POLITECNICO DI TORINO
Repository ISTITUZIONALE

Detecting and Mitigating Challenges in Zero-Shot Video Summarization with Video LLMs

Original
Detecting and Mitigating Challenges in Zero-Shot Video Summarization with Video LLMs / Cagliero, L., Vaiani, L., Pastor, E., Koudounas, A., Baralis, E., Mazzia, V., Pollastrini, S., Gueudre, T., Giollo, M., Amberti, D., Wu, Y.. - (2025), pp. 286-301. (63rd Annual Meeting of the Association for Computational Linguistics: ACL 2025 Vienna (AT) 27Jul - 1 Aug 2025).

Availability:
This version is available at: 11583/3002216 since: 2026-02-19T19:54:31Z

Publisher:
Association for Computational Linguistics (ACL)

Published
DOI:

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

18 November 2026

Detecting and Mitigating Challenges in Zero-Shot Video Summarization with Video LLMs

Luca Cagliero[1], **Lorenzo Vaiani**[1], **Eliana Pastor**[1], **Alkis Koudounas**[1], **Elena Baralis**[1], **Vittorio Mazzia**[2], **Sandro Pollastrini**[2], **Thomas Guedre**[2], **Manuel Giollo**[2], **Daniele Amberti**[2], **Yue Wu**[2]
[1]Politecnico di Torino, Italy [2]Amazon AGI, Italy
luca.cagliero@polito.it

Abstract

Video summarization aims to generate a condensed textual version of an original video. Summaries may consist of either plain text or a shortlist of salient events, possibly including temporal or spatial references. Video Large Language Models (VLLMs) exhibit impressive zero-shot capabilities in video analysis. However, their performance varies significantly according to the LLM prompt, the characteristics of the video, and the properties of the training data and LLM architecture.

In this work, we thoroughly evaluate the zero-shot summarization performance of four state-of-the-art open-source VLLMs specifically designed to address spatial and temporal reasoning. In light of the detected summarization issues, we propose different cost-effective mitigation strategies, based on Chain-of-Thought prompting, that involve the injection of knowledge extracted by external, lightweight models. To perform the VLLM evaluation, we design a new video summarization benchmark consisting of 100 videos with varying characteristics in terms of domain, duration, and spatio-temporal properties. Videos are manually annotated by three independent human experts with plain text, event-based, and spatio-temporal summaries.

The experimental evaluation shows that VLLMs significantly benefit from prompting a list of recognized actions, whereas injecting automatically recognized objects and scene changes respectively improve spatially contextualized and event-based summaries in specific cases.

1 Introduction

Large Language Models (LLMs) excel in a variety of Natural Language Generation tasks, among which question answering, text summarization, and text paraphrasing (Huang and Chang, 2023). However, when used in zero-shot learning LLMs may express unintended behaviors such as making up facts, generating biased or toxic text (Bender et al., 2021). For this reason, exploring zero-shot LLMs capabilities on benchmark datasets is fundamental to early detect model strengths and weaknesses, particularly when there is a lack of annotated data or computational resources for model fine-tuning.

Recent LLMs such as VideoChatGPT (Maaz et al., 2023) and VideoLLaVA (Lin et al., 2023) support video content as part of their input prompt. Thanks to their architecture and training data, they are potentially able to produce textual summaries of the video content under a zero-shot learning setting. In this work, we explore the use of Video Large Language Models (VLLMs) for zero-shot video summarization, detecting challenges and proposing ad hoc, cost-effective mitigation strategies.

According to the type of LLM prompt, summaries of arbitrary data might consist of indicative descriptions expressed in plain text (Syed et al., 2023) (e.g., using the prompt *Produce a free-text summary of the video*) or more informative content (El-Kassas et al., 2021) such as a list of key events (Nakshatri et al., 2023) (e.g., *Summarize the video by indicating a shortlist of salient events*), a timeline (La Quatra et al., 2021) (e.g., *Summarize the video by enumerating the relevant timestamps and their corresponding salient events*), a spatially contextualized summary (Cai and Hovy, 2011) (e.g., *Summarize the video by enumerating relevant locations and their corresponding salient events*), or a spatio-temporally contextualized summary (e.g., *Summarize the video by enumerating relevant pairs of location and timestamp and their corresponding salient events*).

Classical video summarization methods aim to identify the most relevant content while minimizing summary redundancy (Kansal et al., 2023). However, to effectively generate summaries of different types they also require complementary capabilities such as correctly identifying segments in long videos corresponding to salient events, properly

Findings of the Association for Computational Linguistics: ACL 2025, pages 286–301
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

handling temporal aspects, and accurately detecting spatial locations, event subjects and actions. Our purpose is to verify to what extent Video Large Language Models (VLLMs) show the following complementary abilities.

Handling temporal information. LLMs are known to have limitations in dealing with long-lasting events, capturing long-term temporal dependencies, and expressing temporal references without incurring in hallucination effects (Liu et al., 2024b). Recent works have started to explore VLLMs capabilities in video summarization, but mainly on short videos (20-60 seconds) (Liu et al., 2024a) and for plain text summarization only. This prompts the need to further investigate the unexplored aspects of various videos, summary types, and models.

Handling spatial information. Previous works have proposed to incorporate spatio-temporal knowledge in Transformer-based summarizers (Hsu et al., 2023) and also investigated the effectiveness of VLLMs in spatial reasoning (Fu et al., 2024). However, to the best of our knowledge, how to combine video summarization with spatial and temporal reasoning using LLMs is still an open research issue.

To address the issues mentioned above, this paper proposes the following contributions.

VLLM evaluation and issue detection. We thoroughly evaluate and compare the zero-shot summarization performance of four state-of-the-art open-source VLLMs that are specifically designed and trained to handle spatio-temporal information, i.e., VideoChatGPT (Maaz et al., 2023), VideoLLaMA2 (Cheng et al., 2024), VideoLLaVA (Lin et al., 2023), VTimeLLM (Huang et al., 2023)). We envisage appropriate methods to detect common model issues in the generation of summaries of different types. The results exhibit relevant flaws in the generated summaries, particularly while jointly tackling summarization with temporal and spatial reasoning.

New video summarization benchmark. To evaluate VLLM summarization performance, we present MIXTYVSUM (Mixed Type Video Summarization), a new benchmark consisting of 100 videos with varying domains, subjects, durations, and spatial and temporal features. Each video has been enriched with summary-related information with the help of three independent human annotators per video. Hence, unlike all prior benchmarks, it is suited to test the automatic generation of plain text, event-based, timeline, spatially and spatio-temporally contextualized summaries (more insights are given in Section 2). MIXTYVSUM is available in the project repository along with the code, LLM prompts, detected issues, and outputs.[1]

Mitigation strategies. Within the scope of zero-shot summarization, we envisage specific mitigation strategies aimed to partly overcome the summary issues detected at the previous step. To this end, we propose to adopt the following cost-effective external models, whose outputs in textual form are fed to the LLM via Chain-of-Thought prompting: (1) *Action recognition*. Motivated by the large of body of work devoted to action recognition (T and HR, 2024), our purpose is to improve the accuracy of salient event identification and description, which are key summarization steps. (2) *Object detection*. We leverage a list of detected objects to simplify both event description and spatial contextualization. (3) *Scene splitter*. We automatically detect scene changes in videos as they potentially represent key event boundaries. They are synergical to the temporal contextualization of the summary content. Action recognition has proved to be beneficial across almost all summary types and models, whereas scene detection and object recognition yield model- and type-specific improvements (e.g., object detection for spatially contextualized summaries on VideoChatGPT).

The remainder of the paper is organized as follows. Section 2 reviews the existing literature. Section 3 formally states the summarization tasks. Section 4 describes the MIXTYVSUM benchmark. Section 5 describes how we evaluate model performance, address the detection and characterization of summarization issues, and introduces the proposed mitigation strategies. Section 6 reports the main experimental results. Finally, Section 7 draws the conclusions and presents the future research directions.

2 Related works

Works on video summarization published until 2023 mainly rely on Transformers and Recurrent Networks. Conversely, in the last year, the attention of the research community has turned towards VLLMs. Hereafter, we will compare our work with

[1] https://github.com/VaianiLorenzo/video-summarization-with-LLMs

the state-of-the-art under the following dimensions: (1) Summary types, (2) Models, and (3) Temporal aspects.

Summary types Recent studies on summarization using VLLMs just consider plain text summaries (Liu et al., 2024a). To the best of our knowledge, the only attempt to generate timelines using LLMs has been made in (Sojitra et al., 2024) but source data are textual documents rather than videos. STVT (Hsu et al., 2023) embeds spatial information in summarization, but the proposed model is a Transformer instead of an LLM. Further variants of the summarization problem such as extracting key time frames (without a generative phase) (He et al., 2023; Zeng et al., 2025) or generating multimodal summaries (Zhu et al., 2018) are out of the scope of the present work and will be addressed in future work.

Models VLLMs encompass both proprietary and open-source models. OpenAI GPT-4V and GPT4o (OpenAI, 2023) and Google Gemini 1.5 Pro[2] are notable examples of proprietary models that are capable of processing videos along with textual prompts. Among the open-source VLLMs, VideoChatGPT (Maaz et al., 2023), VideoLlAMA2 (Cheng et al., 2024), VideoLLaVA (Lin et al., 2023), VTimeLLM (Huang et al., 2023) have been specifically designed and trained to reason about temporal and spatial aspects. For instance, the Spatial-Temporal Convolution connector of VideoLlAMA2 aims to effectively capture the intricate spatial and temporal dynamics of video data. VideoLLaVA adopts a joint training with videos and images to alleviate object hallucination, whereas VTimeLLM is specifically designed for fine-grained video moment understanding and reasoning with respect to delimit events' temporal boundaries. Our study is exclusively focused on open-source VLLMs. While several other open-source VLLMs have been proposed, we specifically select models that demonstrate state-of-the-art capabilities in temporal reasoning and action understanding. We exclude models that either lack crucial features for comprehensive video understanding (like temporal attention mechanisms or frame-to-frame correlation analysis) or show inferior performance in preliminary experiments.

Temporal aspects Whether VLLMs actually understand temporal aspects of videos is an open research question that is partly addressed by TempCompass (Liu et al., 2024b) and VideoMME (Fu et al., 2024). TempCompass analyzes five temporal aspects (i.e., Action, Speed, Direction, Attribute Change and Event Order) on various question answering and video captioning tasks. VideoMME, instead, evaluates question answering performance of both proprietary and open-source models on a large set of videos with varying duration, domains, and input modalities (i.e., audio, text, and visual). Neither TempCompass nor VideoMME address video summarization (see the task formulations in Section 3), which is, instead, the core of the present work.

3 Problem statement

We aim to evaluate the zero-shot capabilities of VLLMs (Zhang et al., 2024) in automatically generating video summaries of different types on a variety of videos with variable domains, subjects, and spatio-temporal video properties.

3.1 Summary types

Given a video V, we evaluate the following summaries:

- **Plain text Summary** (PS_V): a plain text providing indicative descriptions of the video (Syed et al., 2023). No constraints on the summary structure are given a priori.
- **Event-based Summary** (ES_V): a shortlist of salient events e_j, $j \in [1, N]$, happening in the video. For each event, the VLLM provides an informative textual description (El-Kassas et al., 2021), whereas an explicit spatio-temporal contextualization is not requested. Notice that the number N of salient events is a priori unknown.[3]
- **Timeline Summary** (TS_V): a sequence of video timestamps t_1, t_2, ..., t_Q at which salient events happen. For every timestamp, the VLLM shortlists the salient events occurred at t_i ($i \in [1, Q]$) as well as an informative textual description of each event. Notice that in general $Q \leq N$, because multiple salient events can start at the same time.

[2] https://deepmind.google/technologies/gemini/

[3] Event-based summarization differs from dense video captioning (Krishna et al., 2017) as the latter recognizes and describes each event separately rather than shortlisting them in a global video summary.

- **Spatially Contextualized Summary** (SCS_V): a sequence of spatial locations l_1, l_2, ..., l_R ($R \leq N$) where salient events happen in the video. For each location, the VLLM shortlists the salient events occurred in l_k ($k \in [1, R]$) as well as a textual description of each event.

- **Spatio-Temporally Contextualized Summary** ($STCS_V$): a sequence of location-timestamp pairs $\langle t_i, l_k \rangle$. For each pair the VLLM shortlists the corresponding salient events.

4 The Mixed Type Video Summarization benchmark

Publicly available datasets for video summarization often lack detailed contextual annotations, such as event descriptions enriched with temporal and spatial information about each observed event. Hence, they are not suited to evaluate summarization performance on summary targets other than plain text. To address this limitation, we shortlist a subset of 5 public datasets with variable characteristics in terms of video duration, source domain, subject, number, and types of salient events, i.e., ActivityNetCaptions (Heilbron et al., 2015), TVSum (Song et al., 2015), EpicKitchens (Damen et al., 2022), SumMe (Gygli et al., 2014), MSVD (Chen and Dolan, 2011). We randomly sample 20 videos per dataset and involve 3 PhD-level volunteers (European, between 25 and 40 years old) in the annotation with past experience on NLP-related tasks and crowdsourcing projects. They did not ask to be paid and approve content sharing as a mutual form of research teams' collaborations. The properties of the annotated video, reported in Table 1, show the diversity of the sampled videos, e.g., the duration varies from 7 to 970 seconds, the number of distinct events from 1 to 12, the number of distinct locations from 1 to 6. A more detailed dataset description is given in the Appendix.

We define a clear set of annotation guidelines, providing precise definitions for each required annotation and summary type. These guidelines are crucial for ensuring consistency and avoiding misinterpretation by human annotators. The annotation guidelines are available in the Appendix.

Given the subjectivity of the human annotations, we verify the level of agreement among annotators by comparing the number of detected events and the values of the annotated timestamps. The quantitative analysis confirm the consistency of the provided annotations. For instance, on TVSum the maximum percentage difference in the number of detected events is 11% (1 out of 9 events) whereas the maximum time gap is one second, which is negligible with respect to the average video duration (272s).

5 Model Evaluation, Issue Detection, and Mitigation Strategies

We run an extensive campaign of prompt tuning for all combinations of VLLMs and datasets. A selection of the VLLM prompts and the corresponding output summary is given in Section 6.4. A more extensive set of examples is reported in the Appendix due to the lack of space. We divide the prompts into two main categories: *template-enriched* (i.e., when the desired template is indicated in the prompt) and *without template* (i.e., otherwise).

Based on a preliminary exploration of the achieved results, we identify seven main categories of issues that severely impact the summary outputs. In the following, we first provide a definition of the key issues. Then, we outline the methodology for automatic issue detection and the experiments on issue characterization, with particular attention paid to time-related issues.

5.1 Issue definition

The key issues identified include:

- **Unavailable timestamp (TS-NA).** The timestamp information is not available.

- **Unavailable time range (TR-NA).** The summary does not include the expected time range information.

- **Hallucinated time content (HAL).** The summary contains temporal information, but this is hallucinated or invalid, e.g., 10:65 p.m.

- **Regular time division (REG-DIV).** Fabrication of events in an unnatural regular cadence, e.g., ' at minute 2... At minute 4...'

- **Time Fragmentation or repetition. (FRAG-REP).** Excessive repetition of information, especially in summaries meant to capture concise intervals, e.g., instead of 'from minute 10 to minute 15 eventX', we have ' minute 10 eventX, minute 11 eventX, ..'.

Original sample source	Videos and annotations				External knowledge		
	Avg video Duration (s)	Avg num. of events per video	Avg num. of locations per video	Avg num of tokens per summary	Avg num. of scenes	Avg num. of actions	Avg num. of objects
TVSum (2015)	272.25 ± 137.75	4.30 ± 2.93	4.10 ± 2.17	28.35 ± 11.96	26.5 ± 16.43	4.1 ± 2.84	77.05 ± 51.0
SumMe (2014)	131.6 ± 63.99	2.35 ± 1.39	0.90 ± 0.94	18.5 ± 6.67	7.95 ± 7.89	1.7 ± 1.52	30.9 ± 39.28
MSVD (2011)	10.1 ± 7.16	1.90 ± 1.76	1.30 ± 0.56	17.6 ± 5.96	1.4 ± 0.8	1.0 ± 0.0	40.65 ± 43.46
ActivityNet (2015)	121.3 ± 80.32	9.05 ± 3.02	1.70 ± 0.90	10.95 ± 4.01	8.6 ± 7.29	4.0 ± 2.92	135.15 ± 156.51
Epic-Kitchens (2022)	505.6 ± 465.87	7.60 ± 3.34	2.20 ± 0.98	40.35 ± 13.30	1.25 ± 1.09	11.1 ± 8.38	121.4 ± 156.16

Table 1: Statistics about videos, summaries, and injected knowledge.

- **Not compliant template (NOTCOMPL-TEMP).** The structure of the summary does not follow the requested template.

- **Focus on camera movements (UNREQ-CAMMOV).** The model generates summaries with a camera-like narrative, inaccurately reflecting the annotated events as though viewed by an observer rather than described objectively. The summary hence provides unnecessary mention of camera movements, e.g., 'the camera turns right...'.

5.2 Automatic detection of summary issues

We implement an NLP pipeline to automatically detect the above-mentioned issues. To detect temporal information, we use an established Named Entity Recognition libraries[4]. To identify fragmentation and repetition, we use a mix of regular expressions and LLM-based text annotation using LLaMA 3.1 70B (Touvron et al., 2023)[5].

Characterization of summaries' issues We characterize the issues detected from different summaries, models, and prompts, template-enriched and not. Overall, we evaluate 1,140 summaries consisting of completions of various prompt types. The outputs achieved by the prompts without templates averagely contain a number of issues that are one order of magnitude higher than those obtained by template-enriched ones. The gap is particularly evident for TS-NA and TR-NA because without templates VLLMs often omit the timeline and spatial annotations. When both spatial and temporal data are requested, VLLMs tend to privilege the spatial information while disregarding timeline information. Hereafter, we will mainly focus our analysis on template-enriched templates. A more extensive set of prompt-response pairs is reported in the Appendix.

Table 2 reports the statistics of observed per-type issues for each pair of dataset and model. Despite the videos having different durations, domains, and technical characteristics across the source datasets, the number of detected issues per dataset is pretty similar, confirming the pervasive nature of the reported issues. Full compliance with the expected templates is often missing (see issue NOTCOMPL-TEMP counts). Time content hallucination (HAL) turns out to be more severe with Video-LLaVa and VTimeLLM (despite the latter has a higher level of specialization on temporal reasoning) and is weakly influenced by the presence of templates. Regular time division (NOTCOMPL-TEMP) is pervasive on all datasets and models, whereas time fragmentation and repetition are less frequent on videos with limited duration. Video-LLaVA2 turns out to be the most effective in avoiding time fragmentation thanks to its robust model pretraining.

5.3 Issue mitigation

We envisage the use of in-context learning with Chain-of-Thought (CoT) prompting to mitigate the issue detected in the VLLM benchmarking phase. We leverage lightweight, external models to extract video information relevant to accurately shortlist salient events and define time ranges and spatial locations. More specifically, we employ:

- *Scene splitter*[6], an automatic scene detector that detects scene changes and reports a list of scenes with the corresponding starting and ending times. Our idea is that scene changes could be helpful to define the occurrences of key events and their timing. To avoid introducing bias, we ignore too fast scene changes (i.e., distant less than Video Large Language Modelses).

- *Action Recognizer* (Contributors, 2020), an open-source toolbox for video understanding

[4] https://spacy.io/api/entityrecognizer

[5] We use the cloud service provided at https://www.together.ai/

[6] https://github.com/Breakthrough/PySceneDetect

Dataset/Model	TS-NA	TR-NA	HAL	REG-DIV	FRAG-REP	NOTCOMPL-TEMP	UNREQ-CAMMOV
Video-ChatGPT	**273**	**258**	**0**	**127**	**83**	**281**	**9**
ActivityNet	30	35	0	10	2	56	1
Epic-Kitchens	25	29	0	15	4	52	2
MSVD	32	37	0	8	2	58	0
SumMe	28	34	0	12	3	57	3
TVSum	29	38	0	11	4	58	3
Video-LLaMA2	**65**	**101**	**0**	**135**	**4**	**214**	**5**
ActivityNet	11	22	0	29	0	44	0
Epic-Kitchens	5	10	0	35	1	35	2
MSVD	22	29	0	18	0	52	1
SumMe	14	21	0	26	0	42	1
TVSum	13	19	0	27	3	41	1
Video-LLaVA	**100**	**104**	**68**	**100**	**41**	**204**	**1**
ActivityNet	21	21	16	19	7	41	0
Epic-Kitchens	18	20	13	22	2	40	0
MSVD	20	21	15	20	8	41	1
SumMe	20	21	9	20	13	41	0
TVSum	21	21	15	19	11	41	0
VTimeLLM	**108**	**108**	**63**	**92**	**32**	**208**	**21**
ActivityNet	22	22	14	18	8	42	3
Epic-Kitchens	20	20	16	20	2	40	2
MSVD	23	23	12	17	8	43	4
SumMe	22	22	7	18	6	42	8
TVSum	21	21	14	19	8	41	4
Total	**417**	**486**	**131**	**383**	**92**	**907**	**36**

Table 2: Counts of the number of issues per dataset and model with template-enriched prompts only.

which provides a textual description of the observed actions. The key idea is to leverage actions to compose event descriptions.

- *Object Recognizer*, a real-time video processing tool (namely, Yolo v2 (Redmon and Farhadi, 2016)) that detects objects in videos. The idea behind it is to use the detected objects to characterize events or spatial locations better. Objects are ranked by decreasing time of appearance in the video, and the object list is early pruned to avoid recommending the less relevant items.

We post-process the external models' outputs to produce knowledge-enriched prompts A set of representative prompts is given in the Appendix.

6 Experiments

We run the experiments on a machine equipped with Intel® Core™ i9-10980XE CPU, 1 × NVIDIA® RTX A6000 48GB GPU, 128 GB of RAM running Ubuntu 22.04 LTS.

In this section, we first define the performance metrics (see Section 6.1). Then, we compare zero-shot model performance with and without mitigation (see Section 6.2). Next, we deepen our analysis of separate summary types (see Section 6.3) and provide qualitative examples (see Section 6.4).

6.1 Metrics

Given a video V, we evaluate its LLM-generated summaries against the corresponding ground truth. For each plain text summary PS_V we quantify its similarity with the ground truth PS_V^{gt} using three established methods: (1) *Syntactic similarity* computed by using the Rouge toolkit (Lin, 2004), i.e., the R1/R2/RL Precision/Recall/F1-Score values, which quantify the text overlap in terms of unigrams, bigrams, and longest matching subsequence; (2) *Semantic similarity* using BERTScore F1-Score (Zhang et al., 2020), which measures the similarity in the embedding space (Reimers et al., 2019); (3) *LLM-as-a-judge*, where we inquiry a robust LLM (i.e., LLaMA 3.1 70B (Touvron et al., 2023)) with the summaries produced by different methods and ask him to declare the winner.

To evaluate event-based summaries, we first retrieve the ground truth events that appear in the summary and then count the Precision/Recall/F1-Score values of the output event list against the ground truth. To address event retrieval, we compare the text of the summary with the description of each ground truth event using both the BERTScore F1-Score and the *LLM-as-a-judge* approach (*Given the event-based summary ES_V and the textual description of an event e, the LLM indicates whether the event appears in the summary or not*).

While the accuracy of spatial information is verified using the classical Precision score, we evaluate timeline generation by also accounting for the *latency* (Dhingra et al., 2022), which is denoted by the time gap (in seconds) between predicted and expected event timestamps.

6.2 LLM-as-a-judge

For each summary type, model, and video we employ a LLM-as-a-judge approach to compare the summary produced with the best prompt without CoT with the summaries generated using different mitigation strategies. Table 3 reports for each strategy the number of video samples for which each strategy has been judged the best separately for every combination of summary type and VLLM. When the judge declares no winner, the samples are labeled as *Ties*. The mitigation strategy based on action recognition exhibits the best performance for most combinations of model and type, except for the mix of spatial and temporal contexts, where the TR-NA and NOTCOMPL-TEMP issues remain unsolved. The benefits of scene split and object detection are more limited and do not emerge from this experiment.

6.3 Additional results' insights

Event-based summaries VTimeLLM, VideoLLaMA2 and VideoLLaVa exhibit similar F1 score variations on different datasets and settings (between 0.47 and 0.7, as shown in Figure 1). VideoChatGPT is less robust to long videos (e.g., TVSum samples), but the injection of actions and scenes compensates the performance gap (see Figures 2a and 2b). MSVD is instead too short to take advantage of actions or scene detection.

Spatially contextualized summaries Object detection yields F1 score improvements in spatial contextualization (see Figures 2c), even though the event descriptions are, in general, less accurate (see Table 3).

Timeline summaries Temporal information is often absent from timelines and spatio-temporally contextualized summaries (from 15% to 40% of the cases) or largely imprecise. Neither scene splitting nor object recognition are beneficial. Conversely, injecting the recognized actions yields improvements for all models. Specifically, (1) The percentage of summaries with missing temporal information drops to 10% or even less in most cases; (2) the latency (see Section 6.1) become acceptable (between 8% and 11% of the video duration) on ActivityNet, TVSum, and SumMe video samples, is irrelevant on MSVD samples due to the very limited video durations, whereas remains challenging on EpicKitchen, probably due to the high specificity of the domain which would require a higher model specialization.

Plain text summary Table 4 compares the VLLMs in terms of various metrics used for plain-text summary evaluation (see Section 6.1). Video-LLaVA achieves the highest recall on the syntactic measures, indicating a superior ability to include salient content in the summary. Oppositely, Video-LLaMA2 is, on average, superior in terms of ROUGE precision, suggesting a lower redundancy of the generated output. In terms of semantic similarity, they achieve comparable performance. The complete results set is given in the Appendix.

6.4 Qualitative example

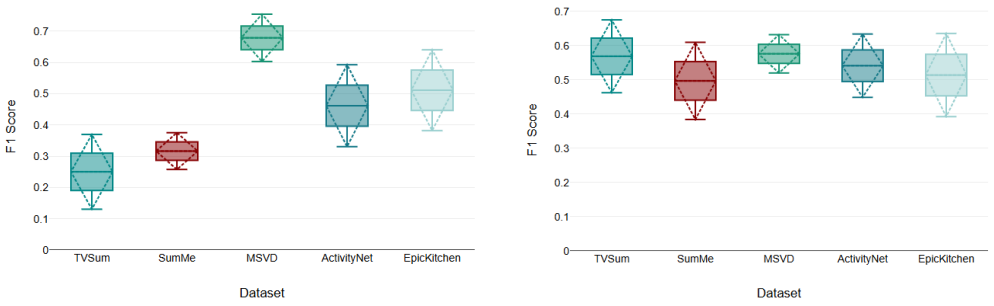
Here we comment on the timeline summary generated from the annotated TVSum video named *37rz-WOQsNIw.mp4*. Additional examples can be found in the official repository. The VideoLLaVA timeline lacks temporal information even while using template-enriched prompts: *The video starts with a person holding a piece of food in their hand, which appears to be a type of sandwich or wrap. The person then offers the food to a group of pigeons gathered on the ground....* By injecting knowledge from the action recognizer, the timeline gets correct: *<24-26>: making a sandwich <53-1:07>: feeding birds <1:26-1:47>: feeding birds <1:48-1:51>: walking the dog <2:01-2:03>: making tea <2:10-2:22>: feeding birds*. Conversely, the scene splitter detects 12 scenes, but half of the scene changes turn out to be redundant, e.g., *...In the second scene, the person is feeding the pigeons with the food. In the third scene, the person is holding a piece of food and looking at it....*

7 Conclusions and future work

The paper explored the zero-shot summarization performance of four open-source VLLMs and presented a benchmark datasets suited to evaluate five different summary types. The models' outputs show relevant issues, particularly regarding the adherence to summary templates, the temporal reasoning, and the mix of spatial and temporal dimensions. To tackle these issues, the paper proposed and assessed three CoT-based strategies. The experiments show that injecting knowledge from cost-effective, external models could compensate the inherent limitations of VLLMs used in a zero-shot setting. The proposed approach is suited to scenarios in which LLM fine-tuning is unfeasible due to the lack of training data or computational

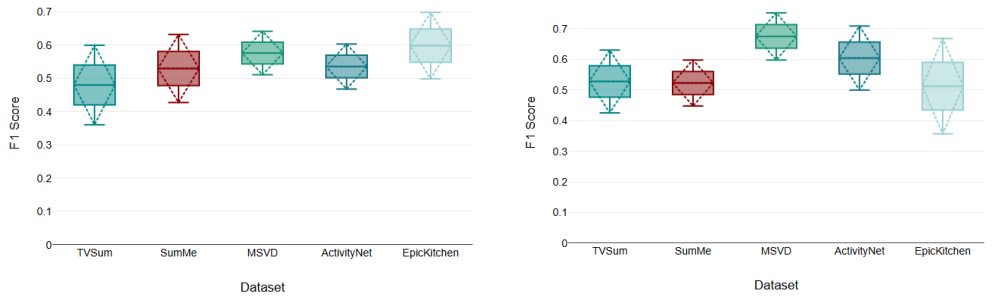
Summary Type	VLLM	Ties	No mitigation	+Scenes	+Objects	+Actions
Plain text	VideoChatGPT	14	17	6	14	**49**
	VideoLLaVA	**28**	18	11	15	**28**
	VideoLLaMA2	18	24	12	11	**35**
	VTimeLLM	16	**39**	7	10	28
	Total	76	98	36	50	**140**
Event-based	VideoChatGPT	24	9	5	4	**58**
	VideoLLaVA	25	13	13	17	**32**
	VideoLLaMA2	31	10	10	12	**37**
	VTimeLLM	25	25	12	5	**33**
	Total	105	57	40	38	**160**
Spatially contextualized	VideoChatGPT	20	13	7	11	**49**
	VideoLLaVA	**27**	14	12	20	**27**
	VideoLLaMA2	23	19	10	14	**34**
	VTimeLLM	10	29	17	14	**30**
	Total	80	75	46	59	**140**
Timeline	VideoChatGPT	1	37	3	5	**54**
	VideoLLaVA	2	25	18	19	**36**
	VideoLLaMA2	0	29	23	7	**41**
	VTimeLLM	2	23	27	3	**45**
	Total	5	114	71	34	**176**
spatio-temporal	VideoChatGPT	0	**48**	4	1	47
	VideoLLaVA	3	**42**	12	11	32
	VideoLLaMA2	0	**37**	27	7	29
	VTimeLLM	0	**47**	18	4	31
	Total	3	**174**	61	23	139

Table 3: LLM-as-a-Judge evaluation with LLaMA70B as an expert. For each summary type and model, and video the strategy declared the winner is written in boldface.



(a) VideoChatGPT

(b) VideoLLaMA2



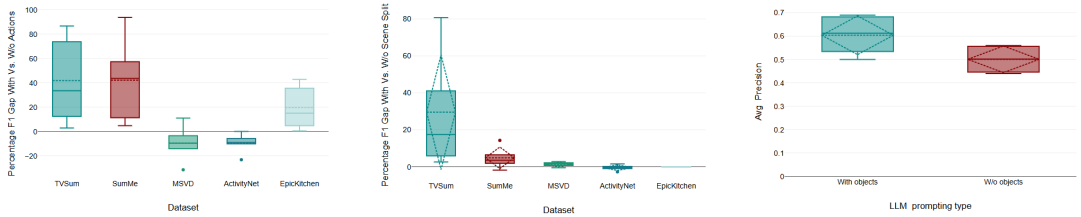
(c) VideoLLaVa

(d) VTimeLLM

Figure 1: Event-based summaries (without mitigation). Average F1-Score achieved by different VLLMs.

Model	Rouge-1 Precision	Rouge-1 Recall	Rouge-1 F1	Rouge-2 Precision	Rouge-2 Recall	Rouge-2 F1	Rouge-L Precision	Rouge-L Recall	Rouge-L F1	BERTScore F1
Video-ChatGPT	0/0/5	0/0/5	0/0/5	0/0/5	0/0/5	0/0/5	0/0/5	0/0/5	0/0/5	0/0/5
Video-LLaVA	0/0/5	**5/0/0**	**4/0/1**	1/0/4	**4/1/0**	**4/0/1**	2/0/3	**5/0/0**	**4/0/1**	**2/0/3**
Video-LLaMA2	**5/0/0**	0/0/5	0/0/5	**3/0/2**	0/0/5	0/0/5	**3/0/2**	0/0/5	0/0/5	1/0/4
VTimeLLM	0/0/5	0/0/5	1/0/4	1/0/4	0/1/4	1/0/4	0/0/5	0/0/5	1/0/4	**2/0/3**

Table 4: Number of Wins/Ties/Losses for each pair of VLLM and metric (out of 5 different data sources).



(a) Event-based summary. VideoChatGPT with actions

(b) Event-based summary. VideoChatGPT with scenes

(c) Spatially contextual summary. With vs. w/o object recognition

Figure 2: Percentage F1-Score improvements achieved by different mitigation strategies.

resources. Beyond extending the analysis to proprietary models, we will plan to enrich video content with audio and speech data and to study the extractive summarization and co-summarization tasks.

Limitations

Models We limit the scope of our analysis to relatively small opensource models, i.e., 7 Billion VLLMs. The use of larger models, proprietary or not, could mitigate part of the detected issues. The proposed mitigation strategies can be helpful to bridge the technology gap with larger models when the computational resources are limited.

Sample size Given the significant human effort required to annotate videos with accurate summary-related information, the benchmark size cannot be further extended. This prevents us from addressing LLM fine-tuning, which is out of the scope of the present work. Our aim is, instead, to explore LLM zero-shot capabilities in handling different summaries. Therefore, we maximize the richness of summary annotation and the diversity of the video samples across multiple domains, subjects, video durations, and number and type of events and locations.

Data modalities We currently analyze the video content while disregarding acoustic and speech features. Applying speech to text or adopting Audio Visual LLMs are possible extensions which could simplify the identification and characterization of complex events.

Summary types We consider abstractive summarization tasks as they best fit Language Models' goal. They are known to suffer from the excessive video source length. To alleviate this issue, extractive summarization techniques can be applied prior to the abstractive step. In our research, we adopt, instead, a scene splitter to avoid dealing with excessively long video tracks.

Fine-tuning for specific domains and video types We annotate and analyze a collection of videos retrieved from mixed sources. To adapt LLM responses to the specific domain and summary type, LLM fine-tuning would significantly help to boost summarization performance. In the presence work, we exclusively analyze zero-shot LLM capabilities and try to mitigate summary issue using CoT prompting.

Ethics Statement

The paper adheres to the ACL Ethics Policy. The annotated videos were recorded and publicly released by third parties. Human annotations do not contain offensive, harmful, or non-inclusive expressions. Since our approach relies on pretrained VLLMs and ad hoc external models for knowledge injection, we cannot exclude the presence of bias or hallucination in the model outputs. However, the key contributions of the paper (detection and mitigation of VLLMs challenges) go in the direction of improving the awareness of VLLM limitations and preventing such negative effects when using them in a zero-shot learning setting.

Acknowledgements

This study was also partially carried out within the FAIR (Future Artificial Intelligence Research) and received funding from Next-GenerationEU (Italian PNRR – M4 C2, Invest 1.3 – D.D. 1555.11-10-2022, PE00000013). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

References

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *FAccT '21: 2021 ACM Conference on Fairness, Accountability, and Transparency, Virtual Event / Toronto, Canada, March 3-10, 2021*, pages 610–623. ACM.

Congxing Cai and Eduard Hovy. 2011. Summarizing textual information about locations. In *Proceedings of the 2nd International Conference on Computing for Geospatial Research & Applications*, COM.Geo '11, New York, NY, USA. Association for Computing Machinery.

David L. Chen and William B. Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, page 190–200, USA. Association for Computational Linguistics.

Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, and Lidong Bing. 2024. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms.

MMAction2 Contributors. 2020. Openmmlab's next generation video understanding toolbox and benchmark. `https://github.com/open-mmlab/mmaction2`.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Jian Ma, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, and Michael Wray. 2022. Rescaling egocentric vision: Collection, pipeline and challenges for epic-kitchens-100. *International Journal of Computer Vision (IJCV)*, 130:33–55.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. 2022. Time-aware language models as temporal knowledge bases. *Trans. Assoc. Comput. Linguistics*, 10:257–273.

Wafaa S. El-Kassas, Cherif R. Salama, Ahmed A. Rafea, and Hoda K. Mohamed. 2021. Automatic text summarization: A comprehensive survey. *Expert Systems with Applications*, 165:113679.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Rongrong Ji, and Xing Sun. 2024. Video-mme: The first-ever comprehensive evaluation benchmark of multi-modal llms in video analysis.

Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. 2014. Creating summaries from user videos. In *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VII*, volume 8695 of *Lecture Notes in Computer Science*, pages 505–520. Springer.

Bo He, Jun Wang, Jielin Qiu, Trung Bui, Abhinav Shrivastava, and Zhaowen Wang. 2023. Align and attend: Multimodal summarization with dual contrastive losses. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14867–14878.

Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. 2015. Activitynet: A large-scale video benchmark for human activity understanding. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 961–970.

Tzu-Chun Hsu, Yi-Sheng Liao, and Chun-Rong Huang. 2023. Video summarization with spatiotemporal vision transformer. *IEEE Transactions on Image Processing*, 32:3013–3026.

Bin Huang, Xin Wang, Hong Chen, Zihan Song, and Wenwu Zhu. 2023. Vtimellm: Empower llm to grasp video moments.

Jie Huang and Kevin Chen-Chuan Chang. 2023. Towards reasoning in large language models: A survey. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 1049–1065. Association for Computational Linguistics.

Kajal Kansal, Nikita Kansal, Sreevaatsav Bavana, Bodla Krishna Vamshi, and Nidhi Goyal. 2023. A systematic study on video summarization: Approaches, challenges, and future directions. In *Proceedings of the 2nd Workshop on User-Centric Narrative Summarization of Long Videos*, NarSUM '23, page 65–73, New York, NY, USA. Association for Computing Machinery.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. 2017. Dense-captioning events in videos. In *International Conference on Computer Vision (ICCV)*.

Moreno La Quatra, Luca Cagliero, Elena Baralis, Alberto Messina, and Maurizio Montagnuolo. 2021. Summarize dates first: A paradigm shift in timeline summarization. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 418–427, New York, NY, USA. Association for Computing Machinery.

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

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Tingkai Liu, Yunzhe Tao, Haogeng Liu, Qihang Fang, Ding Zhou, Huaibo Huang, Ran He, and Hongxia Yang. 2024a. DeVAn: Dense video annotation for video-language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14305–14321, Bangkok, Thailand. Association for Computational Linguistics.

Yuanxin Liu, Shicheng Li, Yi Liu, Yuxiang Wang, Shuhuai Ren, Lei Li, Sishuo Chen, Xu Sun, and Lu Hou. 2024b. TempCompass: Do video LLMs really understand videos? In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 8731–8772, Bangkok, Thailand. Association for Computational Linguistics.

Muhammad Maaz, Hanoona Rasheed, Salman Khan, and Fahad Shahbaz Khan. 2023. Video-chatgpt: Towards detailed video understanding via large vision and language models. *arXiv:2306.05424*.

Nishanth Nakshatri, Siyi Liu, Sihao Chen, Dan Roth, Dan Goldwasser, and Daniel Hopkins. 2023. Using LLM for improving key event discovery: Temporal-guided news stream clustering with event summaries. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4162–4173, Singapore. Association for Computational Linguistics.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Joseph Redmon and Ali Farhadi. 2016. Yolo9000: Better, faster, stronger. *arXiv preprint arXiv:1612.08242*.

Nils Reimers, Iryna Gurevych, Nils Reimers, Iryna Gurevych, Nandan Thakur, Nils Reimers, Johannes Daxenberger, Iryna Gurevych, Nils Reimers, Iryna Gurevych, et al. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Daivik Sojitra, Raghav Jain, Sriparna Saha, Adam Jatowt, and Manish Gupta. 2024. Timeline summarization in the era of llms. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 2657–2661, New York, NY, USA. Association for Computing Machinery.

Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. 2015. Tvsum: Summarizing web videos using titles. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, pages 5179–5187. IEEE Computer Society.

Shahbaz Syed, Dominik Schwabe, Khalid Al-Khatib, and Martin Potthast. 2023. Indicative summarization of long discussions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2752–2788, Singapore. Association for Computational Linguistics.

Gayathri T and Mamatha HR. 2024. How to improve video analytics with action recognition: A survey. *ACM Comput. Surv.*, 57(1).

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Xiangyu Zeng, Kunchang Li, Chenting Wang, Xinhao Li, Tianxiang Jiang, Ziang Yan, Songze Li, Yansong Shi, Zhengrong Yue, Yi Wang, Yali Wang, Yu Qiao, and Limin Wang. 2025. Timesuite: Improving mllms for long video understanding via grounded tuning.

Haopeng Zhang, Philip S. Yu, and Jiawei Zhang. 2024. A systematic survey of text summarization: From statistical methods to large language models.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Junnan Zhu, Haoran Li, Tianshang Liu, Yu Zhou, Jiajun Zhang, and Chengqing Zong. 2018. MSMO: Multimodal summarization with multimodal output. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4154–4164, Brussels, Belgium. Association for Computational Linguistics.

A Appendix

Content outline

- Description of the data sources used a starting point for the MIXTYVSUM annotation (see Section A.1).
- Guidelines for the video annotators (see Section A.2).
- Additional statistics about the detected summarization issues (see Section A.3).
- Representative examples for LLM prompts, outputs, and injected knowledge (see Section A.4).
- Additional results for plain text summarization (see Section A.5).

A.1 Description of the initial datasets

We build our MIXTYVSUM benchmark on the following benchmark datasets tailored to video summarization:

- ActivityNetCaptions (Heilbron et al., 2015)
- TVSum (Song et al., 2015)
- EpicKitchens (Damen et al., 2022)
- SumMe (Gygli et al., 2014)
- MSVD (Chen and Dolan, 2011)

In the following, we outline the identified dataset, specifying the available annotations and missing ones, highlighting the need for collecting annotations to benchmark summaries with respect to the summarization objective functions.

ActivityNetCaptions It consists of videos lasting from 1 to 5 minutes. For every salient event happening in the video, it reports the corresponding starting and ending times (intervals are potentially overlapped with other events) and textual description (*The old man is playing the piano*). Only the annotation of individual events is present, and there is no annotation that summarizes the video in its entirety.

TVSum It consists of videos lasting from 1 to 10 minutes. Each video frame is annotated with a relevance score (1:*Irrelevant*, 5:*Very relevant*). No textual annotation is present, neither global nor at the event level. Timestamps are not provided but can be derived from important frame-level annotations.

EpicKitchen It consists of ego-centric videos lasting from 2 to 60 minutes, showing actions performed in a kitchen. For every salient action happening in the video, it reports the corresponding starting and ending times and textual description consisting of action and entity (*Open-door*, *Take-plate*). Only the annotation of individual actions is present, and there is no annotation that summarizes the video in its entirety.

SumMe It consists of 25 short videos lasting from less than one to slightly more than 5 minutes. Each video is segmented into consecutive frames and annotated with ground truth relevance scores at the frame level. No textual annotation is present, neither global nor at the event level. Timestamps are not provided but can be derived from important frame-level annotations.

MSVD It contains 1970 short videos, each approximately one minute long. Each video is annotated with a set of equivalent descriptions, each provided by a different annotator. Each description mainly focuses on the subject and the action without particular spatial contextualization. No timestamps are provided. The annotations are generic and do not focus on subjects, actions, or places. No timestamp or time reference is provided in the annotations.

A.2 Annotation guidelines and annotator details

Annotation guidelines We provide annotators with a clear set of annotation guidelines to ensure consistency and reliability. We designed these guidelines to standardize the interpretation of the tasks, reducing subjectivity and ambiguity in the annotation process.

In the guidelines, we first overviewed the task and clarified the main objectives. Then, we provided a definition of the three key components annotators were required to produce for each video: plain summary, salient events, timestamps and locations.

The *plain text summary* should offer a concise and factual overview of the video. We instructed annotators to watch the entire video and identify its main subject or theme, key participants or entities, and general structure, avoiding personal opinions or interpretations. We advised the annotators to limit the summaries to two or three sentences and to avoid personal opinions and interpretations.

Salient events represent significant occurrences within the video that are essential to understanding its context. Annotators had to identify and describe these events based on their relevance, distinctiveness, and impact, ensuring that each event was clearly distinguishable from others. We instructed annotators to document each event in chronological order, with each description focusing on key actions, interactions, or transitions in the video. We encouraged users to annotate 1 to 10 events per video, depending on its content and length, with flexibility for additional events when needed. Annotators had then to annotate each salient event with specific *temporal and spatial information*. Annotators specified the start and end of the event using the format [HH:MM:SS]. Locations were described in terms of physical settings (e.g., "conference room," "center of the soccer field") or specific areas within the video frame.

The guidelines also included a FAQ section, addressing common questions and clarifying procedures based on an internal trial annotation phase. These answers provided further information, answering potential doubts in the process in a more direct and informal manner. We also provided examples of annotations to help annotators familiarize themselves with the requirements and the tasks. The full guidelines provided to annotators are available in our repository.

Annotators We involved three annotators per dataset to address the variability and subjectivity inherent in annotation. We recruited annotators from students and collaborators at our institutions, all with backgrounds in Computer Science and Data Science. Demographically, 25% identified as female, while 75% identified as male; most annotators (75%) were aged 26–29, with 12.5% aged ≤ 25 and 12.5% aged ≥ 31.

A.3 Additional statistics on summarization issues

Tables 5-9 provide more detailed statistics on frequencies of the identified summarization issues for both template-enriched prompts only and for all prompts. The enforcement of templates significantly reduces the impact of the major summarization issues. However, a significant number of outputs still lack compliance with the requested template. Moreover, temporal information is often missing or imprecise, calling for ad hoc mitigation strategies (see Section 5.3).

A.4 LLM prompts and injected knowledge

Tables 10 and 11 respectively report a set of representative examples of prompts and outputs, and injected knowledge. The strategies denoted by *Scene*, *Action*, or *Object* correspond to different mitigation strategies based on CoT prompting.

A.5 Additional results for plain text summarization

Tables 12-16 report the Rouge and BERTScore results achieved by the tested VLLMs on the MIX-TYVSUM video samples. The per-dataset results confirm the highest recall of Video-LlaVA and the highest precision of Video-LlAMA2.

Dataset	TS-NA	TR-NA	HAL	REG-DIV	FRAG-REP	NOTCOMPL-TEMP	UNREQ-CAMMOV
Video-ChatGPT	144	173	0	56	15	281	9
Video-LLaMA2	65	101	0	135	4	214	5
Video-LLaVA	100	104	68	100	41	204	1
VTimeLLM	108	108	63	92	32	208	21
Total	**417**	**486**	**131**	**383**	**92**	**907**	**36**

Table 5: Counts of the number of issues detected separately for each VLLM. Template-enriched prompts.

Dataset	TS-NA	TR-NA	HAL	REG-DIV	FRAG-REP	NOTCOMPL-TEMP	UNREQ-CAMMOV
Video-ChatGPT	273	258	0	127	83	281	27
Video-LLaMA2	236	189	1	164	8	214	36
Video-LLaVA	200	204	76	200	47	204	6
VTimeLLM	308	208	63	92	41	208	60
Total	**1017**	**859**	**140**	**583**	**179**	**907**	**129**

Table 6: Counts of the number of issues detected separately for each VLLM. Generic prompts.

Dataset/Model	TS-NA	TR-NA	HAL	REG-DIV	FRAG-REP	NOTCOMPL-TEMP	UNREQ-CAMMOV
Video-ChatGPT	**273**	**258**	**0**	**127**	**83**	**281**	**9**
ActivityNet	60	53	0	20	12	56	1
Epic-Kitchens	48	47	0	32	19	52	2
MSVD	59	54	0	21	11	58	0
SumMe	**54**	**49**	**0**	**26**	**15**	**57**	**3**
TVSum	**52**	**55**	**0**	**28**	**26**	**58**	**3**
Video-LLaMA2	**236**	**189**	**1**	**164**	**8**	**214**	**5**
ActivityNet	41	39	0	39	2	44	0
Epic-Kitchens	39	26	1	41	1	35	2
MSVD	60	48	0	20	1	52	1
SumMe	51	40	0	29	0	42	1
TVSum	45	36	0	35	4	41	1
Video-LLaVA	**200**	**204**	**76**	**200**	**47**	**204**	**1**
ActivityNet	41	41	21	39	8	41	0
Epic-Kitchens	38	40	14	42	2	40	0
MSVD	40	41	15	40	8	41	1
SumMe	40	41	9	40	15	41	0
TVSum	41	41	17	39	14	41	0
VTimeLLM	**308**	**208**	**63**	**92**	**41**	**208**	**21**
ActivityNet	62	42	14	18	9	42	3
Epic-Kitchens	60	40	16	20	2	40	2
MSVD	63	43	12	17	12	43	4
SumMe	62	42	7	18	7	42	8
TVSum	61	41	14	19	11	41	4
Total	**1017**	**859**	**140**	**583**	**179**	**907**	**36**

Table 7: Counts of the number of issues detected separately for each data source and VLLM. Prompts without template.

Dataset	TS-NA	TR-NA	HAL	REG-DIV	FRAG-REP	NOTCOMPL-TEMP	UNREQ-CAMMOV
ActivityNet	84	100	30	76	17	183	4
Epic-Kitchens	68	79	29	92	9	167	6
MSVD	97	110	27	63	18	194	6
SumMe	84	98	16	76	22	182	12
TVSum	84	99	29	76	26	181	8
Total	**417**	**486**	**131**	**383**	**92**	**907**	**36**

Table 8: Counts of the number of issues detected separately for each data source. Template-enriched prompts.

Dataset	TS-NA	TR-NA	HAL	REG-DIV	FRAG-REP	NOTCOMPL-TEMP	UNREQ-CAMMOV
ActivityNet	204	175	35	116	31	183	18
Epic-Kitchens	185	153	31	135	24	167	16
MSVD	222	186	27	98	32	194	25
SumMe	207	172	16	0	37	182	41
TVSum	199	173	31	121	55	181	29
Total	**1017**	**859**	**140**	**583**	**179**	**907**	**129**

Table 9: Counts of the number of issues detected separately for each data source. Generic prompts.

Summary type	Prompt	Output
Plain-text	Summarize the video.	The video shows a man using a tire to demonstrate how to remove rust from a metal object. He applies a substance to the tire and sprays it onto the metal object, then uses a brush to scrub the rust off.
Event-based	Enumerate the salient events or actions occurring in the video.	The video shows a man using a tire to clean a mower. He sprays the tire with a cleaning solution, wipes it down with a cloth, and then uses a brush to scrub the tire. He then rinses the tire with water and dries it with a cloth.
Spatially context	Provide a description of the video grouping together the events occurring in the place/location.	The video shows a man in a garage, who is seen cleaning a tire with a brush and a spray bottle. He then proceeds to clean the inside of the tire with the same brush and spray bottle. The man is seen standing in front of a fence and a garage door, and he is also seen standing in front of a wall with a tire on it.
Timeline	Provide the timeline of salient events or actions occurring in the video. Use the format: <start-timestamp>-<end-timestamp>:<event/action>	00:00-00:10: A man is seen holding a tire. 00:10-00:20: The man is seen holding a bottle of spray. 00:20-00:30: The man sprays the tire with the bottle. 00:30-00:40: The man is seen holding a tool. 00:40-00:50: The man sprays the tire again. 00:50-01:00: The man is seen holding a tire.
Spatio-temporal	Provide a summary highlighting the main spatio-temporal contexts in the video. For every event that occurs in the video, provide details about the location, start time, end time, and the event/action. Use the format: <place/location>-<start-timestamp>-<end-timestamp>: <event/action>	The video takes place in a garage and a yard. The man is seen working on a tire in the garage, and later in the yard, he is seen working on a lawn mower. The video starts at 00:00:00 and ends at 00:20:00.

Table 10: Examples of prompts without mitigation and corresponding outputs. VLLM: Video-LLaVA. Dataset: ActivityNet. Video: AwmHb44_ouw.mp4

Mitigation	Prompt	Output
Scene	Summarize the video. The video consists of 22 scenes total. The scene 1 starts at 0 second(s) and ends at 2 second(s). ... The scene 10 starts at 2 minute(s) and 0 second(s) and ends at 2 minute(s) and 5 second(s). ...	The video shows a man using a tire to clean a metal object. He sprays the tire with a cleaning solution and then uses it to scrub the metal object. After that, he uses a brush to clean the metal object.
Object	Summarize the video. Some details about the elements in this video. In this video 1 person appears from 4.00 seconds to 41.04 seconds. ... 1 donut appears from 180.85 seconds to 191.19 seconds. ...	The video shows a man cleaning a tire with a brush and a spray bottle. He then uses a star-shaped tool to remove the tire from the rim.
Action	Summarize the video. The following actions have been detected in the video. Action: changing wheel was detected from 2 second(s) to 7 second(s). ... Action: cleaning gutters was detected from 2 minute(s) and 56 second(s) to 2 minute(s) and 57 second(s). ...	The video shows a man changing a tire on a lawn mower. He uses a tool to remove the tire, cleans it, and then puts it back on the mower.

Table 11: Examples of prompts with mitigation and corresponding outputs (truncated for the sake of readability). VLLM: Video-LLaVA. Dataset: ActivityNet. Video: AwmHb44_ouw.mp4

Model	Rouge-1 Precision	Rouge-1 Recall	Rouge-1 F1	Rouge-2 Precision	Rouge-2 Recall	Rouge-2 F1	Rouge-L Precision	Rouge-L Recall	Rouge-L F1	SBERT
Video-ChatGPT	0.3786	0.0640	0.1080	0.0703	0.0117	0.0197	0.3058	0.0502	0.0851	0.3627
Video-LLaVA	0.4236	**0.1228**	**0.1808**	0.0720	**0.0206**	**0.0303**	0.3557	**0.0985**	**0.1463**	**0.4874**
Video-LLaMA2	**0.5219**	0.0673	0.1177	**0.1572**	0.0171	0.0302	**0.4621**	0.0583	0.1021	0.4761
VTimeLLM	0.4627	0.1017	0.1594	0.0694	0.0129	0.0209	0.3726	0.0788	0.1244	0.4752

Table 12: Plain text summaries. Rouge and BERTScore performance on ActivityNet video samples. Best per-metric scores are written in boldface.

Model	Rouge-1 Precision	Rouge-1 Recall	Rouge-1 F1	Rouge-2 Precision	Rouge-2 Recall	Rouge-2 F1	Rouge-L Precision	Rouge-L Recall	Rouge-L F1	SBERT
Video-ChatGPT	0.3557	0.2195	0.2566	0.0339	0.0217	0.0257	0.2582	0.1563	0.1825	0.3804
Video-LLaVA	0.3292	**0.3019**	0.3014	0.0655	**0.0609**	0.0615	0.2608	**0.2354**	0.2358	0.4301
Video-LLaMA2	**0.4381**	0.1738	0.2260	0.0778	0.0310	0.0405	**0.3312**	0.1240	0.1627	0.4211
VTimeLLM	0.3938	0.2975	**0.3289**	**0.0787**	**0.0609**	**0.0670**	0.2882	0.2134	**0.2367**	**0.5052**

Table 13: Plain text summaries. Rouge and BERTScore performance on EpicKitchens video samples. Best per-metric scores are written in boldface.

Model	Rouge-1 Precision	Rouge-1 Recall	Rouge-1 F1	Rouge-2 Precision	Rouge-2 Recall	Rouge-2 F1	Rouge-L Precision	Rouge-L Recall	Rouge-L F1	SBERT
Video-ChatGPT	0.6210	0.1742	0.2669	0.3298	0.0842	0.1317	0.5531	0.1523	0.2343	0.6354
Video-LLaVA	0.6068	**0.3079**	**0.3937**	**0.3496**	**0.1627**	**0.2125**	**0.5476**	**0.2750**	**0.3519**	**0.6879**
Video-LLaMA2	**0.6806**	0.1443	0.2313	0.3478	0.0640	0.1045	0.5983	0.1233	0.1984	0.6678
VTimeLLM	0.5816	0.2548	0.3411	0.2561	0.1038	0.1416	0.5043	0.2195	0.2946	0.6771

Table 14: Plain text summaries. Rouge and BERTScore performance on MSVD video samples. Best per-metric scores are written in boldface.

Model	Rouge-1 Precision	Rouge-1 Recall	Rouge-1 F1	Rouge-2 Precision	Rouge-2 Recall	Rouge-2 F1	Rouge-L Precision	Rouge-L Recall	Rouge-L F1	SBERT
Video-ChatGPT	0.4927	0.1530	0.2218	0.2180	0.0613	0.0901	0.4291	0.1275	0.1864	0.4247
Video-LLaVA	0.5299	**0.2501**	**0.3286**	0.2436	**0.1068**	**0.1430**	0.4660	**0.2156**	**0.2851**	0.6468
Video-LLaMA2	**0.6100**	0.1494	0.2351	**0.3406**	0.0719	0.1163	**0.5309**	0.1268	0.2005	**0.6553**
VTimeLLM	0.4889	0.2154	0.2875	0.1929	0.0737	0.1020	0.4238	0.1860	0.2484	0.6462

Table 15: Plain text summaries. Rouge and BERTScore performance on SumMe video samples. Best per-metric scores are written in boldface.

Model	Rouge-1 Precision	Rouge-1 Recall	Rouge-1 F1	Rouge-2 Precision	Rouge-2 Recall	Rouge-2 F1	Rouge-L Precision	Rouge-L Recall	Rouge-L F1	SBERT
Video-ChatGPT	0.3835	0.1456	0.2022	0.1153	0.0409	0.0583	0.3225	0.1188	0.1665	0.3209
Video-LLaVA	0.3635	**0.2602**	**0.2814**	0.1177	**0.0791**	**0.0858**	0.3068	**0.2104**	**0.2312**	0.5377
Video-LLaMA2	**0.4708**	0.1619	0.2306	**0.1445**	0.0460	0.0665	**0.3746**	0.1246	0.1792	0.5275
VTimeLLM	0.4091	0.2132	0.2649	0.1416	0.0691	0.0869	0.3448	0.1743	0.2193	**0.5413**

Table 16: Plain text summaries. Rouge and BERTScore performance on TVSum video samples. Best per-metric scores are written in boldface.