-filled interiors while protecting civilians. In *Harry Potter*

Movie: Signs



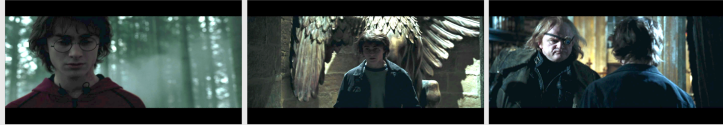
StoryTeller: Bo Hess grills **chicken** on the barbecue. The girl pours water into a bowl while the boy watches, **a dog standing nearby**. Bo Hess **approaches the dog**, which sniffs a bowl held by the girl.

Movie: Battle Los Angeles



StoryTeller: Soldier carries child through **dark room**. Soldiers advance through a **foggy room**, approaching civilians who flee in fear. The soldier speaks with intensity, her **face illuminated by a focused light**.

Movie: Harry Potter and the Goblet of Fire



StoryTeller: Harry stands in the **forest**. Harry walks through **a dim corridor** with winged statues. Harry stands in **a dim room**, watching Alastor Moody appear at the doorway with a wand.

Fig. S4: Representative frames corresponding to the facts highlighted in Fig. S3, together with StoryTeller-generated audio descriptions. Persistent narrative cues (highlighted in red) remain relevant across multiple scenes and are reinforced by the memory mechanism. Transient observations (highlighted in blue) appear only briefly and quickly decay.

and the Goblet of Fire, repeated corridor shots reinforce the interaction between Harry and Moody as they move through the dim stone corridor.

In contrast, words highlighted in blue correspond to transient visual details that appear only briefly. Because these observations are not reinforced by later scenes, their memory weights decay and they do not influence subsequent descriptions.

These examples demonstrate the intended behavior of the StoryTeller memory mechanism: facts tied to ongoing narrative context accumulate weight and remain retrievable across scenes, while isolated observations gradually fade. This selective retention enables the system to maintain story-level continuity without overloading the memory with short-lived details.

S6 Backbone Model Comparison

We compare StoryTeller with two generation backbones to separate architectural effects from backbone choice. The VideoLLaMA2 row corresponds to the StoryAD-QA setting reported in the main narrative-evaluation table, while the Qwen3-VL row corresponds to the default implementation used for the Qwen3-VL MAD-Eval result in the main comparison table.

Table S4: Comparison of different VLM backbones. StoryAD-QA results are reported as accuracy (%) for Track A (blue shade) and Track B (orange shade).

Backbone	CIDEr	StoryAD-QA Accuracy (%)						
		30s	60s	120s	240s	30+30	60+30	90+30
Qwen3-VL	21.4	89.2	92.8	95.3	97.2	75.4	71.6	78.8
VideoLLaMA2	19.1	85.9	90.2	92.9	95.4	72.7	70.9	72.4

S7 Additional Evaluation on MovieChat

We additionally evaluate StoryTeller on a subset of the MovieChat-1K test set [3], containing 9 videos and 27 questions. Table S5 compares methods under the input modality appropriate to each setting. MovieChat follows the standard MovieChat setup and accordingly receives the original source video (visual modality) as input. In contrast, AutoAD-Zero and StoryTeller are AD-generation systems, so we evaluate them by replacing the video with their generated AD text and asking the same LLM answerer to answer the MovieChat questions from text alone. This measures how much long-video QA information is preserved in the generated descriptions.

For a fair no-curved-resource comparison between AD systems, AutoAD-Zero is evaluated without its character bank, and StoryTeller is evaluated without public IMDb metadata. We also report StoryTeller with IMDb metadata to measure the additional benefit of public movie context. As shown in Table S5, StoryTeller outperforms AutoAD-Zero under the same generated-AD input setting, remains strong without IMDb metadata, and improves further when public metadata is available. StoryAD-QA remains our primary narrative benchmark because it provides denser coverage of story events, with approximately one question per 30s compared with three questions per movie in this MovieChat subset.

S8 StoryAD-QA Benchmark Overview and Validation

This section provides additional details about the construction and validation of the StoryAD-QA benchmark introduced in Section 4 of the main paper. We

Table S5: Additional evaluation on MovieChat The subset contains 9 videos and 27 questions. The subset contains 9 videos and 27 questions. MovieChat[†] uses the original source video, while AutoAD-Zero and StoryTeller use generated AD text as input to the same LLM answerer.

Method	Input	Acc.	Score
MovieChat	Source video	0.623	3.230
Human reference caption	Short caption	0.037	0.185
AutoAD-Zero (no character bank)	Generated AD	0.704	3.593
StoryTeller (no IMDb)	Generated AD	0.741	3.778
StoryTeller (with IMDb)	Generated AD	0.778	3.852

Table S6: Distribution of the 2,574 questions in StoryAD-QA across evaluation tracks and segment durations. Track A uses only the target segment, while Track B provides a 30, 60, or 90-second preceding context window plus a fixed 30-second target segment.

Track	Segment Duration	# Questions
Track A (segment-only QA)	30s	860
	60s	430
	120s	212
	240s	109
Track A Total		1,611
Track B (context-conditioned QA)	30s context + 30s	449
	60s context + 30s	294
	90s context + 30s	220
Track B Total		963

summarize the distribution of question types and segment durations, describe the manual verification protocol used for all retained questions, and report validation rates and annotation consistency.

S8.1 Benchmark Overview and Statistics

Table S6 summarizes the distribution of the final retained StoryAD-QA questions across the two evaluation tracks. Track A evaluates models using only the target segment, while Track B requires additional narrative context from preceding segments.

To capture events unfolding across multiple shots, we construct segments by grouping consecutive clips into temporal windows of varying duration. In Track A, questions are associated with segments ranging from 30 to 240 seconds. In Track B, each question is paired with a 30-second target segment and an additional context window preceding it, encouraging models to reason over narrative dependencies across segments.

The interface is divided into three main sections:

- Video:** A video player showing a scene from Harry Potter.
- Question:** A text box containing a question and five multiple-choice options (A-E). Below it is the 'Answer key: B' and a 'Rationale' section with a paragraph of text.
- Verification:** A series of checkboxes for evaluation criteria: 'Grounded', 'Plausible', 'Correct', and 'Minor fix needed'. Each has radio buttons for 'Yes', 'No', and 'Unknown'. There is also a 'Notes' text area and a 'Log verification' button at the bottom.

Fig. S5: Annotation interface used for human evaluation. Annotators watch the video segment, review the multiple-choice question and answer options, and record judgments for several verification criteria.

Answer Randomization. To prevent positional bias from the model generating the questions, the answer options are randomly shuffled before being provided to the language model during evaluation.

API Usage. Although the API safety filters were set to the least restrictive available configuration, a small number of AD inputs were still blocked by the service and could not be processed. These instances were treated as missing values and excluded from aggregate results.

S8.2 Manual Verification

All questions retained in StoryAD-QA are manually verified before inclusion. Each question is reviewed through a custom *Gradio* annotation interface, which presents annotators with the corresponding video segment together with the generated question, answer options, the labeled correct answer, and the accompanying rationale produced during question generation. Figure S5 demonstrates the interface from where annotators watch the corresponding video segment and the corresponding generated questions and submit their evaluation of several quality metrics designed to assess the integrity of the question-answer pairs. The review covers both the question and its answer set: annotators check that the question is visually grounded and unambiguous, that the labeled answer is uniquely supported by the video evidence, and that the distractor options are plausible but incorrect. Questions that fail verification are discarded or regenerated and then checked again before they can enter the final benchmark.

Verification criteria We use **Correct answer** to assess whether the labeled answer is correct and uniquely supported by the visual evidence in the clip. In addition, we adopt two quality metrics from [2]. **Visually grounded** evaluates whether the correct answer can be determined solely from the visual content of the clip, without relying on dialogue, audio, or external knowledge. **Plausible**

Table S7: Human verification results on all questions of the initial StoryAD-QA benchmark questions. Rates report the proportion of questions satisfying each criterion.

Setting	Grounded	Plausible Distractors	Correct Ans.
<i>Track A</i>			
30 s	0.890	0.857	0.752
60 s	0.901	0.875	0.795
120 s	0.892	0.887	0.814
240 s	0.963	0.936	0.833
<i>Track B</i>			
60 s	0.838	0.827	0.744
90 s	0.822	0.784	0.703
120 s	0.886	0.831	0.699
Unsure	3	3	45

distractors assesses whether the incorrect answer options constitute realistic alternatives that could reasonably be confused with the correct answer given the scene.

In addition to the binary *Yes/No* labels, annotators could also select *Unsure* when the judgment was genuinely ambiguous. The verification statistics for the full benchmark are presented in Table Tab. S7.

S8.3 Question Regeneration

Questions that received a negative rating in any evaluation criterion during the initial human verification were discarded and regenerated using the same video clip and prompt, but with a stronger VLM, namely Gemini 3.5 Flash. In total, we regenerated 773 questions, of which 618 were retained after filtering out clips that the VLM determined lacked sufficient information to support a meaningful question. All regenerated questions then underwent a second round of manual verification by a human annotator. Every question in this pass was accepted and included in the final benchmark.

S9 Human Evaluation of Audio Descriptions

To evaluate the perceptual quality of generated audio descriptions, we conducted a human study comparing StoryTeller with AutoAD-Zero.

S9.1 Study Setup

We randomly sampled 50 clips from the evaluation set, each with an average duration of approximately 20 seconds. For each clip, three annotators watched the video and compared two candidate audio descriptions: one generated by StoryTeller and one generated by AutoAD-Zero. The descriptions were presented with

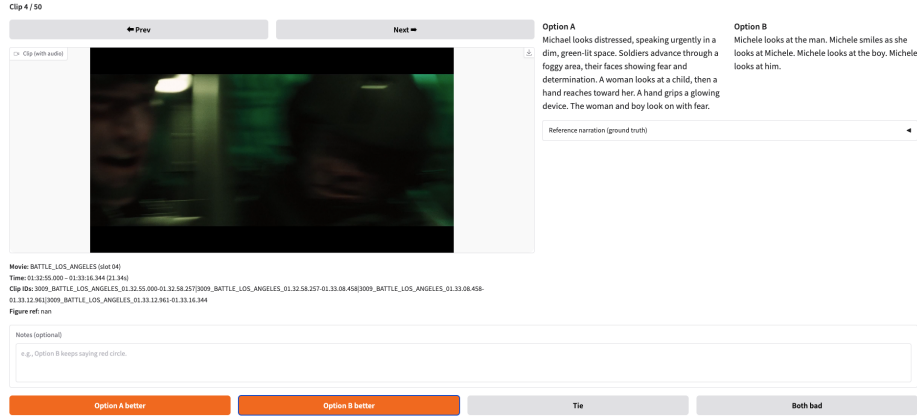


Fig. S6: Human evaluation interface used in the study. Annotators watch a video clip and compare two candidate audio descriptions (Option A and Option B), whose order is randomized per trial. They select whether one description is better, whether the two are comparable (*Tie*), or whether both descriptions are inadequate.

anonymized labels (*Option A* and *Option B*), and their order was randomized for each trial to avoid positional bias. Because each clip spans approximately 20 seconds, the candidate descriptions are formed by concatenating multiple consecutive AD sentences generated for that clip, resulting in short multi-sentence descriptions.

Figure S6 illustrates the evaluation interface. Annotators could replay the clip and optionally view the reference audio narration (ground truth) for additional context, but no system identifiers or auxiliary metadata were shown in order to maintain a blind evaluation.

For each clip, annotators selected one of four options:

- **Option A better:** description A provides a better audio description.
- **Option B better:** description B provides a better audio description.
- **Tie:** both descriptions are of comparable quality.
- **Both bad:** neither description adequately describes the scene.

Annotators were instructed to judge descriptions based on two criteria: (i) how accurately the description reflects the visible content of the clip, and (ii) whether the narration conveys the scene clearly and coherently for a blind or low-vision audience. The study involved three human evaluators, each rating all 50 clips.

S9.2 Results

Table S8 summarizes the human preference results. Across the 50 evaluated clips, annotators preferred StoryTeller substantially more often than AutoAD-Zero, with StoryTeller receiving 76.0% of the preferences compared to 5.8% for

Table S8: Human preference comparison between StoryTeller and AutoAD-Zero.

Method	Preference (%)
StoryTeller	76.0
AutoAD-Zero	5.8
Tie	9.1
Both bad	9.1

the baseline while 9.1% of the evaluations resulted in a tie and 9.1% judged both descriptions inadequate.

In many cases, annotators selected StoryTeller when its descriptions captured scene context or environmental cues visible in the frames, whereas AutoAD-Zero descriptions tended to focus on shorter or more generic actions.

S10 Prompt Templates

This section lists the prompt templates used in the StoryTeller pipeline, organized according to the modules described in Sec. 3 of the main paper: scene summarization, structured fact extraction, fact verification, and narration generation. The prompts guide the model to distinguish observed evidence from auxiliary cues such as public movie metadata. In particular, the fact-extraction schema records whether each candidate fact is suggested by visual, audio, or metadata input. Metadata may help propose candidates, such as possible character names, but these candidates are passed through a separate verification step before being added to memory or used in narration.

S10.1 Scene Summarization Prompt

Stage 1: Scene Summarization. This prompt generates a concise summary of the visual content of a single video clip. The goal of this stage is to produce a short, grounded description that captures the main observable event without introducing speculative interpretation. The summary serves as the initial representation of the scene and is later expanded into structured narrative facts.

Instruction. You create concise, evidence-grounded clip summaries for audio description. Rules: present tense; one sentence; less than 25 words; describe only what is clearly visible or audible; no speculation, no spoilers; if identity is unclear, use neutral labels.

Prompt. Summarize the clip in one concise sentence, describing only what is clearly visible or audible.

S10.2 Structured Fact Extraction Prompt

Stage 2: Structured Fact Extraction. Given the scene summary and video clip, this prompt extracts structured facts describing observable actions, entities,

and locations. The output follows a strict JSON schema in order to produce atomic, verifiable facts that can be independently checked. These structured facts form the basis of the narrative memory used by StoryTeller.

Instruction. You extract structured, evidence-grounded facts from a video clip for audio description. Return ONLY valid JSON exactly matching the schema. No markdown. No extra keys. No inference. Keep facts atomic (one action each) and limit to 6 to 10 items. Use neutral labels if identity is unclear. Fill the JSON schema below using only what is clearly visible or audible in the clip.

Schema:

```
{
  "scene_summary": "one sentence visual summary",
  "time_window": "start-end seconds or mm:ss-mm:ss",
  "facts": [
    {
      "subject": "who performed the action",
      "action": "verb phrase",
      "object": "what/who was acted upon; use empty string if
not needed",
      "location": "where in the scene",
      "time": "timestamp or short range",
      "evidence": "visual|audio|metadata",
      "confidence": "high|medium|low",
      "notes": "optional clarifications",
      "cluster_id": "if the action involves a TRACKLET_GUIDE entry,
copy its tracklet_id (e.g., trk_0003)",
      "character_id": "when 100% certain,
supply the canonical character_id from metadata"
    }
  ],
  "uncertain_observations": ["optional list describing ambiguities"]
}
```

TRACKLET_GUIDE (cluster IDs visible in this clip): {tracklet_hint}
 Spoiler-safe metadata snippets that may match this moment: {context_snippets}
 Metadata hints for this movie (use only if they match the video): {metadata_hint}

S10.3 Fact Verification Prompt

Stage 3: Fact Verification. To prevent hallucinated or weakly grounded observations from entering the memory, each extracted fact is passed through a verification step. The model acts as a strict fact checker and determines whether the proposed fact is clearly supported by the visual evidence in the clip. Only accepted facts are retained for downstream narration.

Instruction. You are a strict video-grounded fact checker. Accept only if the clip clearly supports the fact. If ambiguous, inferred, off-screen, or not shown: reject.
Prompt. You are verifying whether the FACT is truly visible or supported in this video clip. Use the video as the primary source. On the first line respond exactly with either ACCEPT or REJECT, then provide a one-sentence justification citing what you observed.
 FACT: {fact_text}

S10.4 Final Input Prompt

Stage 4: Final Narration Generation. The final prompt produces the audio description used for evaluation. It conditions on the verified fact set and optional narrative memory context from previous clips. The prompt instructs the model to generate a concise, visually grounded narration that reflects the scene while avoiding references to cameras or recording artifacts.

Instruction. You are an in-scene narrator describing events to blind or low-vision listeners. Deliver the narration as a storyteller within the world of the film. Never mention cameras, screens, or phrases such as “in this video,” “on screen,” “toward the camera,” or “looks at the camera.” If a draft would mention recording equipment, rewrite it so the line instead describes how characters relate to one another or to their environment.

Rely on grounded details from the FACT_BLOCK, keep the chronology clear, and favor concise active sentences. When a FACT_BLOCK entry includes a **character_id**, refer to that person by the provided first name only. Do not invent or guess names. Use the provided first name when the character is first mentioned; afterwards, pronouns such as he, she, or they may be used.

Highlight motion, intent, and key sounds. Omit filler and self-referential phrasing. Keep the final description under 20 words. Use MEMORY_CONTEXT only when it directly supports continuity without speculation.

Clip duration: {duration:.2f}s

Output format: Return JSON in the form {"summarised_AD": "..."}.

Examples. For a 0.8s clip: {"summarised_AD": "She looks at Riker."}

For a 1.4s clip: {"summarised_AD": "Paul looks at his wife lovingly."}

For a 2.6s clip: {"summarised_AD": "Returning to the room, Sarah peers into the darkness."}

For a 3.5s clip: {"summarised_AD": "Stephen looks at Sara as she walks to the door."}

S10.5 StoryAD-QA Question Generation Prompt

Track A (segment-only QA). The Track A prompt asks the VLM to generate a multiple-choice question about the main visual event within a single clip. For question generation only, the VLM also receives a short Wikipedia movie overview as background. The prompt explicitly restricts the model to information that is directly visible, prohibits the use of dialogue, external knowledge, or

speculative reasoning, and requires a short rationale grounded in visual evidence. If the clip does not contain a clear self-contained event, the model is instructed to output **SKIP**.

You are given:

- A video clip from a movie.
- A movie overview for background context only.

Overview: {movie_overview}

IMPORTANT: This clip may still be incomplete relative to the full movie. Do NOT assume anything outside what is visible in this clip. Do NOT use prior knowledge of the movie, characters, or story. Use only what is directly shown.

TASK: Your task is to generate ONE 5-option multiple-choice question about the main visual event or progression shown in this clip. If the clip does NOT contain a clear, self-contained action, reaction, reveal, or cause-effect moment that is fully shown, output:

SKIP

The question MUST reflect the main development of the clip, not a small detail from a brief moment. Focus on the most important or climactic visible event.

Do NOT generate a question if:

- The moment is routine or transitional
- The answer depends on events outside this clip
- It requires guessing thoughts, intentions, or emotions
- It relies on what happens before or after the clip

If uncertain, output:

SKIP

Requirements:

- 1–2 sentence question
- Answerable using only this clip
- No “Why” questions
- No references to before or after the clip
- No psychological interpretation
- Focus on clearly visible actions or outcomes

All incorrect options must be plausible.

Output format:

Question:

<question>

A.

B.

C.

D.

E.

Correct Answer: <letter>

Rationale:

<1-2 sentences describing only visible evidence>

Track B (context-conditioned QA). The Track B prompt introduces an explicit separation between a *context window* and a *target scene*. The context may introduce entities or situations, but the correct answer must be visually verifiable in the target scene. The prompt further discourages references to timestamps, dialogue, or outside knowledge to ensure questions remain grounded in observable visual information.

You are given a movie video clip and a brief movie overview for background only.
Overview: {movie_overview}

The first {context_len} seconds of the clip are CONTEXT.
The remaining part of the clip is the TARGET SCENE.

The context may introduce people, objects, or locations that help interpret the target scene.

Generate ONE multiple-choice question (5 options A-E) that requires using the context to understand what is happening in the target scene.

Key principle: The answer must be visible in the TARGET SCENE, but the CONTEXT should help identify or interpret the object, person, or situation referenced in the question.

Preferred patterns:

- An object or person appears in the context, and the question asks about its state, action, or location in the target scene.
- Something in the target scene is ambiguous unless the viewer remembers the context.
- The context introduces an item or situation whose consequence or change becomes visible in the target scene.

Rules:

- The question must be answerable using only what is visibly shown in the video.
- The correct answer must be visually verifiable in the TARGET SCENE (after the first {context_len} seconds).
- The context should help identify what the question refers to, but should not contain the answer itself.
- Do not rely on character names, dialogue, subtitles, or outside knowledge.
- Focus on observable actions, objects, interactions, or spatial relationships.
- Avoid trivial decorative details that do not affect the scene.
- Do not ask “Why” questions or infer emotions, intentions, or story meaning.
- Do not mention numeric timestamps, seconds, or “the first {context_len} seconds”.
- Do not refer to the video itself (no phrases like “in this clip” or “on screen”).

The question MUST reflect the main development of the clip, not a small detail from a brief moment. Focus on the most important or climactic visible event.

Do NOT generate a question if:

- The moment is routine or transitional
- The answer depends on events outside this clip
- It requires guessing thoughts, intentions, or emotions
- It relies on what happens before or after the clip

If there is no clear visual moment after the context that connects to something introduced earlier, output:

SKIP

Output format:

Question:

<question>

A. <option>

B. <option>

C. <option>

D. <option>

E. <option>

Correct Answer: <letter>

Rationale:

<1-2 sentences describing only the visible evidence in the target scene that confirms the answer>

S10.6 StoryAD-QA Question Answering Prompt

During evaluation, we provide the language model with only textual inputs: the question and the corresponding audio descriptions for the scene. The model is instructed to select an answer using only the information contained in the AD, without the use of external knowledge, prior familiarity with the movie, or assumptions about events not described in the AD text. In addition, the model is required to produce a short rationale citing explicit evidence from the AD, encouraging answers that are grounded in the provided narration.

To reduce positional bias, the answer options are randomly shuffled before being presented to the language model during evaluation.

You are given:

- Audio Descriptions (AD) for a movie scene.
- A multiple-choice question about that scene.

Question:
{question}

Audio Descriptions (AD):
{ad_text}

IMPORTANT: Use only information explicitly stated in the AD text. Do NOT use prior knowledge of the movie. Do NOT assume events outside the AD. Do NOT guess characters' thoughts or intentions unless the AD explicitly states them.

TASK: Select the best answer (A, B, C, D, or E) based only on the AD.

Output format:

Answer: <letter>

Rationale:

<1-2 sentences citing only explicit AD evidence>

References

1. Deng, J., Guo, J., Xue, N., Zafeiriou, S.: Arcface: Additive angular margin loss for deep face recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4690–4699 (2019)
2. Kala, D., Khandelwal, E., Tapaswi, M.: What you see is what you ask: Evaluating audio descriptions. In: Christodoulopoulos, C., Chakraborty, T., Rose, C., Peng, V. (eds.) Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 23496–23518. Association for Computational Linguistics, Suzhou, China (Nov 2025). <https://doi.org/10.18653/v1/2025.emnlp-main.1199>, <https://aclanthology.org/2025.emnlp-main.1199/>
3. Song, E., Chai, W., Wang, G., Zhang, Y., Zhou, H., Wu, F., Chi, H., Guo, X., Ye, T., Zhang, Y., Lu, Y., Hwang, J.N., Wang, G.: 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 (June 2024)
4. Xie, J., Han, T., Bain, M., Nagrani, A., Varol, G., Xie, W., Zisserman, A.: AutoAD-Zero: A training-free framework for zero-shot audio description. In: Proceedings of the Asian Conference on Computer Vision (ACCV). pp. 2265–2281 (2024)