Structured vs. Unstructured Inputs in LLMs: Evaluating the Semantic and Pragmatic Predictive Power in Abnormal Event Forecasting

Jou-An Chi
Graduate Institute of Linguistics
National Taiwan University
Taipei, Taiwan
r11142005@ntu.edu.tw

Shu-Kai Hsieh
Graduate Institute of Linguistics
National Taiwan University
Taipei, Taiwan
shukaihsieh@ntu.edu.tw

Abstract

Large Language Models (LLMs) are increasingly applied to temporally grounded reasoning tasks, yet the role of input representation remains unclear. This paper compares structured temporal inputs, represented as Temporal Knowledge Graphs (TKGs), with unstructured captions in two settings: forecasting future events and detecting anomalies in surveillance video descriptions. To enable direct comparison, we build a unified dataset by aligning anomaly labels from UCF-Crime with caption annotations from UCA. Experiments show that unstructured captions consistently yield slightly higher scores across both tasks, but the differences do not reach statistical significance. Their trade-offs, however, differ: captions provide richer semantic cues for generation, while TKGs reduce input length, suppress noise, and enhance interpretability. These findings suggest that action-centric corpora, such as surveillance or forensic narratives, naturally lend themselves to structured representations, which can provide temporal scaffolds for timeline reconstruction and more traceable reasoning. All code, data processing scripts, and experimental results are available at our GitHub repository.[1]

Keywords: Large Language Models (LLMs), Temporal Knowledge Graphs (TKGs), Forecasting, Anomaly Detection, Structured vs. Unstructured Input, Surveillance Video Understanding

1 Introduction

Large Language Models (LLMs) have demonstrated impressive performance across a wide spectrum of natural language processing tasks, ranging from open-domain question answering to temporal reasoning (Gruver et al., 2023; Jin et al., 2023a). Yet, when these models are applied to real-world scenarios where events unfold over time—such as surveillance video understanding, event forecasting, or anomaly detection—the choice of input representation becomes crucial. The way temporal context is presented to an LLM can significantly affect its ability to generate accurate predictions or make reliable judgments (Su et al., 2024a; Zhou and Yu, 2024).

Two common approaches to representing temporal context are unstructured text and structured knowledge representations. Raw textual descriptions, such as captions or transcripts, preserve rich semantic details and contextual cues, which may benefit generative tasks. However, they are also noisy and can introduce irrelevant information that distracts the model. In contrast, Temporal Knowledge Graphs (TKGs) encode events as structured quadruples (head entity, relation, tail entity, timestamp) (Gastinger et al., 2022; Trivedi et al., 2017b), thereby distilling interactions into a more compact and less noisy form. TKGs have been widely applied in temporal reasoning tasks such as forecasting and anomaly detection (Goel et al., 2020; Lee et al., 2023a; Jin et al., 2020). They facilitate knowledge management and temporal reasoning (Ji et al., 2021; Kejriwal, 2019), but may omit subtle semantic cues available in natural language. Despite the growing interest in both representations, there remains little systematic comparison of how structured and unstructured inputs affect LLM performance across different temporal tasks.

In this work, we investigate this gap by

[1] https://github.com/lowannann/StructVsUnstruct-LLM

Proceedings of the 37th Conference on Computational Linguistics and Speech Processing (ROCLING 2025), pages 237–248

asking: (1) Does structured temporal input provide advantages over unstructured input for forecasting tasks? (2) How does temporal context—whether structured or unstructured—impact anomaly detection tasks? We evaluate LLMs on two settings using a fine-grained surveillance video dataset that combines anomaly labels from UCF-Crime (Sultani et al., 2018) with caption annotations from UCA (Yuan et al., 2023).

Our contributions are threefold. First, we propose a comparative framework that evaluates structured and unstructured inputs under two complementary temporal reasoning tasks: forecasting and anomaly detection. Second, we provide empirical evidence that unstructured captions consistently perform slightly better across both tasks, though the differences are not statistically significant. This finding suggests that LLMs may not inherently favor one representation, but that the choice between structured and unstructured inputs should depend on task demands. Finally, our results carry practical implications for applying LLMs in temporally dynamic domains, highlighting how structured formats like TKGs can support contexts where reduced input cost, transparency, or traceability are essential.

2 Related Work

KGs, TKGs, and TKG Forecasting. Knowledge Graphs (KGs) organize entities and their relations into triples $\langle h, r, t \rangle$, offering a compact and interpretable representation that supports reasoning in applications such as semantic search and question answering (Kejriwal, 2019; Ji et al., 2021). However, many real-world scenarios are inherently temporal. To capture evolving dynamics, Temporal Knowledge Graphs (TKGs) extend this structure by associating each fact with a timestamp, forming quadruples $\langle h, r, t, \tau \rangle$ (Trivedi et al., 2017a; Leblay and Chekol, 2018; Goel et al., 2020; Jin et al., 2023b). This temporal extension enables modeling sequential dependencies and facilitates downstream tasks such as forecasting and anomaly detection in time-sensitive domains. By explicitly encoding temporal order, TKGs preserve event trajectories while reducing redundancy and noise compared to free-form text.

Research on TKG forecasting (TKGF) has traditionally relied on graph-based methods, which adapt knowledge graph embedding and graph neural network (GNN) architectures to temporal settings. Examples include RE-NET and recurrent RGCN variants that propagate historical states across timesteps (Jin et al., 2020; Chang et al., 2025), as well as symbolic approaches like TLogic and Temporal ILP that induce temporal rules (Liu et al., 2022; Xiong et al., 2024). While effective, these methods often require dataset-specific tuning and struggle in sparse or noisy contexts (Ma et al., 2023; Han et al., 2021). More recently, LLM-based approaches have reframed TKG forecasting as a language modeling problem, either by integrating graph embeddings into prompts (Zhang et al., 2024b; Wang et al., 2024; Zhang et al., 2024a) or by casting historical quadruples into textual sequences for in-context learning (Lee et al., 2023a; Liao et al., 2023; Luo et al., 2024). Remarkably, even general-purpose LLMs can perform competitively with specialized graph models, suggesting that LLMs capture not only semantic cues but also structural patterns in temporal data (Lee et al., 2023a).

LLMs in Forecasting and Anomaly Detection Forecasting is a fundamental temporal reasoning task that aims to predict future events or values from historical patterns. While traditionally addressed by statistical and deep learning models, recent work has demonstrated that LLMs provide strong generalization and flexible prompting mechanisms for this task (Jin et al., 2023a; Alnegheimish et al., 2024). Approaches include zero- or few-shot prompting, fine-tuning on domain-specific datasets, and direct application of foundation models. For example, Gruver et al. (2023) and Xue and Salim (2023) showed that GPT-family models and LLaMA variants can achieve competitive results on standard benchmarks in zero-shot settings, while fine-tuned BERT-based models improved regression accuracy on structured datasets (Xue et al., 2022). These studies highlight that LLMs can encode temporal dependencies through natural language interfaces, providing a flexible alternative to specialized time-series architectures.

Anomaly detection focuses on identifying deviations from expected temporal behavior and is increasingly framed as a diagnostic test of models' temporal reasoning ability (Su et al., 2024b; Zhou and Yu, 2024). LLMs have been applied here through three main strategies: using frozen encoders for log or sensor data, fine-tuning for binary anomaly classification, and prompt-based reasoning. For instance, Dang et al. (2021) fine-tuned BERT for detecting anomalies in KPI and Yahoo datasets, while Lee et al. (2023b) evaluated few-shot and zero-shot anomaly detection on system logs. Other prompt-based methods (Zhang et al., 2023; Huang et al., 2023) demonstrated that LLMs can capture subtle irregularities in noisy or weakly labeled data. Collectively, these findings suggest that LLMs not only generalize well across forecasting and anomaly detection but also provide a unified framework for handling diverse temporal reasoning tasks.

Input Representations and Prompting Strategies for LLM The representation of temporal information critically shapes how LLMs perform reasoning over time. Structured inputs—such as KG triples or graph embeddings—encode relations explicitly, providing precision and reducing ambiguity. Studies have shown that even when entity names are replaced with arbitrary IDs, LLMs can still perform forecasting by exploiting the structural patterns alone (Lee et al., 2023a). Similarly, prompts that present historical events as discrete triples allow the model to better recognize temporal dependencies than long descriptive texts, since the latter introduce noise and redundancy (Chang et al., 2024, 2025). In contrast, unstructured inputs—such as captions or free-form text—carry richer semantic information and contextual cues, but are noisier and harder for models to consistently parse.

Despite their noisiness, unstructured representations can complement structured data by capturing semantic or pragmatic information that graphs often omit. For example, textual descriptions may highlight causal links or implicit attributes useful for reasoning about events. Prior work has shown that combining structured triples with summarized or retrieved text improves model performance by balancing precision with semantic nuance (Chang et al., 2024). In temporal question answering, GenTKGQA (Gao et al., 2024) and M3TQA (Zha et al., 2024) illustrate how textual context and graph structure can be fused to cover each other's blind spots. These results suggest that structured and unstructured inputs are not mutually exclusive but offer complementary strengths: graphs provide clarity and temporal grounding, while text introduces richness and flexibility.

We regard temporal forecasting and anomaly detection as complementary settings for evaluating how LLMs process temporally structured input. Forecasting captures whether a model can extrapolate from observed sequences to anticipate plausible next events, while anomaly detection emphasizes the ability to recognize deviations that require attention to semantic coherence, pragmatic norms, and contextual irregularities. As Zhou and Yu (2024) notes, anomaly detection serves as a particularly diagnostic probe, since it goes beyond numerical accuracy and requires models to identify exceptions and contextual shifts rather than relying on surface-level continuation. Together, these two tasks provide complementary perspectives on temporal reasoning: one oriented toward projection, the other toward sensitivity to irregularities.

In this work, we leverage the UCF-Crime Annotation (UCA) dataset, whose human-written captions offer semantically and pragmatically grounded temporal descriptions of surveillance footage. By formulating both forecasting and anomaly detection on this data, we create a unified evaluation setting that allows us to examine how LLMs interpret structured inputs (TKGs) versus unstructured inputs (captions). This dual-task design is not aimed at comparing the tasks themselves, but at using them jointly to assess how input modality shapes models' ability to internalize temporal structures and reason about events.

3 Methods

3.1 Dataset

We employs the UCF-Crime dataset (Sultani et al., 2018) and its multimodal extension, the UCF-Crime Annotation (UCA) dataset (Yuan et al., 2023). UCF-Crime contains 1,900 long surveillance videos (over 128 hours) with ei-

ther normal activities or one of 13 predefined anomalous event types, such as Fighting, Robbery, Arson, Assault, and Burglary. In our setting, we define an anomaly as an event or activity within a video sequence that deviates significantly from expected normal patterns of behavior. Anomalies are inherently context-dependent, rare in occurrence, and in surveillance scenarios typically correspond to suspicious or potentially criminal actions (e.g., fighting, robbery, or arson). Following prior work on video anomaly detection, anomaly labels in our experiments are derived from benchmark annotations, where each anomalous frame is marked according to the presence of such irregular or threatening activities.

While UCF-Crime provides video-level binary anomaly labels and segment-level annotations for evaluation, it lacks natural language descriptions of visual content. To address this, the UCA dataset augments UCF-Crime with over 23,000 sentence-level captions (110 hours), each temporally aligned at 0.1-second resolution. These captions describe both normal and anomalous events in detail, offering semantically and pragmatically rich accounts of evolving scenes. The integration of UCF-Crime and UCA yields a unified data with anomaly labels, temporal spans, and human-written descriptions, enabling us to compare structured inputs (e.g., TKG quadruples) and unstructured inputs (caption sequences) for LLM-based forecasting and anomaly detection.

Table 1 provides illustrative examples from this unified dataset, showing how video segments are paired with human-written captions, their corresponding TKG representations, and anomaly labels. This format highlights the dual structured—unstructured nature of the data, which supports systematic evaluation of LLMs across different input modalities.

3.2 Models Used

We employed two LLMs, each serving a distinct role in the experimental pipeline for forecasting and anomaly detection tasks.

GPT-4o-Mini (via OpenAI API). GPT-4o-Mini was used exclusively for extracting TKG representations from natural language captions. The model was accessed through the OpenAI API[2] with LangChain[3]'s `LLMGraphTransformer()` module, using a temperature of 0.1 to ensure deterministic triple extraction. No fine-tuning or post-processing was applied beyond temporal alignment. A closed-source model was selected for this step due to its superior performance in zero-shot structural parsing and KG extraction (Huang et al., 2024; Carta et al., 2023), thereby ensuring high-quality and reliable TKG representations that minimize confounding errors in downstream evaluations.

Mistral-large-latest (via Open Source API). All downstream inference—forecasting and anomaly detection—was conducted with the open-source `mistral-large-latest`[4] . This model was chosen for two main reasons: (1) its open-source nature ensures reproducibility and transparency, which are essential for academic research; and (2) as an instruction-tuned model, it demonstrates strong reasoning and generation capabilities across diverse NLP tasks. To maintain consistency, all runs used identical inference parameters: temperature = 0.1, top-p = 1.0, and maximum input length = 128. This setup guarantees a controlled comparison between structured (TKG-based) and unstructured (caption-based) inputs.

By separating the TKG extraction phase from the main evaluation model, we ensure that observed differences between input modalities stem from the LLM's reasoning capacity rather than inconsistencies in structural encoding quality.

3.3 Experiment 1: Forecasting

Objective. The forecasting experiment evaluates whether LLMs can generate semantically plausible next-event descriptions based on prior temporal context. Instead of predicting new triples, the task is framed as forecasting the natural language caption of a future video frame given preceding input in two forms: (1) structured TKG quadruples and (2) unstructured captions. The key goal is

[2]OpenAI API: https://openai.com/index/openai-api/

[3]LangChain: https://python.langchain.com/docs/introduction/

[4]Mistral AI: https://docs.mistral.ai/getting-started/models/models_overview/

Video Type	Timestamp	Caption (Text and TKG Format)	Anomalous
Arson	81.3–106	**Text:** The man walked down and tried to light a piece of paper but failed to light it. **TKG:** {[Man, WALKED_DOWN, Paper], [Man, TRIED_TO_LIGHT, Paper], [Man, FAILED_TO_LIGHT, Paper]}	False
	115.8–121.2	**Text:** The man returned to the Christmas tree and continued to light it and successfully lit it. **TKG:** {[Man, RETURNED_TO, Christmas Tree], [Man, CONTINUED_TO_LIGHT, Christmas Tree], [Man, SUCCESSFULLY_LIT, Christmas Tree]}	True
Burglary	254.4–255.8	**Text:** Another person opened the trunk, and there were several men in white hiding in the trunk. **TKG:** {[Another Person, HIDING_IN, Men In White]}	False
	256.1–350.4	**Text:** A total of five people gathered around the door and cooperated to pry it open. **TKG:** {[People, GATHERED_AROUND, Door], [People, COOPERATED_TO_PRY_OPEN, Door]}	True
Explosion	0.0–9.0	**Text:** Many cars were parked on the roadside and many people walking on the roadside. **TKG:** {[Cars, PARKED_ON, Roadside], [People, WALKING_ON, Roadside]}	False
	9.0–21.3	**Text:** An explosion occurred in a building and produced smoke, and the glass of the nearby building was shaken. **TKG:** {[Explosion, OCCURRED_IN, Building], [Explosion, PRODUCED, Smoke], [Building, SHAKEN, Glass]}	True

Table 1: Examples of aligned captions, their corresponding TKG quadruples, and anomaly labels across video types.

to assess semantic coherence and contextual appropriateness of the generated output. An overview of the pipeline is shown in Figure 1.

Input Settings. Two input conditions were tested:

- Structured (TKG → Text): Captions were converted into subject—relation—object triples with aligned timestamps. These quadruples were verbalized into structured prompt templates.
- Unstructured (Text → Text): Raw or lightly summarized captions were concatenated to form free-text temporal context, which was directly inserted into the prompt.

Prompt Design. Prompts were designed to ensure parity across conditions, differing only in input format. In both cases, the LLM was instructed to predict the most likely action immediately preceding an anomaly and to output exactly one complete sentence. Example prompt templates are shown in Figure 2 and Figure 3.

Prompted Generation. Formally, the prediction is modeled as:

$$\hat{y}_{text} = \Phi_{LLM}(P_{\mathcal{I}}), \quad \mathcal{I} \in \{TKG \rightarrow Text,\ Text \rightarrow Text\} \tag{1}$$

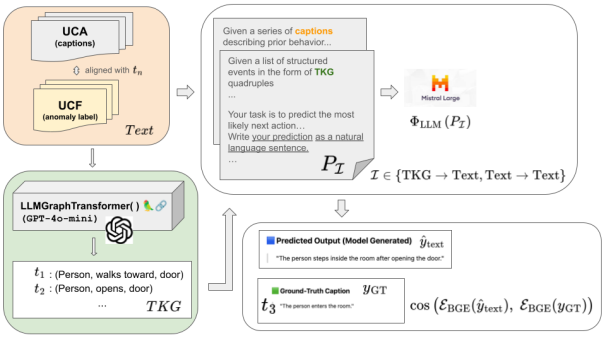

Figure 1: Pipeline of the Experiment 1: forecasting task. The model receives temporally ordered input (either structured TKGs or unstructured captions) and generates a next-frame description. The generated output is then compared against ground-truth captions to evaluate semantic alignment.

where Φ_{LLM} denotes the inference model and $P_{\mathcal{I}}$ the prompt constructed from temporal context.

Metrics Predicted sentences $\hat{y}_{text}$ were compared against human-annotated ground-truth captions y_{GT} using semantic similarity. Both sentences were encoded with the BAAI General Embedding (BGE) model, and cosine similarity was computed:

$$Similarity = \cos(\mathcal{E}_{BGE}(\hat{y}_{text}),\ \mathcal{E}_{BGE}(y_{GT})) \tag{2}$$

Cosine similarity captures paraphrastic overlap without requiring exact lexical matches, making it well-suited for evaluating free-text generation. Segment-level scores were averaged across the evaluation set to yield the final similarity metric.

3.4 Experiment 2: Anomaly Detection

Objective. The anomaly detection experiment evaluates how well LLMs identify abnormal events in surveillance video descriptions under different temporal input conditions. Given a sequence of frame-level captions, the model must judge whether the current frame is anomalous. Anomalies are defined as events that deviate significantly from expected behavioral patterns and typically correspond to suspicious or criminal actions (e.g., fighting, robbery, arson). This task probes the model' s ability to reason over event coherence and detect pragmatic inconsistencies. An overview of the pipeline is shown in Figure 4.

```
[Goal]: You are given a list of structured events in the
form of temporal knowledge graph (TKG) quadruples:
(subject, relation, object, timestamp). These represent
a subject's past actions over time.

Your task is to predict the most likely next action that
the subject will perform **immediately before an
abnormal event occurs**. Write your prediction as a
natural language sentence.

[Input - TKG History Before Anomaly]:

T1: {[Man, WALKED_DOWN, Paper], [Man,
TRIED_TO_LIGHT, Paper], [Man, FAILED_TO_LIGHT,
Paper]}

T2: {[Man, RETURNED_TO, Christmas Tree], [Man,
CONTINUED_TO_LIGHT, Christmas Tree], [Man,
SUCCESSFULLY_LIT, Christmas Tree]}

[Constraint]:
- Predict exactly **one sentence** that describes the
next likely action.
- Your output should be **one complete sentence**.

[Output - Predicted Sentence]:
```

Figure 2: Structured Input Prompt (TKG → Text) used in the forecasting task. The model is provided with a sequence of TKG quadruples representing past events and is asked to predict, in one complete sentence, the most likely next action before an anomalous event.

Prompt Design. Following the training-free strategy of Zanella et al. (2024), we prompt the LLM to assign a scalar anomaly score $a \in [0, 1]$ for each frame. Examples of each prompt are provided in Figures 5—6. The prompt is composed of three parts:

- $\mathcal{P}_S$: a system instruction framing the task as risk assessment on a 0—1 scale;
- $\mathcal{P}_F$: an output-format instruction requiring one number from a discrete set of 11 values (0.0—1.0 in steps of 0.1);
- $\mathcal{P}_C$: the temporal context, either unsummarized captions, LLM-summarized captions, or TKG quadruples:

```
[Goal]: The following is a series of natural language
captions describing the subject's behavior leading up to
an abnormal event.

Your task is to predict the most likely next action that
the subject will take right before the anomaly occurs.
The prediction should be in natural language.

[Input - Captions Before Anomaly]:

T1: The man walked down and tried to light a piece of
paper but failed to light it.

T2: The man returned to the Christmas tree
and continued to light it and successfully lit it.

[Constraint]:
- Predict exactly **one sentence** that describes the
subject's next likely action.
- Your output should be **one complete sentence**.

[Output - Predicted Caption]:
```

Figure 3: Unstructured Input Prompt (Text → Text) used in the forecasting task. The model is given a sequence of natural language captions describing prior events and is instructed to generate one complete sentence predicting the subject's next likely action before an anomaly.

$$\mathcal{C}_{temporal} \in \{\mathcal{C}_{unsummarized}, \mathcal{C}_{summarized}, \mathcal{C}_{TKG}\}. \tag{3}$$

The final prompt concatenates these components, and the LLM outputs a single anomaly score:

$$\hat{a} = \Phi_{LLM}(\mathcal{P}_S \circ \mathcal{P}_F \circ \mathcal{P}_C). \tag{4}$$

Metrics. We adopt AUC-ROC as the primary evaluation metric. Each prediction $\hat{a}$ is compared against the binary ground-truth label $a_{GT} \in 0, 1$ from UCF-Crime. AUC measures the model's ranking ability across all thresholds:

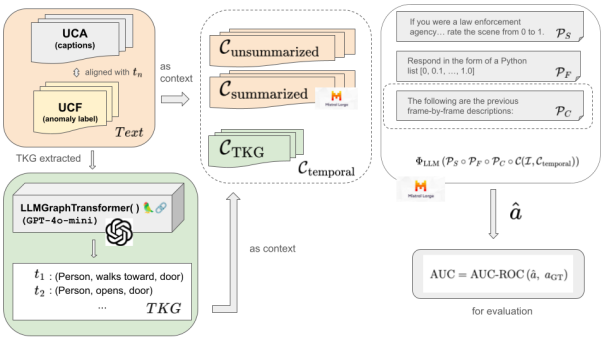

Figure 4: Pipeline of the Experiment 2: anomaly detection task. The model receives prior context in one of three forms—raw captions, summarized text, or structured TKGs—and predicts whether the current frame is anomalous. The prediction is compared against the ground-truth anomaly label for evaluation.

$$AUC = AUC - ROC(\hat{a}, a_{GT}). \tag{5}$$

Unlike accuracy, AUC is threshold-independent and reflects whether anomalies are consistently ranked above normal events. This makes it well-suited for surveillance, where operational definitions of abnormality vary. In our setting, AUC provides a robust criterion to compare how structured versus unstructured temporal inputs enable LLMs to detect deviations from normative event sequences.

4 Discussion

4.1 Research Questions

RQ1: Does structured temporal input provide advantages over unstructured input for temporal forecasting tasks? As shown in Table 2 and Table 3, the unstructured input condition yielded a slightly higher mean similarity score (0.5978) than the structured TKG input (0.5718). However, this difference was not statistically significant, as indicated by both the paired t-test ($t = 1.7259, p = 0.0978$) and the non-parametric Wilcoxon signed-rank test ($W = 90.0, p = 0.0894$). These findings suggest that unstructured input consistently produces marginally higher semantic alignment on average, though the difference does not reach statistical significance.

If you were a law enforcement agency, rate the scene on a scale from 0 to 1, where 0 = standard (normal) and 1 = suspicious activity (abnormal). Use the following semantic grading as guidance (examples are illustrative only; your output must still be numeric from the list below):

- 0.80–1.00 → Almost Certain / Very Likely
→ The scene is almost certainly abnormal. Example: Abuse, Arrest, Arson, Assault, Road Accident, Burglary, Explosion, Fighting, Robbery, Shooting, Stealing, Shoplifting, and Vandalism.

- 0.60–0.80 → High Suspicion
→ Clear signs of abnormality, though alternative explanations may exist. Example: pushing, running aggressively, heated argument.

- 0.40–0.60 → Uncertain / Ambiguous
→ Difficult to judge; may require human review or depend on domain context.

- 0.20–0.40 → Low Suspicion
→ Scene is mostly normal, but with minor suspicious cues. Example: unusual crowd gathering, subtle suspicious movements.

- 0.00–0.20 → Normal / Unlikely
→ Almost certainly routine daily activity.

Figure 5: System prompt ($\mathcal{P}_S$) used in the anomaly detection task. This instruction frames the model's role as a law enforcement agent and asks it to assess whether the described scene is normal or suspicious on a scale from 0 (normal) to 1 (highly anomalous).

Respond STRICTLY as a Python list containing ONE number chosen from: [0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0] No text, no labels, no extra spaces or characters. The response MUST begin with '[' and end with ']'.

You are given prior frame-by-frame textual descriptions as CONTEXT (no anomaly scores are provided): {context}

Use this context ONLY to understand the flow of events (e.g., who is present, what objects are involved, how actions transition over time).

⚠ IMPORTANT: The anomaly score of the CURRENT frame must be determined independently. Do NOT inherit or carry over abnormality from previous frames. Context is background information only; the final rating must reflect what is explicitly described in the CURRENT frame. Current frame description:

Figure 6: Prompt components for the anomaly detection task. Top: output-format prompt ($\mathcal{P}_F$), which constrains the model to return exactly one anomaly score as a Python list containing a single value between 0.0 and 1.0. Bottom: context prompt ($\mathcal{P}_C$), which provides frame-by-frame descriptions as temporal context. The context is used only to interpret event flow, while the anomaly score must be determined independently for the current frame.

A closer inspection of forecasting outputs reveals systematic error patterns that help explain this small but non-significant gap. With TKG-based inputs, predictions often stalled at preparatory actions rather than advancing toward anomalous outcomes (e.g., anticipating ignition attempts but not the actual arson). Highly specific or low-probability events —such as an arsonist accidentally catching fire—were rarely captured, reflecting the difficulty of forecasting unexpected developments from sparse cues. The model also frequently lacked narrative progression, anchoring on earlier triplets and producing semantically plausible but stagnant outputs. Finally, forecasting performance varied by category: gradual, visually grounded events (e.g., arson, shoplifting) were more predictable than abrupt or ambiguous ones (e.g., explosions, accidents, shootings), highlighting the dependence of structured inputs on contextual richness.

These results carry important implications for the utility of structured input. TKGs offer a consistent and formal representation that abstracts away surface-level linguistic noise and encourages the model to reason based on event structure and temporal progression. This consistency may be beneficial in downstream tasks that require symbolic manipulation or multimodal alignment. By contrast, raw captions naturally carry richer lexical and syntactic cues, which directly benefit tasks emphasizing surface-level semantic similarity. While TKGs did not surpass unstructured captions in raw semantic similarity in this experiment, their representational strengths suggest potential advantages in more complex, reasoning-intensive applications.

Input Type	Cosine Similarity
Unstructured (Text → Text)	0.5978
Structured (TKG → Text)	0.5718

Table 2: Mean cosine similarity scores for structured and unstructured input conditions.

Test	Stat.	p-value
Paired t-test	$t = 1.7259$	0.0978
Wilcoxon (SR)	$W = 90.0$	0.0894

Table 3: Statistical test results comparing structured and unstructured input conditions.

RQ2: How does temporal context—whether structured or unstructured—impact LLM performance in anomaly detection tasks? We examine how temporal context—structured vs. unstructured—affects LLM anomaly detection. As shown in Fig. 7, summarized text attains AUC = 0.7817, raw text 0.7766, and TKG 0.7673. Pairwise DeLong tests (Table 4) indicate no significant differences among conditions: summarized–TKG ΔAUC = +0.014 ($p = 0.345$, 95

Qualitative error analysis reveals a few systematic behaviors. The model showed oversensitivity to ambiguous behaviors, classifying vague or cautious actions (e.g., pacing or looking around) as anomalies. Another bias appeared in action-triggered cases: attempts such as "trying to light"were flagged as anomalous even when unsuccessful.

A plausible mechanism is that TKG provides a low-noise, reference-only context. By encoding ⟨subject, relation, object, time⟩, it preserves the action backbone (who did what, when) while filtering lexical and pragmatic clutter that can nudge the model toward spurious cues. Unstructured text—especially summaries—retains fine-grained signals (e.g., negation, intensity, scene qualifiers) that occasionally help, which could explain the small numerical edge, though the average advantage remains modest. Overall, in action-centric surveillance scenes, compact structured context can achieve similar statistical performance to longer textual context while reducing token cost, offering a cost-efficient alternative when latency or context length matters.

At the same time, both tasks reveal common limitations of current LLMs for temporal reasoning: difficulty projecting narrative progression, a tendency to conflate intent with actual threat, and challenges in maintaining calibrated anomaly judgments. These findings suggest that while LLMs can leverage both structured and unstructured inputs, they still require mechanisms that better capture causal progression, distinguish ambiguous intent from concrete outcomes, and handle noisy labels.

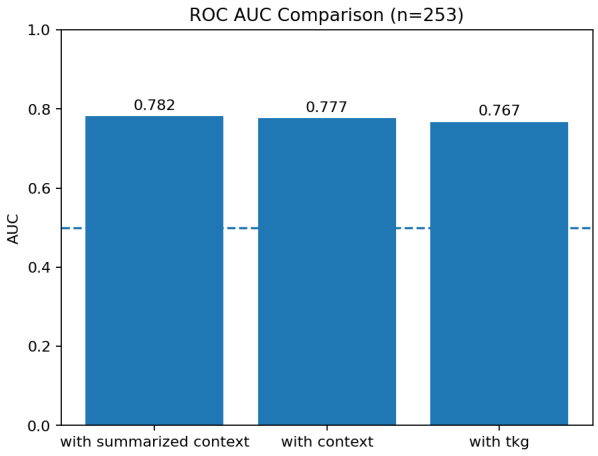

Figure 7

5 Conclusion

This study compared structured (TKG) and unstructured (caption) temporal inputs in abnormal event forecasting and anomaly detection with LLMs. Our results show that unstructured captions consistently yield slightly higher scores in both tasks, but these differences do not reach statistical significance. This finding highlights that when data are inherently action-centric—as in UCF-Crime and UCA, where human activities are described in subject—verb—object form—structured representations like TKGs provide a conceptually natural scaffold. Even when empirical gains over unstructured inputs are modest, TKGs reduce token length, enhance interpretability, and align closely with the relational structure of the data. These advantages carry practical implications for domains such as surveillance, legal, and forensic analysis, where transforming fragmented narratives into structured graphs can facilitate timeline reconstruction, highlight contradictions, and support traceable reasoning over events.

6 Limitations

Our study has several limitations. Results are based on a single open-source model (Mistral-large-latest), and may differ with other architectures or fine-tuning. Token length was not systematically explored, leaving open how

Comparison	AUC(A)	AUC(B)	ΔAUC	SE	z	p	95% CI
`raw_txt` − `tkg`	0.7673	0.7766	0.0093	0.0188	0.494	0.621	[−0.0275, 0.0461]
`sum_txt` − `tkg`	0.7673	0.7817	0.0143	0.0152	0.943	0.345	[−0.0155, 0.0441]
`sum_txt` − `raw_txt`	0.7766	0.7817	0.0051	0.0201	0.252	0.801	[−0.0344, 0.0445]

Table 4: DeLong tests for pairwise ROC AUC differences; ΔAUC = AUC(B) − AUC(A). None of the differences are statistically significant at $\alpha = 0.05$.

TKG efficiency scales under extreme long-context settings. The dataset size may also limit statistical power: unstructured inputs consistently scored slightly higher, yet differences were not significant. Finally, UCF-Crime and UCA anomaly labels may contain temporal misalignments or noise. Future work should test diverse models, larger and more varied datasets, long-context benchmarks, and improved annotations.

References

Sarah Alnegheimish, Linh Nguyen, Laure Berti-Equille, and Kalyan Veeramachaneni. 2024. Large language models can be zero-shot anomaly detectors for time series? *arXiv preprint arXiv:2405.14755*.

Salvatore Carta, Alessandro Giuliani, Leonardo Piano, Alessandro Sebastian Podda, Livio Pompianu, and Sandro Gabriele Tiddia. 2023. Iterative zero-shot llm prompting for knowledge graph construction. *arXiv preprint arXiv:2307.01128*.

He Chang, Jie Wu, Zhulin Tao, Yunshan Ma, Xianglin Huang, and Tat-Seng Chua. 2025. Integrate temporal graph learning into llm-based temporal knowledge graph model.

He Chang, Chenchen Ye, Zhulin Tao, Jie Wu, Zhengmao Yang, Yunshan Ma, Xianglin Huang, and Tat-Seng Chua. 2024. A comprehensive evaluation of large language models on temporal event forecasting. *arXiv preprint arXiv:2407.11638*.

Weixia Dang, Biyu Zhou, Lingwei Wei, Weigang Zhang, Ziang Yang, and Songlin Hu. 2021. Tsbert: Time series anomaly detection via pre-training model bert. In *Computational Science–ICCS 2021: 21st International Conference, Krakow, Poland, June 16–18, 2021, Proceedings, Part II 21*, pages 209–223. Springer.

Yifu Gao, Linbo Qiao, Zhigang Kan, Zhihua Wen, Yongquan He, and Dongsheng Li. 2024. Two-stage generative question answering on temporal knowledge graph using large language models. *arXiv preprint arXiv:2402.16568*.

Julia Gastinger, Timo Sztyler, Lokesh Sharma, and Anett Schuelke. 2022. On the evaluation of methods for temporal knowledge graph forecasting. In *NeurIPS 2022 Temporal Graph Learning Workshop*.

Rishab Goel, Seyed Mehran Kazemi, Marcus Brubaker, and Pascal Poupart. 2020. Diachronic embedding for temporal knowledge graph completion. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 3988–3995.

Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. 2023. Large language models are zero-shot time series forecasters. *Advances in Neural Information Processing Systems*, 36:19622–19635.

Xu Han, Zhiyuan Liu, and Maosong Sun. 2021. Robustness of temporal knowledge graph forecasting models to sparsity. In *Findings of EMNLP*.

Haoyu Huang, Chong Chen, Conghui He, Yang Li, Jiawei Jiang, and Wentao Zhang. 2024. Can llms be good graph judger for knowledge graph construction? *arXiv preprint arXiv:2411.17388*.

Shaohan Huang, Yi Liu, Carol Fung, He Wang, Hailong Yang, and Zhongzhi Luan. 2023. Improving log-based anomaly detection by pre-training hierarchical transformers. *IEEE Transactions on Computers*, 72(9):2656–2667.

Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip S Yu. 2021. A survey on knowledge graphs: Representation, acquisition and applications. *IEEE Transactions on Neural Networks and Learning Systems*.

Di Jin, Ivana Balazevic, Carl Allen, and Timothy M. Hospedales. 2020. Re-net: Reasoning over knowledge graph paths for temporal knowledge base completion. In *AAAI*.

Ming Jin, Shiyu Wang, Lintao Ma, Zhixuan Chu, James Y Zhang, Xiaoming Shi, Pin-Yu Chen, Yuxuan Liang, Yuan-Fang Li, Shirui Pan, et al. 2023a. Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*.

Ming Jin, Qingsong Wen, Yuxuan Liang, Chaoli Zhang, Siqiao Xue, Xue Wang, James Zhang, Yi Wang, Haifeng Chen, Xiaoli Li, Shirui Pan, Vincent S. Tseng, Yu Zheng, Lei Chen, and Hui Xiong. 2023b. Large models for time series and spatio-temporal data: A survey and outlook.

Mayank Kejriwal. 2019. *Domain-specific knowledge graph construction*. Springer.

Julien Leblay and Melisachew Wudage Chekol. 2018. Deriving validity time in knowledge graph. In *Companion proceedings of the the web conference 2018*, pages 1771–1776.

Dong-Ho Lee, Kian Ahrabian, Woojeong Jin, Fred Morstatter, and Jay Pujara. 2023a. Temporal knowledge graph forecasting without knowledge using in-context learning. *arXiv preprint arXiv:2305.10613*.

Yukyung Lee, Jina Kim, and Pilsung Kang. 2023b. Lanobert: System log anomaly detection based on bert masked language model. *Applied Soft Computing*, 146:110689.

Ruotong Liao, Xu Jia, Yangzhe Li, Yunpu Ma, and Volker Tresp. 2023. Gentkg: Generative forecasting on temporal knowledge graph with large language models. *arXiv preprint arXiv:2310.07793*.

Yushan Liu, Yunpu Ma, Marcel Hildebrandt, Mitchell Joblin, and Volker Tresp. 2022. Tlogic: Temporal logical rules for explainable link forecasting on temporal knowledge graphs. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 4120–4127.

Ruilin Luo, Tianle Gu, Haoling Li, Junzhe Li, Zicheng Lin, Jiayi Li, and Yujiu Yang. 2024. Chain of history: Learning and forecasting with llms for temporal knowledge graph completion. *arXiv preprint arXiv:2401.06072*.

Yunshan Ma, Chenchen Ye, Zijian Wu, Xiang Wang, Yixin Cao, and Tat-Seng Chua. 2023. Context-aware event forecasting via graph disentanglement. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1643–1652.

Jing Su, Chufeng Jiang, Xin Jin, Yuxin Qiao, Tingsong Xiao, Hongda Ma, Rong Wei, Zhi Jing, Jiajun Xu, and Junhong Lin. 2024a. Large language models for forecasting and anomaly detection: A systematic literature review. *arXiv preprint arXiv:2402.10350*.

Jing Su, Chufeng Jiang, Xin Jin, Yuxin Qiao, Tingsong Xiao, Hongda Ma, Rong Wei, Zhi Jing, Jiajun Xu, and Junhong Lin. 2024b. Large language models for forecasting and anomaly detection: A systematic literature review.

Waqas Sultani, Chen Chen, and Mubarak Shah. 2018. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6479–6488.

Rakshit Trivedi, Hanjun Dai, Yichen Wang, and Le Song. 2017a. Know-evolve: Deep temporal reasoning for dynamic knowledge graphs.

Rakshit Trivedi, Manaal Faruqui, Yann Dauphin, and Dani Yogatama. 2017b. Know-evolve: Deep temporal reasoning for dynamic knowledge graphs. In *ICML*.

Duo Wang, Yuan Zuo, Fengzhi Li, and Junjie Wu. 2024. Llms as zero-shot graph learners: Alignment of gnn representations with llm token embeddings. *Advances in Neural Information Processing Systems*, 37:5950–5973.

Siheng Xiong, Yuan Yang, Faramarz Fekri, and James Clayton Kerce. 2024. Tilp: Differentiable learning of temporal logical rules on knowledge graphs. *arXiv preprint arXiv:2402.12309*.

Hao Xue and Flora D Salim. 2023. Promptcast: A new prompt-based learning paradigm for time series forecasting. *IEEE Transactions on Knowledge and Data Engineering*, 36(11):6851–6864.

Hao Xue, Bhanu Prakash Voutharoja, and Flora D Salim. 2022. Leveraging language foundation models for human mobility forecasting. In *Proceedings of the 30th International Conference on Advances in Geographic Information Systems*, pages 1–9.

Tongtong Yuan, Xuange Zhang, Kun Liu, Bo Liu, Chen Chen, Jian Jin, and Zhenzhen Jiao. 2023. Towards surveillance video-and-language understanding: New dataset, baselines, and challenges.

Luca Zanella, Willi Menapace, Massimiliano Mancini, Yiming Wang, and Elisa Ricci. 2024. Harnessing large language models for training-free video anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18527–18536.

Zhiyuan Zha, Pengnian Qi, Xigang Bao, Mengyuan Tian, and Biao Qin. 2024. M 3 tqa: Multi-view, multi-hop and multi-stage reasoning for temporal question answering. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 10086–10090. IEEE.

Mengmei Zhang, Mingwei Sun, Peng Wang, Shen Fan, Yanhu Mo, Xiaoxiao Xu, Hong Liu, Cheng Yang, and Chuan Shi. 2024a. Graphtranslator: Aligning graph model to large language model for open-ended tasks. In *Proceedings of the ACM Web Conference 2024*, pages 1003–1014.

Ting Zhang, Xin Huang, Wen Zhao, Shaohuang Bian, and Peng Du. 2023. Logprompt: A log-based anomaly detection framework using prompts. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Yichi Zhang, Zhuo Chen, Lingbing Guo, Yajing Xu, Wen Zhang, and Huajun Chen. 2024b. Making

large language models perform better in knowledge graph completion. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 233–242.

Zihao Zhou and Rose Yu. 2024. Can llms understand time series anomalies? *arXiv preprint arXiv:2410.05440*.