Risk-Controlled Event-Driven Cascading Updates for Knowledge Graph Consistency Restoration

Bo Ni[1], **Qinwen Ge**[1], **Haowei Fu**[1], **Ryan A. Rossi**[2], **Xiaorui Liu**[3], **Jiejun Xu**[4], **Tyler Derr**[1]

[1]Vanderbilt University, [2]Adobe Research, [3]North Carolina State University, [4]HRL Laboratories
{bo.ni, qinwen.ge, haowei.fu, tyler.derr}@vanderbilt.edu, ryrossi@adobe.com, xliu96@ncsu.edu, jxu@hrl.com

Abstract

Knowledge Graphs (KGs) provide structured and interpretable representations of real-world entities and relations. While dynamic KGs attempt to capture real-time changes, they typically treat updates as independent facts. This overlooks a critical challenge: a factual, localized update can contradict and invalidate previously correct knowledge, requiring revisions beyond the localized update to maintain KG consistency. Many of these inconsistencies arise from events whose effects propagate through relational dependencies, necessitating coordinated multi-hop reasoning rather than isolated changes. To address this, we introduce a model-agnostic framework for cascading KG update identification that leverages conformal prediction to provide reliable uncertainty guarantees over the cascade as a whole, accounting for dependencies among multi-hop update candidates. Building on this foundation, we further develop a graph-based KG update scoring framework that integrates large language models (LLMs) to enrich event representations with world knowledge. Experiments on two newly constructed real-world datasets, designed to reflect scenarios where events necessitate coordinated multi-hop updates, demonstrate that our framework establishes a strong baseline while offering calibrated confidence estimates. Our code and datasets are available at https://github.com/Arstanley/cascadekg.

1 Introduction

Knowledge Graphs (KGs) are widely used across many applications, e.g., question answering, recommendation, and information retrieval (Ji et al., 2022; Wang et al., 2024; Ni et al., 2025b). Most existing KG systems typically assume that the underlying graph is factual, consistent, and reliable (Ni et al., 2025a). Yet real-world knowledge is inherently dynamic (Cai et al., 2024): new facts emerges and may lead to changes in previously valid facts. Consequently, blindly incorporating updates can leave event-invalidated facts uncorrected, and resolving one inconsistency may expose others via dependent relations, affecting downstream inferences. Ensuring the reliability and consistency of knowledge over time has therefore become an increasingly urgent challenge.

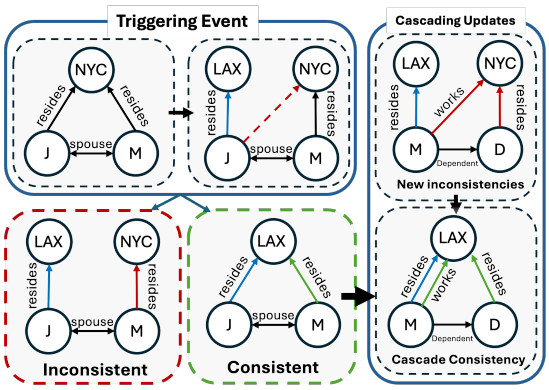


Figure 1: An illustration of event-driven cascading KG updates for consistency restoration. A triggering event, such as John moving to Los Angeles, invalidates an existing relation, and revising it reveals additional affected relations that require cascading updates.

In practice, the effects of real-world events rarely remain confined to a single fact. As illustrated in Figure 1, events such as a change in residence can invalidate existing relations and require updates to restore local consistency. Importantly, revising one event-invalidated relation may expose additional dependent relations that also require revision. This dependency-driven behavior makes traditional KG completion and temporal forecasting models insufficient, as they assume missing but internally consistent facts. Simply adding new triplets without accounting for such dependencies can therefore leave the knowledge graph inconsistent, resulting in an inaccurate representation of the evolving world.

To accommodate the new information, prior work has primarily focused on synchronizing knowledge graphs with structured sources such as encyclopedias and external knowledge bases (Morsey et al., 2012; Liang et al., 2017).

Findings of the Association for Computational Linguistics: ACL 2026, pages 42534–42548

For instance, Liang et al. (2017) relies on predefined rules and extractors tailored for Wikipedia to update the KG. While effective for integrating structured updates, such methods do not extend naturally to unstructured or event-driven inputs due to their reliance on rigid pipelines. More importantly, they lack mechanisms to assess whether updates restore consistency beyond the immediate change, making them ill-suited for scenarios where events require coordinated multi-hop revisions.

More recently, Tang et al. (2019) investigated knowledge graph updates from news snippets, leveraging a text-based attention mechanism to model both explicit and implicit updates from unstructured text inputs. Despite these advances, important limitations remain. First, the reliance on textual descriptions limits applicability in domains where rich textual coverage is unavailable. Second, the approach focuses primarily on local, one-hop changes and does not adequately capture multi-hop cascading updates induced by events through structural dependencies in the knowledge graph.

In this work, we propose a model-agnostic framework, CASCADEKG, to systematically address event-driven cascading updates in knowledge graphs. Our framework leverages the Learn-Then-Test (LTT) (Angelopoulos et al., 2022) calibration to provide statistical guarantees over multi-hop edge updates within the needed update cascade, ensuring that the propagated changes remain within a calibrated uncertainty bound. Rather than committing to individual updates in isolation, LTT supports uncertainty-aware identification of coordinated updates required to restore consistency. Additionally, ensuring consistency in real-world updates often requires external world knowledge to interpret event semantics and relational dependencies that are not explicitly encoded in the graph. Existing methods that rely solely on structural patterns therefore fall short in capturing these nuanced contextual cues. To this end, we develop a **T**ext-**au**gmented KG Update **S**coring Framework, TAUS, that integrates large language models (LLMs) to enrich event representations with contextual and world knowledge, enabling more faithful consistency restoration under real-world conditions.

To evaluate the proposed framework, we construct two new datasets derived from the Integrated Crisis Early Warning System (ICEWS) (Boschee et al., 2015) and Wiki knowledge graphs (Suchanek et al., 2007), designed to reflect scenarios where events necessitate coordinated multi-hop updates. Extensive experiments demonstrate that our approach consistently outperforms baselines, including rule-based methods, embedding-only models, and text-dependent GNNs, across multiple evaluation metrics. These results highlight the effectiveness of uncertainty-aware, event-conditioned graph reasoning for restoring consistency in knowledge graphs as real-world dynamics unfold.

Our key contributions can be summarized as:

- We identify and address the problem of event-driven cascading knowledge graph updates, where trigger events invalidate existing facts and require coordinated multi-hop revisions to restore consistency.
- We propose a model-agnostic risk control framework for multi-hop update identification that provides statistical guarantees over cascades of update candidates.
- We develop an event-conditioned, text-augmented graph reasoning framework for KG update scoring that captures relational effects triggered by events.
- We construct two new benchmark datasets from ICEWS and Wiki, designed to reflect event-driven cascading update scenarios, and conduct extensive experiments on these datasets to demonstrate the effectiveness of the proposed framework.

2 Problem Formulation

Before presenting our framework, we formally define the problem of event-driven cascading knowledge graph updates. We begin by introducing the necessary notations and the concept of consistency-aware updates, then present our proposed framework called CASCADEKG in Section 3.

2.1 Preliminaries

Let $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$ denote a knowledge graph, where $\mathcal{E}$ and $\mathcal{R}$ are sets of entities and relations, respectively, and $\mathcal{T} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ is the set of factual triplets. We define a *trigger event* e as a single KG update that necessitates a modification to $\mathcal{G}$ to reflect the current state of the world.

2.2 Cascading KG Updates

Standard KG completion treats updates as isolated predictions (Chen et al., 2020). However, in dynamic environments, a single event often initiates a chain reaction of logical consequences. To capture this, we model the update process as an iterative cascade as follows:

Definition 1 (Event-Driven Cascading KG Update)**.** *Given an initial knowledge graph $\mathcal{G}_0$ and a trigger event e, let $\mathcal{G}_e \subseteq \mathcal{G}_0$ be an initial event-centered subgraph within L-hop of nodes in e. The cascading update process is defined as a sequence of state transitions $\mathcal{G}_0 \xrightarrow{\Delta_0} \mathcal{G}_1 \xrightarrow{\Delta_1} \dots \xrightarrow{\Delta_K} \mathcal{G}_K$, where:*

1. *Δ_0 represents the explicit update(s) directly specified by e.*
2. *For each subsequent step $k > 0$, the update Δ_k is the set of implicit cascading updates triggered by the changes in Δ_{k-1}. Crucially, Δ_k is derived by reasoning over the evolved local context $\mathcal{G}_e \cup \left(\bigcup_{i=0}^{k-1} \Delta_i\right)$. This formulation ensures that the prediction of the k-th stage integrates the cumulative relational dependencies introduced in all preceding stages.*

The objective is to identify the cumulative update set $\Delta_{total} = \bigcup_{k=0}^{K} \Delta_k$ such that the final graph $\mathcal{G}_K$ achieves global consistency and faithfulness to the real-world dynamics initiated by e.

Essentially, the *output* of step k serves as the *input trigger* for step $k+1$, propagating changes through the graph structure of the knowledge graph.

Remark 1 (Consistency vs. Forecasting). It is crucial to distinguish our setting from Dynamic Knowledge Graph forecasting (Jiang et al., 2025), which typically aims to predict *future* facts based on historical trends. In contrast, our objective is *maintaining consistency*. We aim to derive the logical ramifications of an event to ensure the KG remains non-contradictory and complete immediately after the trigger. In other words, this is a synchronization of state, not a prediction of the future.

2.3 Error Accumulation in Cascading Knowledge Graph Updates

One of the biggest challenges in a cascading setting is error accumulation (He et al., 2025); a false positive at hop k can trigger a hallucinated cascade at $k+1$. Therefore, relying on deterministic predictions is insufficient for reliable system deployment.

Remark 2 (Risk Control with Prediction Sets). To resolve the issue of error propagation, instead of predicting a single deterministic graph state, we formulate the problem through the lens of uncertainty quantification (Angelopoulos and Bates, 2022). A trivial solution of predicting all possible relations would provide perfect recall, but renders an unusable complete KG. Conversely, a highly selective

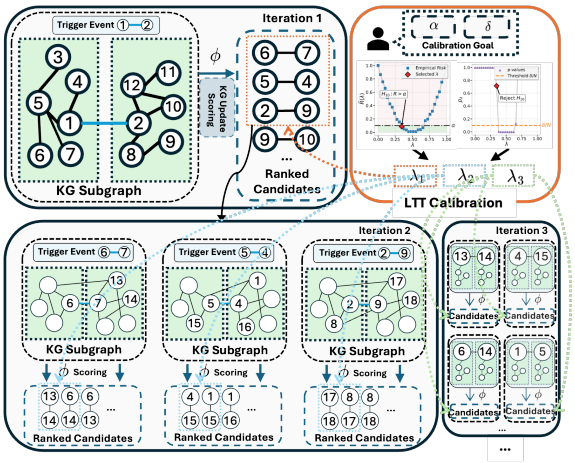


Figure 2: Overview of the proposed CASCADEKG.

model may miss critical cascading effects. To balance this, we seek to generate a *prediction set* $\mathcal{C}(e)$ of candidate updates that maximizes the retrieval of true updates while minimizing the size of the set $|\mathcal{C}(e)|$. Similar to uncertainty quantification tasks, formally, we aim to generate a prediction set $|\mathcal{C}(e)|$ that controls the expected recall risk:

$$\mathbb{E}\left[\frac{|\Delta_{\text{true}} \cap \mathcal{C}(e)|}{|\Delta_{\text{true}}|}\right] \geq 1 - \alpha \qquad (1)$$

where α is a user-specified error rate and Δ_{true} is ground truth update set.

3 CASCADEKG

We propose CASCADEKG, a unified framework designed to address the challenge of event-driven cascading updates in knowledge graphs. At a high level, CASCADEKG aims to tackle two fundamental challenges inherent to cascading knowledge graph updates. First, to address the issue of *error accumulation*, we propose a model-agnostic calibration framework based on Learn-then-Test (LTT) (Angelopoulos et al., 2022). This component provides rigorous statistical guarantees, ensuring that propagated updates remain within a confidence interval. Second, to effectively model *world state consistency* after an event, which necessitates reasoning beyond rigid graph structures, we introduce a text-augmented graph encoder-decoder framework that integrates LLMs to generate rich textual descriptions of triggering events, thereby enriching the structural and semantic node representations with external world knowledge. In the remainder of this section, we first introduce the calibration framework for controlling cascading error accumulation, and subsequently detail the architecture of our text-augmented graph reasoning model.

3.1 Risk Control for Cascading KG Updates

The cascading nature of event-driven Knowledge Graph (KG) predictions poses significant challenges for standard uncertainty quantification methods, such as standard Conformal Prediction (Vovk et al., 2005; Angelopoulos and Bates, 2022), which typically assume single-step inference. In our setting, updates are multi-step and recursive; a decision made at iteration k determines the input for iteration $k+1$. To control the cumulative error risk across this dynamic process, we leverage the Learn-Then-Test (LTT) (Angelopoulos et al., 2022) framework. LTT treats the calibration task as a multiple-hypothesis testing problem over the hyperparameter space, allowing us to find a configuration that controls the overall risk of the cascade.

The Learn-Then-Test Framework. Formally, we consider a loss function $L_\lambda : \mathcal{Y} \times \hat{\mathcal{Y}} \to \mathbb{R}$ parameterized by a configuration λ drawn from a multi-dimensional search space Λ. Our objective is to ensure that the expected risk remains below a user-specified threshold α, such that $\mathbb{E}[L_\lambda] \leq \alpha$. Given a calibration dataset $\mathcal{D}_{\text{cal}}$, the LTT framework identifies a subset of valid hyperparameters, denoted as $\Lambda_{\text{valid}} \subseteq \Lambda$, which satisfies the probabilistic guarantee:

$$\mathbb{P}\left(\sup_{\lambda \in \Lambda_{\text{valid}}} \mathbb{E}[L_\lambda] \leq \alpha \right) \geq 1 - \delta \qquad (2)$$

where the probability $\mathbb{P}$ is taken over the randomness of the calibration set $\mathcal{D}_{\text{cal}}$, and the expectation $\mathbb{E}$ is taken over the underlying data distribution. Effectively, δ bounds the probability that the selected valid set fails to meet the risk constraint. To achieve this, we formulate a hypothesis test for each configuration $\lambda \in \Lambda$ with the null hypothesis $H_0^\lambda : \mathbb{E}[L_\lambda] > \alpha$. We derive super-uniform p-values p_λ via concentration inequalities (e.g., Hoeffding-Bentkus) and subsequently apply a Family-Wise Error Rate (FWER) controlling procedure to isolate the set of valid configurations Λ_{valid} as proof in Appendix A.

KG Update Scoring. In the context of cascading KG updates, we define the configuration $\boldsymbol{\lambda}$ as a vector of decision thresholds, where λ_k corresponds to the cutoff threshold at cascade stage k. Let $\phi(\tau \mid e, \mathcal{G})$ be a scoring function (e.g., GNN-based models (Rong et al., 2020) or KG embedding models (Ji et al., 2022)) that assigns a likelihood score to a candidate triplet $\tau = (h, r, t)$ given the trigger event e and the current graph state $\mathcal{G}$.

For a given event e, the set of predicted updates $\mathcal{C}_{\boldsymbol{\lambda}}(e)$ is constructed recursively. At each stage k, the model evaluates all candidate triplets $\mathcal{E}_{cand}^{(k)}$ within the event-conditioned subgraph. The resulting state of the graph, $\mathcal{G}_k$, is formed by the set of all triplets that satisfy the stage-specific threshold λ_k. Formally, the updates at stage k are defined by the symmetric difference in graph states:

$$\Delta_k = \mathcal{G}_k \triangle \mathcal{G}_{k-1} = (\mathcal{G}_k \setminus \mathcal{G}_{k-1}) \cup (\mathcal{G}_{k-1} \setminus \mathcal{G}_k) \quad (3)$$

where $\mathcal{G}_k = \{\tau \in \mathcal{E}_{cand}^{(k)} \mid \phi(\tau \mid e, \mathcal{G}_{k-1}) \geq \lambda_k\}$.

To ensure the reliability of the update mechanism, we design our loss function to prioritize recall over precision. In the context of maintaining KG consistency, a missed update (false negative) leads to a corrupted graph state, whereas including extra candidate edges (false positives) is a tolerable cost that can be filtered by downstream verification (e.g. human verification in high-stakes applications). Therefore, we define the loss function based on the recall of the predicted set. We formulate $L_{\boldsymbol{\lambda}}$ as the *recall error*. For a ground truth update set Δ_{true}, the loss is defined as:

$$L_{\boldsymbol{\lambda}}(\Delta_{\text{true}}, \mathcal{C}_{\boldsymbol{\lambda}}(e)) = 1 - \frac{|\Delta_{\text{true}} \cap \mathcal{C}_{\boldsymbol{\lambda}}(e)|}{|\Delta_{\text{true}}|} \qquad (4)$$

In this formulation, a loss of 0 corresponds to perfect recall (all ground truth updates are successfully retrieved). By controlling this risk, we effectively guarantee that the expected recall remains above $1 - \alpha$, satisfying the user specified error rate.

The LTT algorithm then processes the calibration set $\mathcal{D}_{\text{cal}}$ to identify Λ_{valid}, the set of all threshold vectors that statistically satisfy the risk control. We select the optimal configuration:

$$\boldsymbol{\lambda}^* = \operatorname*{argmin}_{\boldsymbol{\lambda} \in \Lambda_{\text{valid}}} \sum_{e \in \mathcal{D}_{\text{cal}}} |\mathcal{C}_{\boldsymbol{\lambda}}(e)| \qquad (5)$$

i.e., the most informative (smallest) prediction set. In practice, Λ has low dimensionality (one threshold per cascade hop), so calibration is inexpensive and scales linearly with the grid granularity chosen by the practitioner; see Appendix G for empirical calibration-time measurements.

Theorem 1 (Probabilistic Guarantee). Let $\mathcal{D}_{\text{cal}}$ be the calibration set and Λ_{valid} be the set of configurations identified by the LTT procedure satisfying Eq. 2. For any chosen configuration $\hat{\boldsymbol{\lambda}} \in \Lambda_{\text{valid}}$, the expected recall on the test distribution satisfies the guarantee defined in Eq. 1 with probability at least $1 - \delta$ over the randomness of the calibration data.

The proof of Theorem 1 is given in Appendix B.

3.2 Event-Conditioned Graph Reasoning for Knowledge Graph Update Scoring

Although the Cascading Calibration in Section 3.1 provides a principled mechanism for controlling error propagation across update cascades, it does not address how candidate updates should be semantically prioritized at each stage. Calibration alone cannot compensate for a noisy or uninformed scoring function: without a way to distinguish event-consistent updates from spurious candidates, the combinatorial structure of KGs causes even calibrated prediction sets to grow impractically large.

A critical bottleneck therefore lies in designing an *event-conditioned update scoring function* that assigns meaningful plausibility scores to candidate structural changes based on the semantic context of the triggering event, incorporating external world knowledge to reason about relational consequences not explicitly encoded in the graph. To this end, we propose TAUS, a Text-Augmented Update Scoring framework that assigns event-conditioned scores to candidate structural changes in the localized KG by grounding graph-based reasoning in natural language descriptions of event-induced updates. Next, we provide the formal details of TAUS.

Event Context Generation. Encoding *world state consistency* directly into the model is challenging, as real-world events are heterogeneous and often governed by complex causal factors that are absent from the rigid KG structure. To address this, we leverage the extensive world knowledge of LLMs to bridge the gap between raw events and their relational consequences. The goal of this component is to provide a rich natural language context for the structural graph edits $\Delta\mathcal{T}$ that arise from an external trigger event e.

We adopt a *Teacher-Student* framework to capture this logic, which is visualized in Figure 3. First, we employ a powerful teacher LLM (M_t) to generate explanatory paragraphs for ground-truth event-update pairs $(e, \Delta\mathcal{T})$. In this step, the teacher explicitly articulates the underlying causal logic connecting the event to the corresponding topology changes. Second, we fine-tune a student model (M_s) on these generated rationales. This ensures the student learns to produce faithful text descriptions that reflect the necessary consistency updates, effectively internalizing the mapping from events to KG updates/modifications.

Specifically, let $\mathcal{D} = \{(e_i, \Delta\mathcal{T}_i)\}$ be the training set of events and their corresponding KG updates. We first use the teacher M_t to generate a descriptive rationale x_i for each pair, such that $x_i \sim M_t(e_i, \Delta\mathcal{T}_i)$. This yields an augmented dataset $\mathcal{D}_{\text{aug}} = \{(e_i, x_i)\}$. We then fine-tune the student model M_s, parameterized by θ, to minimize the negative log-likelihood of generating the rationale given the event:

$$\mathcal{L}_{\text{gen}}(\theta) = -\mathbb{E}_{(e,x)\sim\mathcal{D}_{\text{aug}}}\left[\log P_{M_s}(x \mid e; \theta)\right] \quad (6)$$

During inference, the smaller (and faster) M_s is used to generate the context $\hat{x}$ for a new event e, guiding the downstream graph reasoning.

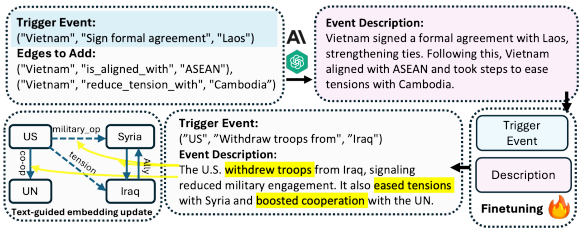


Figure 3: Overview of the proposed TAUS.

Text-Conditioned Graph Encoding. To incorporate the generated event context into the update scoring, we condition graph-based reasoning on semantic representations derived from text. Specifically, we use a pre-trained sentence encoder (e.g., Sentence-Transformers (Reimers and Gurevych, 2019)) to embed the textual rationale $\hat{x}$ produced by the student model M_s into a vector $\mathbf{z}_{\text{txt}} \in \mathbb{R}^{d_t}$. This embedding serves as the semantic conditioning signal that captures event-specific context.

To propagate this semantic signal through the knowledge graph, we instantiate the scoring function using a Relational Graph Convolutional Network (R-GCN) (Schlichtkrull et al., 2018) as the backbone encoder. The text embedding is projected into the graph feature space via a learnable transformation $W_t \in \mathbb{R}^{d\times d_t}$ and injected into the initial node representations, effectively shifting the initial feature distribution based on the event. For each node v, the conditioned initialization is given by $\mathbf{h}_v^{(0)} = \mathbf{h}_v^{\text{raw}} + W_t\mathbf{z}_{\text{txt}}$, where $\mathbf{h}_v^{\text{raw}}$ denotes the static pre-trained KG embedding. We adopt additive conditioning for its simplicity and model-agnostic nature; more expressive fusion mechanisms (gating, FiLM, MLP) yield only marginal improvements, as we show in Appendix F. These conditioned embeddings are then updated through the R-GCN message passing to capture relational dependencies, enabling relation-aware graph reasoning that integrates both local KG structural dependencies and event semantics for KG update scoring.

4 Experiments

In this section, we evaluate CASCADEKG in the context of event-driven cascading updates. Our experimental design follows a two-stage logic to address our core research questions.

First, regarding **RQ1 (Reasoning Capability)**, we evaluate the raw predictive performance of our proposed TAUS against existing KG embedding methods as scoring functions to determine if event-conditioning aids in identifying structural updates. Second, we study the effectiveness of our LTT-based framework for cascading KG updates as **RQ2 (Error Accumulation Control)**. We demonstrate the effectiveness of our plug-and-play calibration component by evaluating it across multiple scoring backbones and comparing it against post-hoc strategies. To rigorously address these questions, we construct two new benchmarks, ICEWS14-Event and YAGO-Event, which explicitly model the cascading changes introduced by external events in the knowledge graph.

4.1 Datasets

Evaluating performance on event-driven cascading KG updates requires a benchmark that captures both event-driven evolution and logical consistency constraints. However, most KG benchmarks are focused on KG completion that predicts missing edges in a fixed structure rather than modeling the causal ripple effects of external events. To bridge this gap and evaluate CASCADEKG, we thus curate two new datasets derived from real-world KGs. Additional dataset details are in Appendix C.

ICEWS14-Event. We construct the ICEWS14-Event dataset from The Integrated Crisis Early Warning System (ICEWS) (Boschee et al., 2015) dataset, which consists of structured geopolitical events between entities over time. To construct the knowledge graphs with corresponding updates, we incrementally build the KG based on the chronological order of incoming events.

YAGO-Event. The YAGO-Event dataset is constructed from the YAGO (Suchanek et al., 2007) knowledge base, which is a large semantic graph constructed from Wikipedia, WordNet, and GeoNames, containing rich entity-type and relation-type annotations. Because the YAGO3-10 dataset does not contain naturally occurring changes, we synthetically simulate event-triggered updates by applying a set of inference rules to its static triples.

4.2 Comparison Methods

To comprehensively evaluate the performance of CASCADEKG and KG update scoring module TAUS, we include two distinct families of baselines: established post-hoc calibration strategies and KG update scoring models.

4.2.1 Cascade Calibration Baselines

Aggregate (Agg). This strategy serves as a naive baseline that ignores the hierarchical structure of the graph data. Instead of distinguishing between hops, we collapse the calibration sets from both the one-hop and two-hop neighbors into a single pool. We then apply standard split conformal prediction on this aggregated set.

Bonferroni Conformal (Con). This baseline acknowledges the multi-hop structure but avoids the complexity of the Learn-then-Test framework. Instead of dynamically selecting hypotheses, we treat the one-hop and two-hop neighborhoods as fixed, independent calibration tasks. We then apply a standard Bonferroni correction to control the global risk by calibrating each hop independently at a stricter error rate of $\alpha/2$.

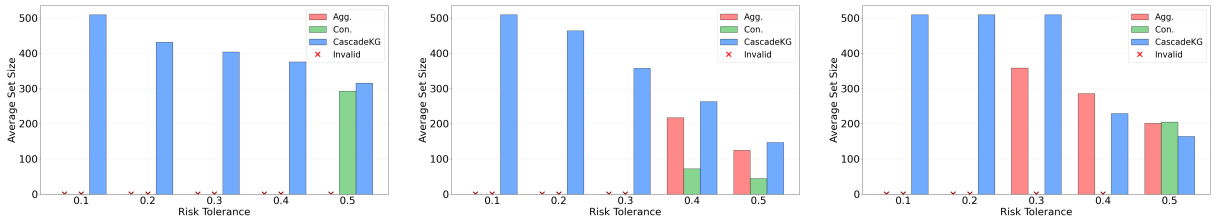
4.2.2 KG Update Scoring Baselines

To evaluate the effectiveness of our proposed text-augmented graph reasoning module TAUS for KG update scoring, we compare against the following representative KG update scoring methods: DistMult (Yang et al., 2015), R-GAT (Busbridge et al., 2019), TransE (Bordes et al., 2013), HittER (Chen et al., 2021), CompGCN (Vashishth et al., 2019), SimKGC (Wang et al., 2022), KGT5 (Saxena et al., 2022). More details are presented in Appendix D.

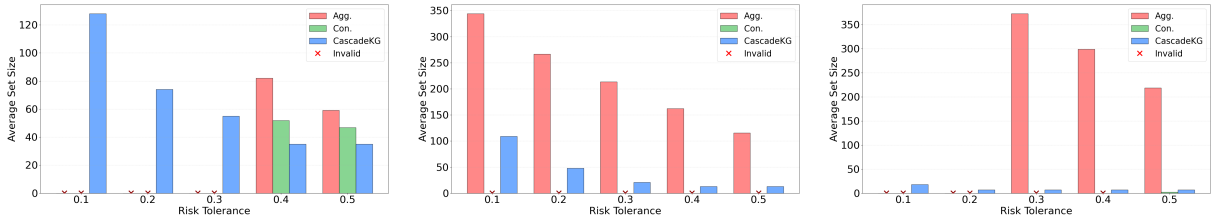
4.3 Evaluation Metrics

To comprehensively evaluate CASCADEKG, we utilize two distinct categories of metrics. First, to assess the raw predictive power of the scoring backbones (**RQ1**), we report standard classification metrics including Accuracy, F1-score, and Recall. These metrics measure the model's ability to distinguish valid additions and deletions from noise before calibration is applied.

Second, to evaluate the statistical reliability and efficiency of our calibration framework (**RQ2**), we follow the uncertainty quantification literature (Angelopoulos and Bates, 2022) and adopt two specialized metrics. **ECR (Empirical Coverage Rate)** measures the reliability of the system by calculating



(a) DistMult (b) TransE (c) HittER



(d) CompGCN (e) R-GAT (f) TAUS

Figure 4: Efficiency comparison of the calibration baselines on the six backbone models for YAGO-Event. Red cross is marked if the calibration method cannot achieve the desired risk tolerance. For each risk tolerance level, the lower the bar the better.

Table 1: KG update scoring results on ICEWS14-Event.

Model	F1	Acc.	Rec.	MRR	AUROC
DistMult	0.804	0.769	0.950	0.331	0.769
R-GAT	0.798	0.788	0.836	0.339	0.788
TransE	0.787	0.784	0.800	0.237	0.784
HittER	0.737	0.667	0.931	0.114	0.667
CompGCN	0.765	0.713	0.932	0.293	0.713
SimKGC	0.746	0.686	0.922	0.287	0.686
KGT5	0.748	0.688	0.926	0.318	0.688
TAUS	**0.880**	**0.866**	**0.985**	**0.435**	**0.866**

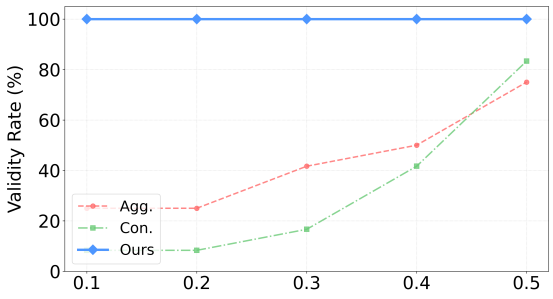


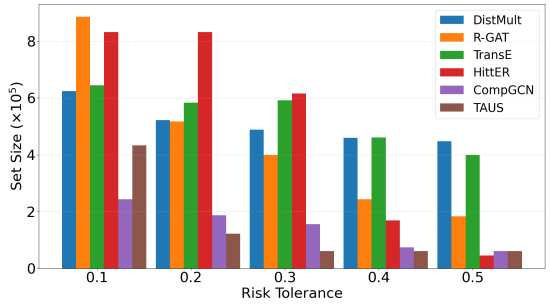
Figure 5: Calibration Methods Valid Coverage Rate.

the frequency with which the ground-truth update set is fully contained within the predicted set; a system is considered valid if its ECR remains above the $1-\alpha$ threshold. Conversely, **APSS (Average Prediction Set Size)** evaluates the efficiency of the model. While high ECR can be trivially achieved with large sets, a practical update mechanism must be informative. APSS measures the average cardinality of the predicted sets, where a smaller value indicates a more precise model that filters out irrelevant candidates while maintaining the required statistical guarantees.

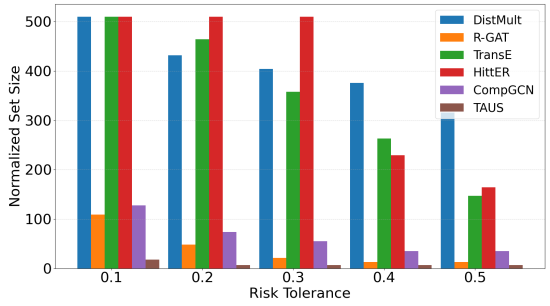
4.4 Experimental Results

Predictive Performance of the Scoring Module (RQ1). We first evaluate the fundamental ability of the underlying scoring models to correctly identify valid structural updates. We report the performance of various KG scoring backbones on the ICEWS14-Event datasets in Table 1. As shown in the table, our proposed TAUS backbone consistently demonstrates superior performance across all evaluation metrics on both datasets. Specifically, TAUS achieves an F1-score of 0.880 and an accuracy of 0.866, significantly outperforming traditional embedding methods like TransE, structural GNNs like R-GAT, and recent LM-augmented KGC models like SimKGC. Additional case study in Appendix E.1 further reveals that by integrating LLM-generated rationales, TAUS effectively grounds its reasoning in external world knowledge, capturing the semantic dependencies required to maintain global KG coherence in real-world.

Error Control (RQ2). To answer RQ2, we evaluate the effectiveness of the proposed CASCADEKG calibration framework by specifying varying risk levels α for the KG update scoring backbones. We report the relationship between coverage and prediction set size for the YAGO-Event dataset in Figure 4. For ECR that does not satisfy the risk tolerance, red crosses are used to mark the method.



(a) Results for ICEWS14-Event.



(b) Results for YAGO-Event.

Figure 6: Experimental results scoring backbones coupled with CascadeKG.

The detailed numerical results (ECR, etc.) are supplied in Appendix E. As observed across all model backbones, our calibration framework consistently achieves the smallest prediction set size while satisfying the target coverage. Notably, as visualized in Figure 5, CascadeKG succeeds in maintaining valid coverage across all specified risk levels. Results for ICEWS14-Event exhibit similar trend and are included in Appendix E.

Furthermore, in Figure 6, we plot the scoring backbones calibrated with CascadeKG. our result demonstrates that the effectiveness of the calibration framework is significantly enhanced when paired with stronger underlying reasoning models such as TauS, producing the smallest prediction sets. This highlights CascadeKG's ability to manage error propagation in cascading updates. By adaptively calibrating thresholds across different stages of the cascade, our framework avoids the "explosion" of predicted edges that typically renders uncalibrated KG update systems impractical.

5 Related Work

KG Predictions and Updating Most KG embedding and temporal KG work studies link prediction or forecasting under the assumption that each KG snapshot is internally coherent (Wang et al., 2017; Ji et al., 2022; Trivedi et al., 2017; Dasgupta et al., 2018; Liu et al., 2020). In contrast, our setting is triggered by external events that can invalidate previously correct facts and require consistency restoration via a sequence of iterative updates that cascade through relational dependencies.

Early KG update systems primarily synchronized KGs with structured sources (e.g., Wikipedia/encyclopedias) via curated extraction pipelines (Morsey et al., 2012; Liang et al., 2017). More recent work explores updating KGs from unstructured streams such as news (Tang et al., 2019), and related lines study KG consistency control (Wang et al., 2021; Padia et al., 2024).

KG Embedding Methods. Knowledge graph embedding methods are mainly developed for KG completion by scoring candidate triplets. In our experiments, we compare representative backbones spanning major modeling families: bilinear (DistMult) (Yang et al., 2015), translational (TransE) (Bordes et al., 2013), relational GNN encoders (R-GAT, CompGCN) (Busbridge et al., 2019; Vashishth et al., 2019), and Transformer-based KGE (HittER) (Chen et al., 2021). We note that a further discussion on these is provided in Appendix D. Recent KG completion models continue to build on these directions by adding stronger path reasoning or language-model-based formulations (e.g., NBFNet (Zhu et al., 2021), SimKGC (Wang et al., 2022) KGT5 (Saxena et al., 2022), KGT5-context (Kochsiek et al., 2023)), which are largely complementary to our cascade-level risk control and event-conditioned update scoring.

6 Conclusion

In this paper, we addressed the challenge of Knowledge Graph consistency through event-driven cascading updates. Our proposed framework, CascadeKG, utilizes the Learn-Then-Test calibration framework to provide rigorous statistical guarantees on multi-hop update reliability, effectively managing error propagation. Furthermore, we introduced a text-augmented GNN module TauS that leverages LLMs to enrich event representations with world knowledge. Experimental results on the newly curated ICEWS14-Event and YAGO-Event datasets demonstrate that CascadeKG significantly outperforms baseline scoring and calibration methods. By integrating structural reasoning with textual context, our approach offers an effective solution for maintaining global KG consistencies in real-world event-driven environments.

Limitations

While CASCADEKG provides rigorous statistical guarantees, a primary limitation lies in the assumption that calibration and test data share the same distribution; in volatile real-world scenarios, frequent re-calibration may be necessary to maintain reliability under distribution shift. Furthermore, although our theoretical symmetric difference formulation $\Delta_k = \mathcal{G}_k \triangle \mathcal{G}_{k-1}$ accounts for both additions and deletions, our current empirical evaluation on the ICEWS14-Event and YAGO-Event benchmarks focuses primarily on edge additions due to the growth-oriented nature of these historical datasets. Future work should explicitly investigate CASCADEKG's performance in high-deletion environments and develop adaptive calibration techniques to ensure long-term reliability under evolving world dynamics.

Acknowledgments

This research is supported by the National Science Foundation (NSF) under grant numbers IIS2239881, IIS2524380, and ECCS2325417.

References

Anastasios N. Angelopoulos and Stephen Bates. 2022. A gentle introduction to conformal prediction and distribution-free uncertainty quantification. *Preprint*, arXiv:2107.07511.

Anastasios N. Angelopoulos, Stephen Bates, Emmanuel J. Candès, Michael I. Jordan, and Lihua Lei. 2022. Learn then test: Calibrating predictive algorithms to achieve risk control. *Preprint*, arXiv:2110.01052.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2787–2795.

Elizabeth Boschee, Jennifer Lautenschlager, Sean O'Brien, Steve Shellman, James Starz, and Michael Ward. 2015. ICEWS Coded Event Data.

Dan Busbridge, Dane Sherburn, Pietro Cavallo, and Nils Y Hammerla. 2019. Relational graph attention networks. *arXiv preprint arXiv:1904.05811*.

Li Cai, Xin Mao, Yuhao Zhou, Zhaoguang Long, Changxu Wu, and Man Lan. 2024. A survey on temporal knowledge graph: Representation learning and applications. *Preprint*, arXiv:2403.04782.

Sanxing Chen, Xiaodong Liu, Jianfeng Gao, Jian Jiao, Ruofei Zhang, and Yangfeng Ji. 2021. HittER: Hierarchical transformers for knowledge graph embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10395–10407, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zhe Chen, Yuehan Wang, Bin Zhao, Jing Cheng, Xin Zhao, and Zongtao Duan. 2020. Knowledge graph completion: A review. *IEEE Access*, 8:192435–192456.

Shib Sankar Dasgupta, Swayambhu Nath Ray, and Partha Talukdar. 2018. Hyte: Hyperplane-based temporally aware knowledge graph embedding. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2001–2011.

Wenchong He, Zhe Jiang, Tingsong Xiao, Zelin Xu, and Yukun Li. 2025. A survey on uncertainty quantification methods for deep learning. *Preprint*, arXiv:2302.13425.

Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip S. Yu. 2022. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems*, 33(2):494–514.

Xuhui Jiang, Chengjin Xu, Yinghan Shen, Xun Sun, Lumingyuan Tang, Saizhuo Wang, Zhongwu Chen, Yuanzhuo Wang, and Jian Guo. 2025. On the evolution of knowledge graphs: A survey and perspective. *Preprint*, arXiv:2310.04835.

Adrian Kochsiek, Apoorv Saxena, Inderjeet Nair, and Rainer Gemulla. 2023. Friendly neighbors: Contextualized sequence-to-sequence link prediction. *arXiv preprint arXiv:2305.13059*.

Jiaqing Liang, Sheng Zhang, and Yanghua Xiao. 2017. How to keep a knowledge base synchronized with its encyclopedia source. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management (CIKM)*, pages 1917–1926. ACM.

Lihui Liu, Boxin Du, Heng Ji, and Hanghang Tong. 2020. Kompare: A knowledge graph comparative reasoning system. *Preprint*, arXiv:2011.03189.

Mohamed Morsey, Jens Lehmann, Sören Auer, Claus Stadler, and Sebastian Hellmann. 2012. Dbpedia and the live extraction of structured data from wikipedia. *Program*, 46(2):157–181.

Bo Ni, Zheyuan Liu, Leyao Wang, Yongjia Lei, Yuying Zhao, Xueqi Cheng, Qingkai Zeng, Luna Dong, Yinglong Xia, Krishnaram Kenthapadi, Ryan Rossi, Franck Dernoncourt, Md Mehrab Tanjim, Nesreen Ahmed, Xiaorui Liu, Wenqi Fan, Erik Blasch, Yu Wang, Meng Jiang, and Tyler Derr. 2025a. Towards trustworthy retrieval augmented generation for large language models: A survey. *Preprint*, arXiv:2502.06872.

Bo Ni, Yu Wang, Lu Cheng, Erik Blasch, and Tyler Derr. 2025b. Towards trustworthy knowledge graph reasoning: An uncertainty aware perspective. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pages 12417–12425.

Ankur Padia, Francis Ferraro, and Tim Finin. 2024. Enhancing knowledge graph consistency through open large language models: A case study. *Proceedings of the AAAI Symposium Series*, 3(1):203–208.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. 2017. Film: Visual reasoning with a general conditioning layer. *Preprint*, arXiv:1709.07871.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *Preprint*, arXiv:1908.10084.

Yu Rong, Tingyang Xu, Junzhou Huang, Wenbing Huang, Hong Cheng, Yao Ma, Yiqi Wang, Tyler Derr, Lingfei Wu, and Tengfei Ma. 2020. Deep graph learning: Foundations, advances and applications. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 3555–3556.

Apoorv Saxena, Adrian Kochsiek, and Rainer Gemulla. 2022. Sequence-to-sequence knowledge graph completion and question answering. *arXiv preprint arXiv:2203.10321*.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *The Semantic Web (ESWC)*, pages 593–607. Springer.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, page 697–706, New York, NY, USA. Association for Computing Machinery.

Jizhi Tang, Yansong Feng, and Dongyan Zhao. 2019. Learning to update knowledge graphs by reading news. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2632–2641, Hong Kong, China. Association for Computational Linguistics.

Rakshit Trivedi, Hanjun Dai, Yichen Wang, and Le Song. 2017. Know-evolve: Deep temporal reasoning for dynamic knowledge graphs. In *international conference on machine learning*, pages 3462–3471. PMLR.

Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha Talukdar. 2019. Composition-based multi-relational graph convolutional networks. *arXiv preprint arXiv:1911.03082*.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. 2005. *Algorithmic Learning in a Random World*. Springer.

Liang Wang, Wei Zhao, Zhuoyu Wei, and Jingming Liu. 2022. Simkgc: Simple contrastive knowledge graph completion with pre-trained language models. *arXiv preprint arXiv:2203.02167*.

Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE transactions on knowledge and data engineering*, 29(12):2724–2743.

Xiangyu Wang, Lyuzhou Chen, Taiyu Ban, Muhammad Usman, Yifeng Guan, Shikang Liu, Tianhao Wu, and Huanhuan Chen. 2021. Knowledge graph quality control: A survey. *Fundamental Research*, 1(5):607–626.

Yu Wang, Amin Javari, Janani Balaji, Walid Shalaby, Tyler Derr, and Xiquan Cui. 2024. Knowledge graph-based session recommendation with session-adaptive propagation. In *Companion Proceedings of the ACM Web Conference 2024*, pages 264–273.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *International Conference on Learning Representations (ICLR)*.

Zhaocheng Zhu, Zuobai Zhang, Louis-Pascal Xhonneux, and Jian Tang. 2021. Neural bellman-ford networks: A general graph neural network framework for link prediction. *Advances in neural information processing systems*, 34:29476–29490.

A Proof on LTT Risk Control P-Value selection via FWER

Let Λ be a finite set of configurations with $m := |\Lambda|$. For each $\lambda \in \Lambda$, let $L_\lambda \in [0,1]$ denote a bounded loss random variable over the data-generating distribution, and let $\alpha \in (0,1)$ be a target risk level. Let $D_{\text{cal}} = \{\mathcal{C}_{\boldsymbol{\lambda}}(e))_{i=1}^n\}$ be an i.i.d. calibration sample from the underlying distribution, and define the empirical risk

$$\widehat{R}_\lambda := \frac{1}{n}\sum_{i=1}^{n} L_\lambda(\Delta_{\text{true}}, \mathcal{C}_{\boldsymbol{\lambda}}(e))_i).$$

According to Hoeffding-Benktus, we can calculate p_λ for each $\lambda \in \Lambda$

$$p_\lambda = \mathbb{P}\Big(\bar{L} \le \hat{R}_\lambda \,\Big|\, \mathbb{E}[L_\lambda] = \alpha\Big) \;\le\; \mathbb{P}\Big(\text{Bin}(n,\alpha) \le n\hat{R}_\lambda\Big)$$

For each $\lambda \in \Lambda$, consider the null hypothesis

$$H_0^\lambda :\; \mathbb{E}[L_\lambda] > \alpha.$$

Assume we construct a p-value p_λ satisfying the super-uniform property: for all $t \in [0, 1]$,

$$\sup_{\mathbb{P} \in H_0^\lambda} \mathbb{P}(p_\lambda \leq t) \ \leq \ t. \tag{7}$$

Define the Bonferroni-FWER valid set

$$\Lambda_{\text{valid}} := \{\lambda \in \Lambda : \ p_\lambda \leq \delta/m\},$$

where $\delta \in (0, 1)$ is a user-specified failure probability. Then

$$\mathbb{P}\Big(\sup_{\lambda \in \Lambda_{\text{valid}}} \mathbb{E}[L_\lambda] \ \leq \ \alpha \Big) \ \geq \ 1 - \delta, \tag{8}$$

where the probability is over the randomness of the calibration sample D_{cal}.

Proof. Define the set of truly invalid configurations

$$\Lambda_{\text{bad}} := \{\lambda \in \Lambda : \ \mathbb{E}[L_\lambda] > \alpha\}.$$

We first bound the probability that any truly bad configuration is incorrectly included in Λ_{valid}:

$$\mathbb{P}\big(\Lambda_{\text{valid}} \cap \Lambda_{\text{bad}} \neq \emptyset\big) = \mathbb{P}\Big(\bigcup_{\lambda \in \Lambda_{\text{bad}}} \{p_\lambda \leq \delta/m\}\Big).$$

By the union bound,

$$\mathbb{P}\Big(\bigcup_{\lambda \in \Lambda_{\text{bad}}} \{p_\lambda \leq \delta/m\}\Big) \leq \sum_{\lambda \in \Lambda_{\text{bad}}} \mathbb{P}(p_\lambda \leq \delta/m).$$

For any $\lambda \in \Lambda_{\text{bad}}$, the null H_0^λ holds, so by the super-uniform property (7) with $t = \delta/m$,

$$\mathbb{P}(p_\lambda \leq \delta/m) \ \leq \ \delta/m.$$

Therefore,

$$\mathbb{P}\big(\Lambda_{\text{valid}} \cap \Lambda_{\text{bad}} \neq \emptyset\big) \leq \sum_{\lambda \in \Lambda_{\text{bad}}} \delta/m \leq |\Lambda_{\text{bad}}| \cdot \delta/m \leq \delta.$$

Equivalently,

$$\mathbb{P}\big(\Lambda_{\text{valid}} \subseteq \Lambda \setminus \Lambda_{\text{bad}}\big) \ \geq \ 1 - \delta,$$

i.e., with probability at least $1 - \delta$ over D_{cal}, every $\lambda \in \Lambda_{\text{valid}}$ satisfies $\mathbb{E}[L_\lambda] \leq \alpha$. On this event,

$$\sup_{\lambda \in \Lambda_{\text{valid}}} \mathbb{E}[L_\lambda] \ \leq \ \alpha,$$

which yields (8). □

B Probabilistic Guarantee Proof of our Cascading Calibration Framework

Here we provide the detailed proof of Theorem 1.

Proof. Recall that our loss function is defined as the recall error:

$$L_{\boldsymbol{\lambda}} = 1 - \text{Recall}_{\boldsymbol{\lambda}} = 1 - \frac{|\Delta_{\text{true}} \cap \mathcal{C}_{\boldsymbol{\lambda}}(e)|}{|\Delta_{\text{true}}|} \tag{9}$$

The Learn-Then-Test condition (Eq. 2) guarantees that:

$$\mathbb{P}\left(\sup_{\lambda \in \Lambda_{\text{valid}}} \mathbb{E}[L_\lambda] \leq \alpha \right) \geq 1 - \delta \tag{10}$$

Let E be the event that the calibration succeeds, i.e., $\forall \lambda \in \Lambda_{\text{valid}}, \mathbb{E}[L_\lambda] \leq \alpha$. The algorithm ensures $\mathbb{P}(E) \geq 1 - \delta$. Conditioned on event E, for any selected $\hat{\boldsymbol{\lambda}} \in \Lambda_{\text{valid}}$, we have:

$$\mathbb{E}[L_{\hat{\boldsymbol{\lambda}}}] \leq \alpha \tag{11}$$

$$\mathbb{E}[1 - \text{Recall}_{\hat{\boldsymbol{\lambda}}}] \leq \alpha \tag{12}$$

$$\mathbb{E}[\text{Recall}_{\hat{\boldsymbol{\lambda}}}] \geq 1 - \alpha \tag{13}$$

Thus, the expected recall is lower-bounded by $1-\alpha$ with probability at least $1-\delta$. When δ is sufficiently small, we can approximate this probabilistic bound as Eq. 1, effectively guaranteeing that the system satisfies the target recall. □

C Dataset Details: Construction and Characteristics

In this section, we provide additional details regarding the construction and characteristics of the two newly constructed benchmarks used in our evaluation: ICEWS14-Event and YAGO-Event.

For both datasets, ground-truth cascading updates are obtained by recursively applying predefined consistency rules to an initial trigger event until the graph reaches a stable, consistent state. This iterative process mimics the "ripple effect" of real-world event consequences. We summarize the key statistics for both datasets in Table 2 and provide the exhaustive set of logical consistency rules in Tables 3 and 4.

C.1 ICEWS14-Event

The ICEWS14-Event dataset is derived from the Integrated Crisis Early Warning System (ICEWS) (Boschee et al., 2015) repository, which records geopolitical interactions between international actors. For each time step t, we define the

Table 2: Statistics of the constructed ICEWS14-Event and YAGO-Event datasets. We report the number of event samples, unique entities, and unique relations active in each split. The **Avg. Subgraph Nodes** column reports the mean number of entities in the event-conditioned subgraph (local L-hop neighborhood) $\pm$ the standard deviation.

Dataset	Split	# Samples	# Entities	# Relations	Avg. Subgraph Nodes
ICEWS14-Event	Train	7,605	1,230	160	41.90 $\pm$ 28.81
	Valid	108	114	36	12.93 $\pm$ 8.95
	Test	107	147	50	14.60 $\pm$ 10.46
	Overall	*7,820*	*1,252*	*161*	–
YAGO-Event	Train	20,152	22,320	43	96.88 $\pm$ 12.81
	Valid	1,963	9,545	43	95.69 $\pm$ 15.56
	Test	2,034	9,597	41	95.15 $\pm$ 16.09
	Overall	*24,149*	*26,851*	*43*	–

Table 3: Selected logical consistency rules for **ICEWS14-Event**. The rules define how a trigger event (A, r_{trig}, B) interacts with precondition (B, r_{cond}, C) to infer (A, r_{new}, C).

Trigger Event	Precondition	Inferred Update	Op
Positive Cascades (Additions)			
Sign formal agreement	Engage in diplomatic coop.	Engage in diplomatic coop.	+
Provide military aid	Cooperate militarily	Cooperate militarily	+
Declare truce, ceasefire	Engage in diplomatic coop.	De-escalate intent	+
Occupy territory	Engage in diplomatic coop.	Complain officially	+
Sign formal agreement	Grant diplo. recognition	Engage in diplomatic coop.	+
Negative Cascades (Deletions)			
Break diplomatic rel.	Engage in diplomatic coop.	Break diplomatic rel.	–
Impose embargo	Cooperate economically	Cooperate economically	–
Threaten with mil. force	Engage in diplomatic coop.	Increase armed forces	–
Multi-hop / Complex Rules			
Sign formal agreement	Diplo. Coop $\rightarrow$ Mil. Coop	Cooperate militarily	+
Impose sanctions	Trade $\rightarrow$ Trade	Impose sanctions	+

knowledge graph $\mathcal{G}_t$ as the aggregation of all historical facts observed up to day t. Changes introduced by new events on day $t+1$ are denoted as $\Delta\mathcal{T}_{t+1}$.

To model logical dependencies, we apply a set of hand-crafted consistency rules that propagate relational changes beyond the explicitly stated event facts. For example, if *Country A* signs a formal agreement with *Country B*, and *Country B* already maintains a military cooperation with *Country C*, our framework infers a new cooperative link between *Country A* and *Country C*. We note that for the ICEWS14-Event benchmark, the graph grows monotonically over time; i.e., updates in our current experiments consist solely of edge additions.

C.2 YAGO-Event

The YAGO-Event dataset is built upon the YAGO3-10 knowledge base (Suchanek et al., 2007). Unlike ICEWS, which is inherently temporal, YAGO is a large-scale semantic graph. To evaluate cascading updates, we synthetically simulate event-triggered changes by selecting initial event triples and applying two-hop rule patterns that reflect plausible downstream relational inconsistencies.

For each trigger event, we generate additional updates based on these patterns to evaluate the model's capacity for multi-hop reasoning in a dense, semantically rich environment. For example, as shown in Table 4, an academic trigger event such as *Person A* `hasAcademicAdvisor` *Person B*—where *Person B* previously `graduatedFrom` *University C*—would recursively infer the update *Person A* `academicLineageFrom` *University C*. Similarly, in professional contexts, if a person `playsFor` a team that `isLocatedIn` a specific city, our rules infer a corresponding `livesIn` relation.

Unlike ICEWS14-Event, the rule set for YAGO-Event is specifically designed to test structural dependencies within a localized context. This provides a controlled environment for measuring calibration efficiency, as it requires the model to ground its predictions in both the initial event and the surrounding static topology to maintain world-state consistency.

Table 4: Selected horn-clause rules for **YAGO-Event** that infer new relations from 2-hop paths in the graph.

Path Step 1 (r_1)	**Path Step 2** (r_2)	**Inferred Relation**	**Mode**
Social & Family			
hasChild	hasChild	hasGrandchild	Fwd
isLeaderOf	isLocatedIn	livesIn	Fwd
playsFor	isLocatedIn	livesIn	Fwd
Academic & Professional			
graduatedFrom	isLocatedIn	wasEducatedIn	Fwd
hasAcademicAdvisor	graduatedFrom	academicLineageFrom	Fwd
owns	isLocatedIn	hasBusinessIn	Fwd
Creative Collaborations			
actedIn	directed	collaboratedWith	Cross
actedIn	wroteMusicFor	sharedCreativeWork	Cross
directed	edited	coDirectedOrEdited	Cross

D Baseline KG Embedding Descriptions

In the below we provide some additional details on the KG embedding baselines we compare our proposed CASCADEKG against in this work.

DistMult (Yang et al., 2015): As a widely-used knowledge graph embedding model, DistMult scores triplets using a bilinear function, capturing the semantic relationships between entities.

R-GAT (Busbridge et al., 2019): Relational Graph Attention Networks (R-GAT) extends GAT to handle multi-relational data, allowing the model to weigh the importance of different neighbors.

TransE (Bordes et al., 2013): TransE is a translation-based embedding model that represents relations as translations between embeddings.

HittER (Chen et al., 2021): Hierarchical Transformers for Knowledge Graph Embeddings (HittER) is a Transformer-based model that jointly learns entity-relation composition and relational contextualization.

CompGCN (Vashishth et al., 2019): Composition-based Multi-Relational Graph Convolutional Networks (CompGCN) integrate multi-relational GCNs with KG embedding techniques.

SimKGC (Wang et al., 2022): SimKGC is a contrastive KG completion model built on pre-trained language models, encoding entity and relation descriptions via a bi-encoder and scoring candidate triplets with an InfoNCE objective augmented by in-batch and pre-batch negatives.

KGT5 (Saxena et al., 2022): KGT5 casts KG completion as a sequence-to-sequence task with a T5 backbone, verbalizing (head, relation) queries as text and autoregressively generating candidate tail entities, allowing the model to leverage pretrained language knowledge for triplet scoring.

E Case Study and Additional Experiments

E.1 Case Study

To demonstrate how the inclusion of additional world context can enhance the effectiveness of event-based KG update prediction, we conduct a qualitative case study on an instance where TextGCN significantly outperforms the DistMult baseline, as shown in Table 5. In this scenario, the model is tasked with predicting a missing link involving South Korea and a combatant group following a trigger event between Vietnam and Iran. As shown in the example, *TextGCN* was able to correctly rank the ground truth triplet atw1.0 for both head and tail predictions. In contrast, *DistMult*, which relies solely on existing static graph structures and lacks access to the dynamic narrative of the event, produces much lower rankings.

The world context generated by the fine-tuned LLM explicitly mentions Vietnam expanding its outreach to South Korea. By encoding this textual information, TextGCN can capture the transitive diplomatic shift that a purely structural model misses. This valuable insight thus allows the model to reason that the recent formal agreement between Vietnam and Iran has downstream effects on South Korea's diplomatic and security positioning.

E.2 Additional Experiment Details

In this section, we provide the complete empirical results for all tested models. Detailed efficiency comparisons for the ICEWS14-Event dataset are presented in Figure 7, while the comprehensive numerical performance for both benchmarks is summarized in Table 6 and Table 7.

As visualized in Figure 7, the calibration trends on ICEWS14-Event mirror those observed on YAGO-Event (Figure 4). Across all scoring calibration methods, our proposed CASCADEKG consistently produces the most compact (smallest) prediction sets while strictly adhering to the user-specified risk tolerance α. These results further validate the robustness of our Learn-Then-Test framework across different graph topologies and relational distributions.

The raw numerical data in Table 6 and Table 7 provides a granular view of the Empirical Coverage Rate (ECR) and Average Prediction Set Size (APSS). These tables demonstrate that CASCADEKG significantly reduces the APSS compared to the *Aggregate* and *Conformal* baselines.

Table 5: Case Study. We list the ranking of the ground truth triplet under the method names.

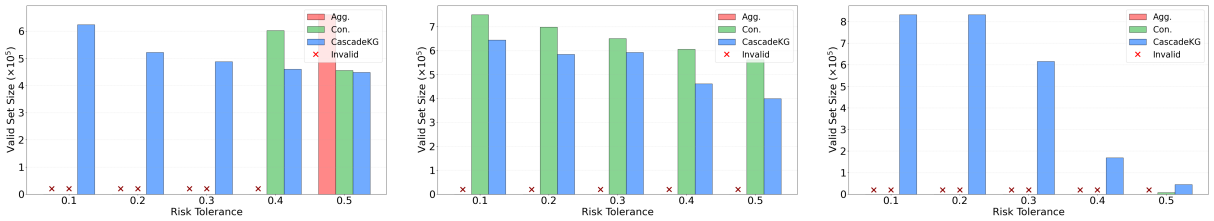
Property	**Value**	**TextGCN**	**DistMult**
Context (TextGCN)	Vietnam signs agreement with Iran; expands to South Korea, Mexico, etc.	–	–
Trigger Event	(Vietnam, Sign formal agreement, Iran, 2014-10-19)	–	–
Triplet	(**South Korea**, Sign formal agreement, **Combatant (Al Qaeda)**)	1.0 / 1.0	22.0 / 16.0

Model	Set Size (SS) / Empirical Coverage Rate (ECR)				
	α=0.1	α=0.2	α=0.3	α=0.4	α=0.5
DistMult + Agg.	800624 / 0.15	775182 / 0.32	761812 / 0.37	715362 / 0.49	660914 / 0.39
DistMult + Con.	703757 / 0.36	668810 / 0.36	636175 / 0.38	602700 / 0.39	456026 / 0.44
DistMult + Ours	**623806** / 0.07	**522011** / 0.18	**488106** / 0.23	**460345** / 0.43	**448201** / 0.44
R-GAT + Agg.	809140 / 0.00	301022 / 0.36	275660 / 0.45	265590 / 0.46	260080 / 0.46
R-GAT + Con.	422049 / 0.43	304502 / 0.45	292531 / 0.45	281099 / 0.45	265083 / 0.46
R-GAT + Ours	**886443** / 0.00	**517983** / 0.05	**399970** / 0.19	**243539** / 0.24	**182992** / 0.37
TransE + Agg.	414932 / 0.14	223511 / 0.21	160232 / 0.53	99342 / 0.56	71755 / 0.52
TransE + Con.	750339 / 0.04	698616 / 0.08	650705 / 0.13	606146 / 0.18	561997 / 0.22
TransE + Ours	**644920** / 0.02	**583993** / 0.03	**592330** / 0.10	**461069** / 0.11	**399659** / 0.15
HittER + Agg.	799278 / 0.28	797914 / 0.31	613198 / 0.37	789465 / 0.41	756304 / 0.52
HittER + Con.	693003 / 0.36	15998 / 0.42	10277 / 0.43	8345 / 0.44	7076 / 0.44
HittER + Ours	**832363** / 0.01	**832363** / 0.01	**615598** / 0.22	**168502** / 0.30	**45011** / 0.31
CompGCN + Agg.	702483 / 0.12	560816 / 0.16	341105 / 0.26	268027 / 0.44	243903 / 0.50
CompGCN + Con.	353033 / 0.38	74216 / 0.36	48989 / 0.40	46873 / 0.44	16862 / 0.58
CompGCN + Ours	**243984**/ 0.00	**186690** / 0.06	**156290** / 0.05	**74086** / 0.05	**61015**/ 0.05
TAUS + Agg.	798554 / 0.02	760599 / 0.06	752152 / 0.06	743308 / 0.06	728563 / 0.08
TAUS + Con.	73811 / 0.25	62118 /0.25	51877 / 0.26	48968 / 0.28	46903 / 0.34
TAUS + Ours	**433251** / 0.00	**121994** / 0.00	**61006** / 0.06	**61006** / 0.06	**61006**/ 0.06

Table 6: Set Size (SS) and Empirical Coverage Rate (ECR) under varying risks α on the ICEWS14-Event dataset. We note that our overall proposed framework CASCADEKG is represented as the TAUS+ Ours setting.

Model	Set Size (SS) / Empirical Coverage Rate (ECR)				
	α=0.1	α=0.2	α=0.3	α=0.4	α=0.5
DistMult + Agg.	204.49 / 0.14	181.25 / 0.23	163.71 / 0.33	150.49 / 0.42	137.63 / 0.52
DistMult + Con.	374.96 / 0.30	341.56 / 0.34	323.63 / 0.37	307.17 / 0.41	**293.21** / 0.44
DistMult + Ours	**510** / 0.00	**432** / 0.09	**404** / 0.09	**376** / 0.19	316 / 0.35
R-GAT + Agg.	344.21 / 0.09	266.61 / 0.15	213.60 / 0.22	162.29 / 0.31	115.39 / 0.41
R-GAT + Con.	64.87 / 0.29	33.74 / 0.33	20.47 / 0.37	15.40 /0.41	12.61 / 0.44
R-GAT + Ours	**109** / 0.02	**48** / 0.09	**21** / 0.17	**13** / 0.24	**13** / 0.24
TransE + Agg.	411.81 / 0.10	349.06 / 0.23	290.44 / 0.31	217.21 / 0.39	124.71 / 0.46
TransE + Con.	320.38 / 0.28	238.07 / 0.31	137.96 / 0.35	**72.52** / 0.39	**44.61** / 0.44
TransE + Ours	**510** / 0.00	**464** / 0.04	**358** / 0.15	263 / 0.27	147 / 0.42
HittER + Agg.	443.96 / 0.19	404.64 / 0.23	358.27 / 0.29	285.80 / 0.38	201.55 / 0.46
HittER + Con.	439.03 / 0.32	433.50 / 0.34	431.53 / 0.38	418.59 / 0.42	205.32 / 0.47
HittER + Ours	**510** / 0.00	**510** / 0.00	**510** / 0.00	**229** / 0.30	**164** / 0.32
CompGCN + Agg.	156.49 / 0.16	126.63 / 0.25	103.45 / 0.36	82.02 / 0.40	59.17 / 0.48
CompGCN + Con.	88.62 / 0.29	67.73 / 0.33	58.12 / 0.36	51.80 / 0.40	46.85 / 0.43
CompGCN + Ours	**128** / 0.01	**74** / 0.06	**55** / 0.11	**35** / 0.24	**35** / 0.24
TAUS + Agg.	472.98 / 0.14	430.27 / 0.22	372.79 / 0.30	299.16 / 0.37	218.87 / 0.43
TAUS + Con.	4.28 / 0.29	3.49 /0.33	3.10 / 0.37	2.79 / 0.41	**2.52** / 0.45
TAUS + Ours	**18**/ 0.02	**7** / 0.05	**7** / 0.05	**7** / 0.05	**7** / 0.05

Table 7: Normalized Set Size (SS) and Empirical Coverage Rate (ECR) under varying risks α on the YAGO-Event dataset. We note that our overall proposed framework CASCADEKG is represented as the TAUS+ Ours setting.



(a) DistMult (b) TransE (c) HittER

(d) CompGCN (e) R-GAT (f) TAUS

Figure 7: Efficiency comparison of the calibration baselines on the six backbone models for ICEWS14-Event. Red cross is marked if the calibration method cannot achieve the desired risk tolerance. For each risk tolerance level, the lower the bar the better.

Table 8: Fusion mechanism ablation on YAGO-Event. All variants satisfy the user-specified error rate, confirming that the LTT guarantee is robust to the choice of text-to-graph fusion.

Variant	α_1	α_2	Fail Rate	APSS
Additive (TAUS)	0.60	0.30	0.00	433
Gated	0.50	0.30	0.00	372
MLP	0.50	0.30	0.00	322
FiLM	0.50	0.30	0.00	375

F Ablation on Text-to-Node Fusion Mechanisms

TAUS uses a simple additive fusion of the text embedding into the initial node representation: $h_v^{(0)} = h_v^{\text{raw}} + W_t z_{\text{txt}}$. To verify that additive conditioning is not a bottleneck, we compare three more expressive alternatives:

- **Gated fusion**: a text-derived gate modulates an update vector, $h_v^{(0)} = h_v^{\text{raw}} + g \odot u$, with $g = \sigma(W_g z_{\text{txt}})$ and $u = W_u z_{\text{txt}}$.
- **FiLM** (Perez et al., 2017): feature-wise linear modulation, $h_v^{(0)} = \gamma(z_{\text{txt}}) \odot h_v^{\text{raw}} + \beta(z_{\text{txt}})$.
- **MLP fusion**: concatenate and project, $h_v^{(0)} = \text{MLP}([h_v^{\text{raw}}; W_t z_{\text{txt}}])$ with a two-layer MLP.

Results on YAGO-Event at $\alpha = 0.1$ are reported in Table 8. All variants satisfy the target risk bound, validating that the LTT guarantee holds across fusion choices. The gated, MLP, and FiLM variants yield modestly tighter prediction sets, but the gap is small and the additive design remains competitive while being the simplest option.

Table 9: Calibration time scales linearly with grid size and stays tractable even at fine granularity ($\sim$14 min). Split conformal reaches only 61.8% coverage, failing the target guarantee, whereas all LTT configurations meet the coverage target. Calibration is done one-time offline; inference time is identical across methods.

Method	Grid	Cal. Time (s)	Coverage	APSS
Split Conformal	—	4.45	0.6184	504
LTT-Pareto (coarse)	36	72.14	1.0000	433
LTT-Pareto (medium)	121	241.39	1.0000	433
LTT-Pareto (fine)	441	869.09	1.0000	402

G Calibration Efficiency

A natural concern with LTT-based calibration is the cost of searching the threshold grid Λ. In CASCADEKG, Λ is low dimensional—one threshold per cascade hop, with $K = 2$ in our experiments—so its size does not scale with the size of the knowledge graph and is fully user-controlled. The per-event computational bottleneck lies in evaluating the scoring function over local event-conditioned subgraphs (average size 15–97 nodes per Table 2), not over the full KG.

Table 9 reports calibration time, empirical coverage, and average prediction set size on YAGO-Event across three grid granularities, compared against standard split conformal prediction.