

SeriesBench: A Benchmark for Narrative-Driven Drama Series Understanding

Chenkai Zhang^{1,3*} Yiming Lei^{1,3*} Zeming Liu^{2†} Haitao Leng⁴ ShaoGuo Liu⁴
Tingting Gao⁴ Qingjie Liu^{1,3†} Yunhong Wang¹

¹State Key Laboratory of Virtual Reality Technology and Systems, Beihang University

²School of Computer Science and Engineering, Beihang University, Beijing, China

³Hangzhou Innovation Institute, Beihang University, Hangzhou China

⁴Kuaishou Technology

*Co-first authors: {zhangchenkai, ymlei}@buaa.edu.cn

†Corresponding authors: {zmliu, qingjie.liu}@buaa.edu.cn

A. More Details of SeriesBench

A.1. Task Dimension Definitions

Modern videos have diverse and intricate elements, including visuals, scripts, audio, and post-production enhancements. To facilitate a more comprehensive evaluation of large models that aligns with the diverse modalities present in contemporary videos [5, 10], we categorize the tasks into five major dimensions: Visuals, Script, Audio, Augmentation, and Comprehension.

1. Visuals. Visuals focus on understanding and analyzing visual content in the video. **1.1 Figures:** The model needs to analyze the figures appearing in the video: (1.1.1) Actions, analyzing what the figures are doing; (1.1.2) Interactions, understanding interactions between figures. **1.2 Scenes:** The model needs to recognize changes in scenes and spatiotemporal shifts: (1.2.1) Scene transitions, distinguishing between different scene changes; (1.2.2) Spatiotemporal shifts, understanding changes in time and space. **1.3 Objects:** The model should recognize and track the state of objects in the video: (1.3.1) Presence, confirming whether certain objects are present; (1.3.2) Interaction, analyzing how objects interact with characters.

2. Script. Script assesses the model’s understanding of the background, plot, and characters within the video. **2.1 Background:** The model needs to understand the setting of the story: (2.1.1) World-building, analyzing the overarching background framework constructed in the video; (2.1.2) Time and location, determining the time and place of the story. **2.2 Plot:** The model needs to analyze the development and complexity of the story: (2.2.1) Plot development, understanding how the story unfolds; (2.2.2) Foreshadowing and payoff, identifying and understanding the setup and resolution of plot elements; (2.2.3) Twists and conflicts, analyzing key turning points and conflicts in the story; (2.2.4)

Climaxes and build-ups, identifying the climax and its preceding buildup; (2.2.5) Suspense and continuity, analyzing the suspense and how different scenes transition; (2.2.6) Emotional dynamics, recognizing emotional peaks in the story. **2.3 Characters:** The model needs to analyze the characters in the story: (2.3.1) Reference, identifying characters in context; (2.3.2) Motivations, analyzing the reasons behind characters’ actions.

3. Audio. Audio elements assess the model’s understanding of sound-based information in the video, including dialogues, background music, and sound effects. **3.1 Dialogue:** The model needs to understand and analyze the details of dialogues: (3.1.1) Dialogue Attribution, matching dialogue with the character speaking; (3.1.2) Pronoun references, understanding the pronouns or references used in conversations; (3.1.3) Tone and emotion, analyzing the tone and emotional shifts in dialogue. **3.2 Music:** The model needs to analyze the role of music in the video: (3.2.1) Atmosphere, assessing how background music influences the emotional atmosphere of the scene. **3.3 Sound Effects:** The model should understand the role of sound effects: (3.3.1) Impact, analyzing how sound effects enhance a scene’s emotional or narrative aspects.

4. Augmentation. Modern videos are no longer purely composed of footage; many post-production elements, such as subtitles, labels, and special effects, are added. The model needs to understand and utilize this information to enhance video comprehension. **4.1 Subtitles:** The model needs to handle subtitles: (4.1.1) Recognition, accurately recognizing and understanding subtitle information. **4.2 Labels:** The model needs to understand the annotations in the video: (4.2.1) Purpose, analyzing how labels convey specific information. **4.3 VFX:** The model should analyze the use of special effects: (4.3.1) Effectiveness, evaluating how special effects influence the visual experience.

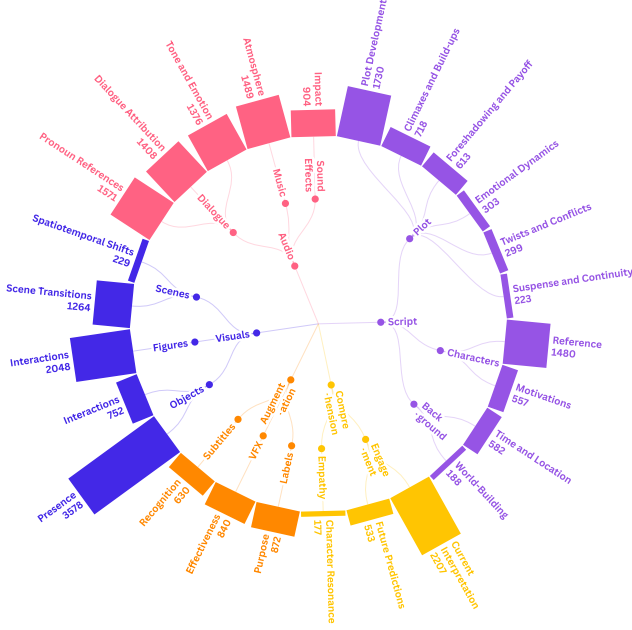


Figure 1. **Task Dimension in SeriesBench.** Detailed sample count for each task in SeriesBench.

5. Comprehension. Comprehension tasks evaluate the model’s overall grasp of the video. **5.1 Engagement:** The model needs to infer viewers’ interest in the plot’s development: (5.1.1) Future predictions, predicting future developments based on current information; (5.1.2) Current interpretation, assessing the model’s understanding of the current storyline. **5.2 Empathy:** The model needs to understand the emotional connection between the audience and the characters: (5.2.1) Character resonance, analyzing the emotional state of characters and generating empathy.

A.2. Data description

To obtain high-quality series data, we sourced series-format videos from the professional video platform, *Kuaishou*¹. To ensure higher-quality series, we selected content only from series created by authors with a substantial follower base, as well as videos meeting specific standards for likes and comments. Specifically, we only included series created by authors with more than 10 million followers, where each episode within the series maintained a minimum of 5,000 likes and at least 1,000 comments.

To further ensure a balanced representation across genres, we selected series relatively evenly from the top 11 most popular genres. Additionally, we controlled the number of episodes within each series, ranging from 2 to 50 episodes per series. This approach provided a diverse yet structured dataset. Ultimately, we curated a total of 105 series, encompassing 1,072 videos. For more detailed insights into the distribution, refer to Fig. 2.

¹<https://www.kuaishou.cn>



Figure 2. **Video Categories in SeriesBench.** The videos in SeriesBench are categorized into two main types: thematic videos and series videos. These encompass 11 of the most popular video themes, including Urban Life, Romance, Fantasy, Counterattack, Family, Ancient Style, Campus Life, Anime, Funny Daily, Short Drama, and Food. Each series is accompanied by a number indicating the total count of videos within that series.

B. More Details of Data Annotation

To ensure the quality of SeriesBench, we invited many professional data annotators to participate in the annotation process for SeriesBench. The process began with comprehensive training sessions, during which annotators were introduced to the task objectives and detailed annotation guidelines. Following this, a trial annotation phase was conducted to evaluate the annotators’ performance and select those most suited to the task. Based on the issues identified during the trial, additional training or personalized guidance was provided to address specific challenges and improve annotation quality. After these iterative steps, we finalized a team of 32 top-performing annotators who demonstrated the highest annotation quality for the main labeling task. Upon completion of the annotation, we conducted a rigorous quality inspection, during which annotations that failed to meet the required standards were either sent back for re-annotation or discarded entirely. Finally, a sampling test of the dataset showed that 96% of the annotations met our stringent requirements, confirming the high quality of the labeled data in SeriesBench.

B.1. Annotation Interface

To facilitate annotation and management, we designed a dedicated annotation interface, as illustrated in Fig. 3. At the top of the interface, information extracted from the

selected video is displayed, offering annotators a convenient reference for comparison and ensuring accurate labeling. On the left side, a hierarchical video index organizes content by genre at the primary level, enabling annotators to efficiently navigate and select video series within each category. The right side serves as the annotation workspace, where videos are meticulously analyzed and annotated across multiple dimensions. Each annotation entry includes precise timestamps, detailed descriptions of scenes or characters, and associated character portrait screenshots to ensure comprehensive labeling. After completing these elements, annotators select the most appropriate task category and craft a concise declarative statement sentence that integrates and contextualizes the annotated content, resulting in a cohesive and informative annotation output. Once the process is complete, annotators can save their work by clicking the “keep” button or reset the interface for the next task using the “clear” button, streamlining the workflow for subsequent annotations.

B.2. Manual annotation

As illustrated in Fig. 4, this example demonstrates a specific annotation process in detail. Once the annotator selects a video series, the interface expands to display all episodes within that series. The annotator is required to begin with the first episode and progress sequentially through to the final one, ensuring comprehensive coverage of the entire series. When the button for a particular episode is clicked, the interface redirects to the corresponding video playback page, where the annotator must watch the entire video before initiating the annotation process. Guided by the established annotation guidelines, the annotator meticulously documents critical aspects of the video, including timelines, character interactions, and major events. This process also involves capturing relevant screenshots of key characters to provide visual references that enrich the annotation.

After completing the detailed annotations, the annotator selects the most suitable task category that aligns with the content. In this example, “Plot development” was chosen, reflecting the emphasis on narrative progression and storyline evolution. As a final step, the annotator crafts a concise declarative statement sentence that synthesizes all the annotated content, ensuring that it reflects a well-connected summary and contains multi-step inferential insights. This approach guarantees that the resulting statement offers a high-information, comprehensive representation of the video content, enhancing the depth and quality of the benchmark annotations.

C. More Details of Heuristic Baselines

We provide further details on the heuristic baselines introduced in Section 5.2, inspired by MathVista [9]: *Random Choice*, *Frequent Guess*, and *Human Evaluation*. These

baselines are critical for comparing model performance on the SeriesBench tasks.

Random Choice. The Random Choice baseline selects an answer randomly from the answer pool for each question and averages the results over five trials. This baseline serves as a simple reference for evaluating model performance, representing the expected outcome of random guessing.

Frequent Guess. Based on the option distribution in each task category of the SeriesBench training set, we select the most frequently occurring option as the predicted answer for the corresponding task in the test set. This baseline demonstrates whether the option distribution in SeriesBench is balanced and serves as a straightforward yet informative reference for evaluating model performance, representing the expected outcome of consistently selecting the most common answer.

Human Evaluation. The human evaluation baseline reflects human performance on SeriesBench, serving as a reliable upper bound for assessing model capabilities. To facilitate this process, we developed a structured manual evaluation workflow with a user-friendly interface, illustrated in Fig. 5. Each evaluation session involves a random sampling of 10 videos from the test set. Upon selecting a video, the evaluator is presented with the corresponding question and a link to the indexed video. After reviewing the video, the evaluator submits their answer with a single click and completes the assessment. This streamlined workflow ensures efficiency and consistency, providing a robust basis for direct comparison with model-generated results.

D. More Details of Evaluation Settings

D.1. Prompt for Evaluation

We provide the detailed prompt template for evaluating the model’s performance on SeriesBench in Fig. 6. Additionally, we provide a prompt template for evaluating the model’s performance on multi-video tasks in Fig. 7, tailored to assess the model’s ability to analyze and synthesize information across multi-episode series. The model input is divided into five parts: `<frames>`, `<subtitles>`, `<theme-chara>`, `<prompt>`, and `<question>`. The `<frames>` part contains the video frames, the `<subtitles>` part includes the video subtitles, the `<theme-chara>` part involves the theme and characters of the video, the `<prompt>` part provides the instruction for the model, and the `<question>` part consists of the question for the model to answer. The guidelines for `<subtitles>` and `<theme-chara>` in Fig. 6 and Fig. 7 are optional and can be adjusted to match the input.

D.2. Model Inference Settings

GPT-4o and GPT-4o-mini Due to API limitations, we uniformly sampled 50 frames from each video for evalua-

Video Annotation System

Selected Videos

Video ID

Video Index

Urban Life

Magic

City

funny

antiquity

campus

Counterattack

family

Short Film

In love

Food Circle

Honor of Kings

Chicken Dinner Commentary

cartoon

Life is funny

Daily life

Duck Chaser uses the default channel

NULL

Suspense

Film and TV Comprehensive

Friendship

rural

gourmet food

High Sweetness

Positive Energy

The Sims

Youth Inspiration

Selected video information

Dimension Input

Time coordinate 1

Enter the time coordinate (such as 00:10)

Annotation Text 1

Enter the label text

Upload image 1

Drag and drop images here

-or-

Click Upload

Time coordinate 2

Enter the time coordinate (such as 00:10)

Label Text 2

Enter the label text

Upload image 2

Drag and drop images here

-or-

Click Upload

Time coordinate 3

Enter the time coordinate (such as 00:10)

Annotation Text 3

Enter the label text

Upload image 3

Drag and drop images here

-or-

Click Upload

Time coordinate 4

Enter the time coordinate (such as 00:10)

Annotation Text 4

Enter the label text

Upload image 4

Drag and drop images here

-or-

Click Upload

Time coordinate 5

Enter the time coordinate (such as 00:10)

Annotate Text 5

Enter the label text

Upload image 5

Drag and drop images here

-or-

Click Upload

Select Task

Overall statement

Overall statement

Please enter an overall statement

Save annotations

keep

Save the results

Clear

Figure 3. Illustration of the web-based annotation interface. Displaying parsed video information for reference, a hierarchical video index for navigation, and an annotation area for detailed content analysis and categorization.

Video Annotation System

Selected Videos

Episode 11112627682374

Video ID

112627682374

Video Index

Urban Life

Magic

City

funny

antiquity

campus

Counterattack

family

Short Film

in love

Food Circle

Honor of Kings

Chicken Dinner Commentary

cartoon

Life is funny

Daily life

Duck Chaser uses the default channel

Seniors Talking About Cars

Little rubber band and monitor

I'll wait for you at the end of time

Wife, don't run away

Perfect Diary

The Queen of Zhuyin

The most complete guide to Disneyland

My wife is rich

my mom

My savior

Desperate moments in school

Episode 1

Episode 2

Episode 3

Episode 4

Episode 5

Selected video information

Dimension Input

Time coordinate 1

00:05

Annotation Text 1

Yingwen passed the exam.

Time coordinate 2

00:13

Label Text 2

The director agreed to install an air conditioner.

Time coordinate 3

00:24

Annotation Text 3

Zhaodezhu wants to turn on the air conditioner.

Time coordinate 4

00:36

Annotation Text 4

The director established detailed rules for the use of the air conditioner.

Time coordinate 5

00:44


Annotate Text 5

The director emphasized that the air conditioner cannot be turned on and then left the classroom.

Upload image 1



Upload image 2



Upload image 3



Upload image 4



Upload image 5



Select Task

Plot Development

Overall statement

Overall statement

The director agreed to install an air conditioner for Yingwen but set unattainable rules to restrict students from using it.

Save annotations

keep

Save the results

Annotations: 112627682374_20241018_200923.txt

Clear

Figure 4. **Example of the annotation process.** Showing the selection of a video series, sequential annotation of episodes, and detailed documentation of key events, timelines, and character interactions, culminating in task categorization and a synthesized declarative summary to enhance content representation.

Video Test System

Video Information

Love Secrets_Episode 12 (Photo ID: 21019681755)

[Click here to watch the video](#)

Save the results

Q1: Which sentence indicates that three years have passed in the video?

☐ A. It has been nearly four years.

☐ B. Looking back, what happened two years ago is still vivid in my mind.

☐ C. means it is now three years later.

☐ D. Five years have passed since today.

Q2: Which of the following descriptions clearly states that three years have passed?

☐ A. Time flies, looking back it has been four years.

☒ B. It means it is now three years later.

☐ C. Two years later, we finally met again.

☐ D. Time flies, as if it was yesterday.

Q3: Which point clearly shows that the plot of the video has jumped to three years later?

☒ A. What happened three months ago seems to be happening right before my eyes.

☐ B. Four years have passed since then.

☐ C. means it is now three years later.

☐ D. Looking back over the past five years, I feel deeply moved.

Q4: Which sentence in the video indicates that the story has developed to three years later?

☐ A. It has been three months since then.

☒ B. It has been about four years since the incident.

☐ C. means it is now three years later.

☐ D. Five years have passed since then.

Q5: What are the scenes in the video?

☐ A. Chairman's office, hotel room, garage

☒ B. The chairman's home, the train station, and the hospital three years ago

☐ C. Chairman's Office, Meeting Room, Restaurant

☐ D. The chairman's home, office, and car three years ago

Q6: In the video, which location was the chairman's home three years ago?

☐ A. Garage

☐ B. Office of the Chairman

☒ C. Meeting Room

☐ D. The chairman's home three years ago

Q7: Which place is included in one of the three scenes shown in the video?

☐ A. Meeting Room

☒ B. The chairman's home three years ago

☐ C. Station

☐ D. Restaurant

Q8: What locations are included in the scene changes in the video?

☒ A. In the car, the chairman's office, the meeting room

☐ B. The chairman's home, office, and car three years ago

☐ C. Restaurant, Chairman's Office, Station

☐ D. Chairman's office, garage, and the chairman's home three years ago

Q9: What scenes appear in the video?

☐ A. People who come for interviews

☒ B. Boys

☐ C. Chairman

☐ D. Boss

Q10: What preparations did the secretary suggest the chairman make, and what impact might these preparations have on the subsequent plot?

☐ A. Three

☐ B. Four

☒ C. Five

☐ D. Two

Video Index

All Videos

The sister who delivers food in a Rolls-Royce_Episode 4

Meet My Ex-husband Again After Three Years_Episode 2

Love Secrets_Episode 12

The Difference Between a Good Student and a Bad Student_Episode 1

Love Secrets_Episode 3

Embarrassing Things at Workplace_Episode 2

Love Secrets_Episode 9

Love Secrets_Episode 19

The sister who delivers food in a Rolls-Royce_Episode 3

The Secret of the Heart Secretary_The Finale

answer the questions

Submit Answer

Clear answer

Figure 5. **The manual evaluation interface.** Comparing model results with human performance, displaying randomly selected videos, corresponding questions, and indexed video links for evaluators to review and submit their responses.

Prompt for SeriesBench Evaluation

System Prompt

You are an AI visual assistant specialized in analyzing videos.

Task Description

Your task is to analyze the given video content and identify its narrative elements (e.g., events, characters, plot development, emotions, causal relationships, etc.). Based on the provided question, generate accurate answers that align with the video’s narrative logic. Carefully examine the video clips and provide answers derived from video frames, audio features, and subtitle content.

GUDGES

1. Video Frames: Video frames are uniformly sampled from the original video, representing its key information. Utilize these frames effectively to extract elements such as plot development, key scenes, or character actions.
2. Subtitles (Optional): Subtitles are generated from the video audio using a Speech-to-Text (STT) method and may contain noise or irrelevant content. Filter out distractions and focus on subtitle content relevant to the question.
3. Themes and Characters (Optional): Theme is a brief introduction to the current drama, and Characters include appearance descriptions of all characters appearing in the video within the drama.
4. Question Types and Answer Formats:
 - a) Multiple Choice Questions: Choose the most suitable answer from options A, B, C, or D.
 - b) True/False Questions: Select either T (True) or F (False) as the answer.
 - c) Open-Ended Questions: Provide a concise and accurate sentence as the answer, ensuring logical clarity and consistency with the video content.

Input

Question Type: {type}

Question: {question}

Options: {options}

Please strictly adhere to the above notes and answer questions based on the video content.

Figure 6. The prompt template adopted for Evaluation on SeriesBench.

tion on SeriesBench, testing under both “without subtitles” and “with subtitles” settings. The model input adopts the format of “<frames> + <subtitles>(optional) + <theme-chara>(optional) + <prompt> + <question>”.

VideoLLaMA2.1-AV We investigated how audio influences model performance by extracting the audio tracks from the videos and incorporating them into the model’s input. The model input adopts the format of “<frames> + <subtitles>(optional) + <audio>(optional) + <prompt> + <question>”.

Other Open-Source Video-MLLMs We adhere to the official inference strategies of these MLLMs. To ensure stable inference for Qwen2-VL, where we observed occasional CUDA-OOM errors under certain conditions, we set the number of input frames to 64 instead of the default setting of 1fps. The model input adopts the format of “<frames> + <subtitles>(optional) + <theme-chara>(optional) + <prompt> + <question>”.

D.3. Evaluation Metrics

MultiChoice and Judgment Type Tasks. All tasks are formulated as multiple-choice questions, and we adopt accuracy as the primary evaluation metric.

Open-Ended Type Tasks. For open-ended questions, where the model’s responses are required to be concise sentences containing the correct answer, we adopt BLEU-2 [13], METEOR [1], and BERTScore F1 [19] as evaluation metrics to assess model outputs from both lexical and semantic perspectives [3, 8, 15]. **BLEU-2** evaluates the bigram (two-word phrase) overlap between generated and reference texts, reflecting the degree of lexical similarity and fluency. **METEOR**, widely used in machine translation and text generation, considers word-level matches, including stemming and synonym matching, providing a more nuanced and comprehensive evaluation. **BERTScore F1** measures semantic similarity by calculating the cosine similarity between the embeddings of the generated and reference sentences in the BERT encoding space, offering an accurate assessment of sentence-level semantic alignment.

Prompt for MultiEpisode Series Evaluation

System Prompt

You are an AI visual assistant specialized in analyzing videos.

Task Description

This is a video sequence from a TV series, including the current episode and additional episodes, input in the logical order of the series. The additional episodes are denoted as 'Previous_i' and 'Next_i', representing the i-th episode before and after the current episode, respectively.

Your task is to analyze the given video content and identify its narrative elements (e.g., events, characters, plot development, emotions, causal relationships, etc.). Based on the provided question, generate accurate answers that align with the video's narrative logic. Carefully examine the video clips and provide answers derived from video frames, audio features, and subtitle content.

GUDGES

1. Video Frames: Video frames are uniformly sampled from the original video, representing its key information. Utilize these frames effectively to extract elements such as plot development, key scenes, or character actions.
2. Subtitles (Optional): Subtitles are generated from the video audio using a Speech-to-Text (STT) method and may contain noise or irrelevant content. Filter out distractions and focus on subtitle content relevant to the question.
3. Themes and Characters (Optional): Theme is a brief introduction to the current drama, and Characters include appearance descriptions of all characters appearing in the video within the drama.
4. Question Types and Answer Formats:
 - a) Multiple Choice Questions: Choose the most suitable answer from options A, B, C, or D.
 - b) True/False Questions: Select either T (True) or F (False) as the answer.
 - c) Open-Ended Questions: Provide a concise and accurate sentence as the answer, ensuring logical clarity and consistency with the video content.

Input

Question Type: {type}

Question: {question}

Options: {options}

Please strictly adhere to the above notes and answer questions based on the video content.

Figure 7. The prompt template adopted for Multi-Episode series tasks.

E. Implement of Method

E.1. Retriever module of PC-DCoT

We developed a Retriever module to extract event- and character-related timelines from video frames, aiming to better understand the narrative structure in series. Specifically, leveraging the manually annotated event and character timelines in the training set of SeriesBench, we selected the corresponding video frames to construct a dataset consisting of 6,046 image-text pairs. To better adapt to Chinese annotations, we refined the module using CN-CLIP_{VIT-H/14} [17] as the base model and finetuned it on the constructed dataset. The training hyperparameters are detailed in Tab. 1.

E.2. Mathematical representation of PC-DCoT

The input to our task consists of two primary components: a video sequence and a set of questions. The video is represented as a collection of frames denoted by

Hyperparameter	config
context length	52
warmup steps	100
batch size	32
learning rate	5e-5
weight decay	0.001
training epochs	5

Table 1. Hyperparameters for CN-CLIP_{VIT-H/14} finetuning.

$$V = \{f_1, f_2, \dots, f_k\} \quad (1)$$

where each element f_i corresponds to an individual frame within the video sequence. Additionally, the input includes a set of questions Q that are intended to guide the analysis and understanding of the video content.

For narrative-driven videos, understanding the plot and characters is essential, as they serve as the key drivers of the storyline. Our method begins by utilizing the Multimodal Large Language Model (MLLM) to process raw video frame input $V = \{f_1, f_2, \dots, f_k\}$ alongside relevant questions Q . The MLLM's role is to extract and identify

key narrative elements, specifically focusing on significant events that are pertinent to the questions and key characters that appear within the narrative context. This extraction process results in two distinct sets: a set of events and a set of characters, both represented as text-based descriptions, represented as

$$\{\mathcal{E}_1, \mathcal{E}_2, \dots, \mathcal{E}_m\}, \{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_n\} = \text{MLLM}_{\text{extract}}(Q, V) \quad (2)$$

where $\{\mathcal{E}_1, \mathcal{E}_2, \dots, \mathcal{E}_m\}$ denotes the extracted events and $\{\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_n\}$ denotes the identified characters.

In narrative-driven videos, character appearances often exhibit discontinuity, whereas events typically unfold through cohesive and interconnected sequences. To align video content with task-relevant events and characters, our approach leverages a video clip model to identify and isolate key scenes. When processing raw video input in conjunction with extracted events related to the given questions, the model searches for frames corresponding to each identified event, represented by

$$\mathcal{F}_{e_j} = \{f_i \in V \mid \text{CLIP}_{\text{event}}(f_i, \mathcal{E}_j) \geq \theta_e\} \quad (3)$$

where $\text{CLIP}_{\text{event}}$ measures the relevance between each frame f_i and event \mathcal{E}_j using a defined threshold θ_e . To maintain temporal coherence, these frames are combined with neighboring frames within a temporal window δ , forming cohesive sets of frames for each event, represented by

$$E_j = \{f_{i'} \in V \mid \exists f_i \in \mathcal{F}_{e_j}, |i - i'| \leq \delta\} \quad (4)$$

The overall set of event sequences is then expressed as

$$\mathcal{F}_e = \bigcup_{j=1}^m E_j \quad (5)$$

In parallel, character tracking is performed by locating and retrieving frames where specific characters appear, based on their portraits. This process identifies frames relevant to each character \mathcal{C}_j , represented by

$$\mathcal{C}_j = \{f_i \in V \mid \text{CLIP}_{\text{character}}(f_i, \mathcal{C}_j) \geq \theta_c\} \quad (6)$$

with $\text{CLIP}_{\text{character}}$ determining frame relevance to the character using a threshold θ_c . The aggregated frames for all characters are represented as

$$\mathcal{F}_c = \bigcup_{j=1}^n \mathcal{C}_j \quad (7)$$

To effectively capture the narrative structure and character dynamics, we construct two distinct chains: the Plot Event Chain of Thought (CoT_E) and the Character Temporal Chain of Thought (CoT_C).

The CoT_E is created by generating detailed descriptions for each event set E_j derived from the aggregated event frames \mathcal{F}_e . This process is expressed as

$$\text{CoT}_E = \bigcup_{j=1}^m \left\{ d_j \mid d_j = \text{MLLM}_{\text{describe-event}}(E_j), E_j \in \mathcal{F}_e, \right. \\ \left. T_j = [t_j^{\text{start}}, t_j^{\text{end}}], \text{ where } T_j \text{ denotes the time interval of } E_j \right\} \quad (8)$$

Here, $\text{MLLM}_{\text{describe-event}}$ generates a narrative description d_j for each event E_j , and $T_j = [t_j^{\text{start}}, t_j^{\text{end}}]$ represents the time interval of the event, capturing its temporal boundaries within the video timeline.

Simultaneously, the CoT_C is constructed by describing the behaviors and appearances of each character \mathcal{C}_k based on the frames aggregated in \mathcal{F}_c . This is represented as

$$\text{CoT}_C = \bigcup_{k=1}^n \left\{ d_k \mid d_k = \text{MLLM}_{\text{describe-character}}(\mathcal{C}_k), \right.$$

$$\left. \mathcal{C}_k \in \mathcal{F}_c, T_k = \{t_{k_1}, t_{k_2}, \dots, t_{k_l}\}, \right.$$

$$\left. \text{where } T_k \text{ is the sequence of frames where } \mathcal{C}_k \text{ appears} \right\} \quad (9)$$

This dual-chain construction captures both the continuous flow of events and the potentially scattered yet significant appearances of characters, providing a comprehensive understanding of narrative progression and character interactions.

To merge the Plot Event Chain of Thought (CoT_E) and the Character Temporal Chain of Thought (CoT_C) into a unified framework, we utilize the precise temporal intervals associated with events and character appearances. Since each event d_j in CoT_E corresponds to a time interval $[t_j^{\text{start}}, t_j^{\text{end}}]$, it is possible to identify characters d_k from CoT_C whose presence overlaps with this interval, represented by the condition $T_k \cap [t_j^{\text{start}}, t_j^{\text{end}}] \neq \emptyset$. This temporal alignment enables us to determine which characters interact within the timeframe of each event.

The merging process is defined as:

$$\text{PC-DCoT} = \bigcup_{j=1}^m \{d'_j \mid d'_j = \text{MLLM}_{\text{aggregate}}(d_j,$$

$$\{d_k \in \text{CoT}_C \mid T_k \cap [t_j^{\text{start}}, t_j^{\text{end}}] \neq \emptyset\}, d_j \in \text{CoT}_E\} \quad (10)$$

Here, $\text{MLLM}_{\text{aggregate}}$ combines each event d_j from CoT_E with relevant character descriptions d_k from CoT_C that appear within the event’s time interval. This aggregation synthesizes the behaviors and occurrences of characters with the narrative context of the event, resulting in a cohesive and enriched representation of the narrative flow and character dynamics.

E.3. Prompt for PC-DCoT

In the mathematical formulation of PC-DCoT presented above, $\text{MLLM}_{\text{extract}}$, $\text{MLLM}_{\text{describe-event}}$, $\text{MLLM}_{\text{describe-character}}$, and $\text{MLLM}_{\text{aggregate}}$ denote the reasoning processes executed by the MLLM. The corresponding prompts utilized to guide these reasoning processes are depicted in Figs. 8, 9, 10, and 11.

F. More Experimental Results

In this section, we give more detailed results about the performance of different models on SeriesBench.

F.1. Main Results with Subtitles Ablation

The comprehensive results under the subtitles ablation setting are shown in Tab. 2. Notably, open-source models, particularly Qwen2-VL, exhibit competitive performance compared to the closed-source model GPT-4o. On the Multi-Choice and Judgment tasks of SeriesBench, Qwen2-VL secures 5 first-place, 5 second-place, and 1 third-place ranking, closely rivaling GPT-4o, which achieves 7 first-place, 3 second-place, and 1 third-place finishes. These results underscore the effectiveness and potential of open-source models in addressing narrative-driven video analysis.

F.2. Performance on Multi-Episode Tasks

Tab. 3 presents the performance of various models on multi-episode tasks. We observe that model performance with multi-episode inputs is consistently lower compared to single-episode inputs, highlighting the difficulty in comprehending narrative structures and character dynamics within series. Addressing these limitations requires advancements in modeling long-term dependencies and context-aware reasoning, which remain critical areas for future research.

F.3. Fine-Grained Task Results

Tab. 4 presents more results of different fine-grained tasks in SeriesBench. For the tasks that are strongly related to the plot (a) (b) (c) (d) (e) (f) and those strongly related to characters (i) (u) (s) (t), the top three models in most cases utilize the PC-DCoT framework. This demonstrates the framework’s effectiveness in handling tasks closely related to narrative and plot elements, showcasing its ability to capture intricate relationships within the storyline, manage complex contextual dependencies, and enhance comprehension of plot progression.

F.4. Qualitative Results

More additional qualitative results can be found in Figs. 12 and 13. Figs. 12 showcases model reasoning performance on multiple-choice and true/false tasks across different task dimensions. These examples highlight the necessity for narrative comprehension when tackling questions within our SeriesBench. It is evident that even state-of-the-art Multimodal Large Language Models (MLLMs) struggle to provide correct answers for certain questions, underscoring the complexity of the tasks. However, by incorporating the PC-DCoT framework, model performance shows notable improvement, demonstrating enhanced comprehension and reasoning capabilities when addressing narrative-driven tasks.

Fig. 13 illustrates the intermediate reasoning process and final answers of MLLMs when addressing open-ended questions using the PC-DCoT framework. The results reveal that integrating the PC-DCoT framework enables MLLMs to perform deeper analyses of events and individuals, enhancing their ability to draw nuanced conclusions. This improvement allows the model to tackle more complex problems effectively, demonstrating the framework’s capability to systematically guide reasoning processes toward more accurate and comprehensive outcomes.

Model	Size	Frames	VS		SC		AU		AG		CO		Overall		BL-2		MET		F1 _{BERT}	
			w/o s.	w/ s.	w/o s.	w/ s.	w/o s.	w/ s.	w/o s.	w/ s.	w/o s.	w/ s.	w/o s.	w/ s.	w/o s.	w/ s.	w/o s.	w/ s.	w/o s.	w/ s.
Heuristics baselines																				
Random Choice	-	-	39.3		38.2		35.5		36.5		38.8		37.7		-	-	-	-	-	-
Frequent Guess	-	-	43.6		46.0		43.1		41.1		50.6		44.4		-	-	-	-	-	-
Human	-	-	98.2		94.4		94.6		97.2		92.6		95.8		-	-	-	-	-	-
Open-source Video MLLMs																				
InternVL2[14]	7B	32	50.2	52.0	51.3	56.8	52.6	57.7	73.6	78.6	55.1	57.7	55.2	59.2	10.11	8.43	23.68	21.22	69.93	68.40
LLaVA-OneVision[6]	7B	32	40.9	51.5	39.3	54.0	43.3	56.2	44.5	70.6	45.5	59.6	42.0	56.9	6.94	7.30	19.67	20.20	67.03	67.82
LLaVA-Video[20]	7B	64	47.8	54.9	48.7	56.8	46.7	57.4	50.2	71.2	49.4	51.9	48.3	58.3	6.50	7.27	17.35	19.06	66.59	67.95
Qwen2-VL[16]	7B	64	52.9	55.7	55.0	57.5	56.4	58.6	72.2	75.3	57.7	59.6	57.7	60.3	12.3	11.41	29.12	27.97	71.53	70.71
MiniCPM-V 2.6[18]	8B	64	48.4	53.3	53.1	57.3	54.7	57.7	73.2	76.6	56.4	55.1	55.6	59.1	9.39	8.57	30.69	30.06	69.53	68.66
Aria[7]	8x3.5B	128	48.2	52.4	50.1	56.6	45.3	51.8	64.2	75.6	46.8	50.0	50.5	56.8	9.25	10.00	23.25	26.08	69.10	70.35
VITA[4]	8x7B	32	41.6	46.9	44.1	46.0	47.2	53.3	59.9	68.9	42.9	48.1	46.5	51.8	8.65	9.32	25.06	27.08	68.81	68.93
Open-source Video-Audio MLLMs																				
VideoLLaMA2.1-AV[2] +audio	7B	32	41.2	47.4	42.3	52.2	43.3	53.3	36.8	66.6	35.9	49.4	40.8	53.1	6.68	7.61	21.67	23.34	66.02	67.53
			39.1	45.3	40.9	52.2	42.6	51.3	34.8	66.2	32.1	46.2	39.0	51.7	6.17	7.24	20.25	22.71	64.85	67.09
Closed-source MLLMs																				
GPT-4o-mini[11]	N/A	50	40.1	46.7	34.4	42.7	40.4	47.9	54.5	70.9	41.0	44.9	41.3	49.8	9.34	9.99	26.79	29.30	68.35	68.76
GPT-4o[12]	N/A	50	52.4	55.8	56.4	62.8	56.0	60.6	72.2	79.9	47.4	59.6	56.9	62.8	7.45	9.61	22.02	25.10	67.74	68.94

Table 2. **Performance of MLLMs on SeriesBench.** Size means the LLM size. Judgement and multichoice metrics Accuracy and open-ended metrics BLEU-2(**BL-2**), METEOR (**MET**), and BERTScore F1 (**F1_{BERT}**) are reported in percentage (%), evaluated under two settings: “without subtitles” (**w/o s.**) and “with subtitles” (**w/ s.**). “-” indicates that results are not feasible with open-ended metrics in heuristic baselines. The best, second-best, and third-best results are marked **purple**, **orange**, and **gray**, respectively.

Model	Episodes	VS	SC	AU	AG	CO	Overall	BL-2	MET	F1 _{BERT}
InternVL2	-	52.0	56.8	57.7	78.6	57.7	59.2	8.43	21.22	68.40
	Prev ₂	50.7	55.2	58.4	73.6	58.3	57.8	9.66	23.16	69.36
	Prev ₁	50.5	57.7	59.1	73.2	57.1	58.4	9.37	22.1	68.81
	Next ₁	50.9	55.4	57.4	76.9	59.6	58.4	8.81	21.57	68.67
	Next ₂	49.1	56.4	58.4	73.9	57.1	57.6	8.33	20.89	68.48
Qwen2-VL	-	55.7	57.5	58.6	75.3	59.6	60.3	11.41	27.97	70.71
	Prev ₂	51.6	58.2	58.6	71.2	60.9	58.7	10.58	27.02	70.01
	Prev ₁	54.9	58.0	58.4	71.2	59.0	59.4	10.91	27.35	70.36
	Next ₁	54.6	58.9	56.9	72.2	59.0	59.3	10.83	27.13	70.39
	Next ₂	52.9	58.0	56.0	72.2	59.6	58.5	10.78	26.91	70.28
MiniCPM-V 2.6	-	53.3	57.3	57.7	76.6	55.1	59.1	8.57	30.06	68.66
	Prev ₂	48.7	52.2	56.9	69.2	50.0	54.8	7.88	29.26	68.33
	Prev ₁	48.0	52.0	55.5	71.2	53.2	54.8	7.77	29.52	68.42
	Next ₁	48.2	52.2	50.4	70.2	48.1	53.2	7.97	30.01	68.32
	Next ₂	46.4	52.2	54.0	61.2	48.7	52.0	7.63	29.47	68.17

Table 3. **Performance of Top-Performing MLLMs on Multi-Episode Series Tasks.**

References

- [1] Satantjeev Banerjee and Alon Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005. 7
- [2] Zesen Cheng, Sicong Leng, Hang Zhang, Yifei Xin, Xin Li, Guanzheng Chen, Yongxin Zhu, Wenqi Zhang, Ziyang Luo, Deli Zhao, and Lidong Bing. Videollama 2: Advancing spatial-temporal modeling and audio understanding in video-llms. *arXiv preprint arXiv:2406.07476*, 2024. 11
- [3] Data Science in Your Pocket. Llm evaluation metrics explained, 2024. Accessed: 2024-11-22. 7
- [4] Chaoyou Fu, Haojia Lin, Zuwei Long, Yunhang Shen, Meng Zhao, YifanZhang, Shaoqi Dong, Xiong Wang, Di Yin, Long Ma, Xiawu Zheng, Ran He andRongrong Ji, Yunsheng Wu, Caifeng Shan, and Xing Sun. Vita: Towards open-source interactive omni multimodal llm, 2024. 11
- [5] Megan Fulwiler and Kim Middleton. After digital storytelling: Video composing in the new media age. *Computers and Composition*, 29(1):39–50, 2012. 1
- [6] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer, 2024. 11
- [7] Dongxu Li, Yudong Liu, Haoning Wu, Yue Wang, Zhiqi Shen, Bowen Qu, Xinyao Niu, Guoyin Wang, Bei Chen, and Junnan Li. Aria: An open multimodal native mixture-of-experts model, 2024. 11
- [8] Yu Li, Shenyu Zhang, Rui Wu, Xiutian Huang, Yongrui Chen, Wenhao Xu, Guilin Qi, and Dehai Min. Mateval: A multi-agent discussion framework for advancing open-ended text evaluation, 2024. 7
- [9] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel

Prompt for Target Extraction(MLLM_{extract})

Extract characters and events related to the given question and video frames, and organize them into a structured JSON format.

Extraction Details

1. Characters:

1. Identify all characters explicitly mentioned in the question.
2. Include any implicitly or indirectly referenced characters related to the question.
3. Provide names and describe their involvement in the events.

2. Events:

1. Clearly describe the core events related to the question.
2. Include secondary events from the video frames that provide context or additional clues.
3. List events in logical or chronological order, maintaining a clear cause-and-effect chain.

3. Background Information (Optional):

1. If the question or video frames include additional background information, such as scene descriptions or timestamps, integrate them as part of the event details.

Output Format

Use the following JSON structure for the output:

- Characters: List all relevant characters, ordered by importance or sequence of mention.
- Events: Use numbered keys to describe each event in detail. Arrange events in logical or chronological order.

```
{
  "Characters": ["[Character1]", "[Character2]", "..."],
  "Events": {
    "1": "[Event Description 1]",
    "2": "[Event Description 2]",
    "...": "[Additional Event Descriptions]"
  }
}
```

Example Input

Question: Who put the ginger slices in the director's cup?

Options: A: Tuojiang Kai B: Zhao Dezhu C: Sima Yi D: Yingyan

Example Output

```
{
  "Characters": ["Director", "Tuojiang Kai", "Zhao Dezhu", "Sima Yi", "Yingyan"],
  "Events": {
    "1": "Someone prepared ginger tea or a similar beverage, during which ginger slices were added.",
    "2": "A person handed a cup containing ginger slices to the director, potentially indicating when the slices were added.",
    "3": "The director's reaction while drinking the beverage may suggest whether they were aware of the ginger slices."
  }
}
```

Important Considerations

1. Completeness:

1. Extract all characters mentioned or implied in the question.
2. Ensure events cover all necessary details based on the video frames and the question.

2. Logical Flow:

1. Number events in logical or chronological order to maintain clarity.

3. Exclude Irrelevant Information:

1. Focus only on details directly relevant to the question.

4. Detail Clarity:

1. Describe events as thoroughly as possible, including actions, background, and potential intentions.

Figure 8. Prompt for Target Extraction.

Prompt for Plot Event Chain (MLLM_{describe – event})

Based on the input event descriptions, time ranges, and video frames, generate detailed event information. The output should include the start and end times of each event, a brief description of the event, detailed.

Describe Details

1. Event Description:

1. Generate a clear and concise description for each event, summarizing the core content of the event.
2. Ensure the description aligns with the input and includes all critical information.

2. Time Range:

1. Each event must include a precise time range in the format MM:SS - MM:SS.

3. Detailed Information:

1. For each event, generate the following fields:
 - **action:** Describe the key actions taking place during the event.
 - **scene:** Provide a detailed depiction of the environment, location, or behavior of characters.
 - **dialogue:** Include any dialogue in the event. If no dialogue exists, explicitly state "No dialogue. "
2. These fields should be presented in a list, allowing for multiple actions or scenes per event.

4. Output Format:

1. The result must be output in standard JSON format.
2. Each event in the JSON output must include:
 - **time:** The time range of the event in the format MM:SS - MM:SS.
 - **description:** A brief summary of the event.
 - **details:** Detailed information about the event, including actions, scenes, and dialogues.

Output Format

Use the following JSON structure for the output:

```
{
  "Event1": {
    "time": "[Start time] - [End time]",
    "description": "[Brief description of the event]",
    "details": {
      "action": "[Description of actions]",
      "scene": "[Description of the scene]",
      "dialogue": "[Dialogue content or 'No dialogue']"
    }
  },
  ...
}
```

Example Input

```
{
  "Event1": {
    "description": "Someone prepared ginger tea or a similar beverage, during which ginger slices were added.",
    "frames": ["frame_106.png", "...", "frame_115.png"],
    "time": "01:15 - 01:25"
  },
  ...
}
```

Example Output

```
{
  "Event1": {
    "time": "01:15 - 01:25",
    "description": "During the preparation of ginger tea, someone added ginger slices to the cup.",
    "details": {
      "action": "Someone is chopping ginger slices and adding them to the cup.",
      "scene": "In the classroom, a cup is placed on the table alongside freshly chopped ginger slices.",
      "dialogue": "This is ginger tea for the Director."
    }
  },
  ...
}
```

Figure 9. Prompt for Plot Event Chain.

Prompt for Character Temporal Chain (MLLM_{describe – character})

Based on input frame data and timestamps, generate a complete timeline for each character. The timeline must cover all periods when the character appears in the video, detailing actions, dialogue, and corresponding video frame information.

Describe Details

1. Character Identification:

1. Extract character names and related information from the input, including portrait descriptions (e.g., appearance, clothing, etc.).
2. Identify all timestamps and frames where each character appears.

2. Time Period Segmentation:

1. Organize the character's appearance data by grouping discrete or continuous frames and timestamps into specific time periods (defined by start and end frames).
2. Ensure no overlap or omission between time periods, covering all relevant frames.

3. Action and Scene Descriptions:

1. For each time period, provide a detailed description of the character's actions and behaviors.
2. Action descriptions must be inferred from the frame data and context, providing clear and specific details about the character's activities.

4. Dialogue Annotation:

1. Record the character's dialogue for each time period, or state "No dialogue" if absent.

5. Output Structured Data:

1. Output each character's timeline in JSON format, ensuring clarity and completeness.
2. The output must include the character's name, time periods with start and end frames, action descriptions, and dialogue.

Output Format

Use the following JSON structure for the output:

```
{
  "Character": "[Character Name]",
  "Time Periods": [
    {
      "time": "[Start time] - [End time]",
      "action": "[Action Description]",
      "dialogue": "[Dialogue Content or 'No dialogue']"
    },
    ...
  ],
  ...
}
```

Example Input

```
{
  "Director": {
    "portrait": "A portly man wearing a white striped shirt.",
    "frames": ["frame_002.png", "...", "frame_125.png"],
    "time": [00:02, 00:03, 00:10, ..., 02:05]
  }, ...
}
```

Example Output

```
{
  "Character": "Director",
  "Time Periods": [
    {
      "time": "00:02 - 00:03",
      "action": "Enter the room.",
      "dialogue": "No air conditioning allowed!"
    },
    ...
  ],
  ...
}
```

Figure 10. Prompt for Character Temporal Chain.

Prompt for Constructing the PC-DCoT (MLLM_{aggregate})

Based on the input plot event and character temporal chains, generate a comprehensive and detailed narrative description. The output should elaborate on the development of each event, including the progression of the plot, interactions between characters, specific actions, and dialogue. Incorporate temporal and spatial cues to ensure logical coherence.

Aggregate Details

1. Expand Event Chains:

1. Develop each event with a detailed account of its cause, progression, and outcome.
2. Maintain logical or causal relationships between events to create a cohesive storyline.

2. Describe Characters and Interactions:

1. Include detailed actions, expressions, emotions, and attitudes of all involved characters.
2. Highlight interactions between characters, including verbal exchanges, gestures, and implicit behaviors.

3. Incorporate Dialogue:

1. Recreate natural dialogue relevant to the scene, showcasing the tone, intent, and personality of the characters.
2. Ensure dialogue drives the narrative forward or reveals additional character traits or motives.

4. Add Contextual Details:

1. Use descriptive elements to recreate the environment, mood, and setting of the events.
2. Clarify the time and place of each event using timestamps, spatial markers, or visual cues provided in the input.

Output Format

Use the following JSON structure for the output:

```
{
  "Event1": {
    "time": "[Start time] - [End time]",
    "description": "[Detailed description of the event]"
  }, ...
}
```

Example Input

```
{
  "Event1": {
    "description": "During the preparation of ginger tea, someone added ginger slices to the cup.",
    "time": "01:15 - 01:25",
    "details": {
      "action": "Someone is chopping ginger slices and adding them to the cup.",
      "scene": "In the classroom, a cup is placed on the table alongside freshly chopped ginger slices.",
      "dialogue": "This is ginger tea for the Director."
    },
    "characters": [
      {
        "name": "Zhaode Zhu",
        "appearance": {
          "time": "01:17 - 01:20",
          "action": "Add ginger slices to the cup.",
          "dialogue": "This is ginger tea for the Director."
        }
      }, ...
    ]
  }, ...
}
```

Example Output

```
{
  "Event1": {
    "time": "01:15 - 01:25",
    "description": "At the beginning of the video, Zhao Dezhu is seen in a classroom setting, preparing ginger tea. Holding pre-sliced pieces of ginger, he carefully places them into a cup with precise movements. As he finishes, he softly says, \"This is ginger tea for the Director.\" With the cup of ginger tea in hand, he then turns and walks toward the lectern, his actions clearly indicating that the tea was specifically prepared for the Director."
  }, ...
}
```

Figure 11. Prompt for Constructing the PC-DCoT.



Figure 12. **More qualitative cases in SeriesBench.** The performance of MLLM can be improved by using the PC-DCoT framework. We use **Green** to indicate correct and **Red** to indicate incorrect.



[00:30-00:40] Dressed in a white traditional costume, Xiao Xian strolls leisurely through a lush garden, lost in thought. Nearby, hidden behind bushes, the princess, adorned in a magnificent red gown, **discreetly watches him, concealing herself with leaves and a peacock-feathered fan.**

[01:08-01:18] Suddenly, **she conjures a blue magical circle in her hand**, clutching her chest in pain. She sways, collapses to the ground, and lies motionless, her face pale and lifeless. The scene exudes classical elegance while building tension and a sense of impending crisis.

Answer: The princess had been observing Xiao Xian beforehand and intentionally **used magic to faint** in order to attract his attention.

Script/Characters/Motivations



[00:25-00:35] Hong Ge and David are having a conversation in the office. Hong Ge appears somewhat agitated, making frequent gestures, while David remains calm. Shortly after, **Hong Ge receives a phone call and then turns to leave.** Following this, the scene shifts to Secretary Huang, who is shown sitting in a car, appearing relaxed, as if waiting for something.

[00:40-00:50] President Huang speaks formally with David, arms crossed confidently. David stands by, attentively awaiting her instructions. Mid-conversation, **Huang takes out her phone**, her expression turning serious as she handles an urgent matter.

Answer: A total of **two phones** appear in the video.

Visuals/Objects/Presence



[00:30-00:40] In the video, **a dormitory supervisor enters the dorm room** and stands next to the bed, appearing to inspect or organize the items. The dormitory looks tidy and orderly, with neatly made bedding on the bed and a desk and chair nearby.

[01:20-01:30] **A supervisor enters the room** and this time goes straight to the bed, lifting the blanket to inspect the items underneath.

[01:55-02:05] **The supervisor enters the room** with curly hair and a serious expression. He stands on the bed, holding a phone, seemingly checking the inventory of dormitory items.

Answer: The dormitory supervisor **entered the dormitory** a total of **three times**.

Script/Plot/Plot Development



[01:57-02:07] The video shows a scene of the director chasing Yingyan. The director, in a light-striped shirt and glasses, looks serious and focused as he grips the revolving door handle, trying to block Yingyan's escape. Yingyan, in a purple sweatshirt with messy hair, moves swiftly to evade him. **They engage in a tug-of-war at the revolving door, with the director persistently reaching for Yingyan, who struggles to break free.** The background features a "YUZHOU PLAZA" sign, suggesting the scene takes place in a commercial building. **The video blends tension with humor, making their interaction captivating.**

Answer: **The director chasing Yingyan** is the comedic highlight of the video.

Script/Plot/Emotional Dynamics

Figure 13. **Additional examples of open-ended responses using PC-DCoT.** MLLMs are capable of conducting more detailed analyses of events, relationships, and individuals, demonstrating a deeper understanding of narrative-driven series.

- Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*, 2023. 3
- [10] Monica Mak. Visual postproduction in participatory video-making processes. *Handbook of participatory video*, pages 194–207, 2012. 1
- [11] OpenAI. Gpt-4o mini: Advancing cost-efficient intelligence, 2024. Accessed: 2024-10-31. 11
- [12] OpenAI. Gpt-4o system card, 2024. Accessed: 2024-10-31. 11
- [13] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002. 7
- [14] InternVL Team. Internvl2: Better than the best—expanding performance boundaries of open-source multimodal models with the progressive scaling strategy, 2024. Accessed: 2024-10-31. 11
- [15] Towards Data Science. 7 ways to monitor large language model behavior, 2024. Accessed: 2024-11-22. 7
- [16] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024. 11
- [17] An Yang, Junshu Pan, Junyang Lin, Rui Men, Yichang Zhang, Jingren Zhou, and Chang Zhou. Chinese clip: Contrastive vision-language pretraining in chinese. *arXiv preprint arXiv:2211.01335*, 2022. 8
- [18] Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. Minicpm-v: A gpt-4v level mllm on your phone. *arXiv preprint arXiv:2408.01800*, 2024. 11
- [19] Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*, 2020. 7
- [20] Yuanhan Zhang, Jinming Wu, Wei Li, Bo Li, Zejun Ma, Ziwei Liu, and Chunyuan Li. Video instruction tuning with synthetic data, 2024. 11