TeCES: Collaborative Geometric Knowledge Representation Framework under Evolving Fact Snapshots

Jiujiang Guo[1] **Zhengliang Guo**[2] **Kai Wang**[3] **Meiyang Wang**[4] **Dehua Peng**[1]
Shaozu Yuan[5] **Chengyin Hu**[2*] **Shuan Ai**[6] **Yiwei Wei**[2*]

[1]Tianjin University, China
[2]China University of Petroleum-Beijing at Karamay, China
[3]State Grid Hebei Electric Power Company, China
[4]Hebei University of Science and Technology, China
[5]Meituan, China
[6]Jilin Business and Technology College, China

{jiujiangguo, dehuapeng}@tju.edu.cn
2025216806@st.cupk.edu.cn, {cyhu, weiyiwei}@cupk.edu.cn
17532106615@163.com, wmy@hebust.edu.cn
1586687565@qq.com, aishuan@jlbtc.edu.cn

Abstract

Existing knowledge graph completion research is gradually shifting from representing logical semantics of static facts to modeling evolving semantics of temporal facts, yet lacks collaborative modeling of both within a unified framework. To this end, we use concept of snapshots to decompose fact features into two complementary mechanisms: (a) intra-snapshot semantic coupling, where entities and relations exhibit snapshot-specific meanings through multidimensional interactions; (b) trans-snapshot evolutionary synergy, where relations between entities evolve across snapshots and manifest varying states. These snapshot mechanisms jointly reveal underlying logic of facts. To track them, we propose TeCES, a framework for high-fidelity modeling of evolving snapshots. TeCES embeds facts into a 2-grade geometric algebra (GA) system to capture complex semantics via multilevel structures. Temporal information is attached to each entity for mapping into snapshot spaces, while relations and timestamps are reconfigured into composite GA representations. Geometric products enable multidimensional interactions, revealing relation state changes over time. Lastly, the head entity at each snapshot combines with fused temporal-relational representation via geometric product to approximate the target tail entity at multiple levels. Overall, TeCES supports joint modeling of evolving snapshots within a lightweight GA system and significantly outperforms SOTA models on six benchmarks.

1 Introduction

Knowledge graphs (KGs) integrate cross-disciplinary theories to reveal and mine meta-semantics of real-world knowledge, and explicitly express it in a structured form. They are typically classified into static knowledge graphs (SKGs) and temporal knowledge graphs (TKGs), with SKGs and TKGs storing facts as triples and quadruples, respectively. Existing KGs (e.g., Acekg (Wang et al., 2018), MMSD (Yuan et al., 2025), and ImgFact (Liu et al., 2024)) contain a vast amount of information, and many have been successfully applied in various domains, including information retrieval, supply chain management, industry-oriented KGs (Guo et al., 2025b), and sarcasm detection (Wei et al., 2024).

To sustainably manage intricate knowledge from physical world, researchers delve into the essence of facts, extract underlying rules, and subsequently develop a variety of robust models for knowledge graph completion (KGC). Recent KGC research is shifting from modeling the logical semantics of static facts to capturing the evolving logical semantics of temporal facts. However, a unified and deeply collaborative framework for jointly modeling the logical semantics of static and temporal facts is still lacking. To further uncover the underlying principles embedded in facts under temporal evolution, we use concept of snapshots (Chandy and Lamport, 1985) to deeply deconstruct the underlying static logic and temporal evolution mechanisms of real-world facts: (a) **Intra-snapshot semantic coupling.** Entities and relations each carry snapshot-specific semantics, and through multidimensional interactions, further give rise to unique inline semantics between them. For example, see Fig. 1, both *James* and *Emma* are friends of *Victoria* at snapshot 1. The difference lies in the fact that *James* begins dating and chatting with *Victoria*,

*Corresponding authors: cyhu@cupk.edu.cn, weiyiwei@cupk.edu.cn.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 25761–25780

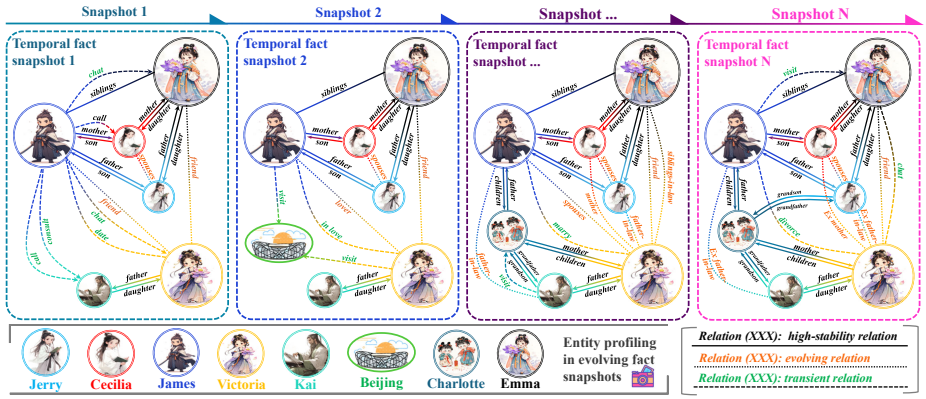

Figure 1: Illustration of Evolving Fact Snapshots. Dashed boxes in different colors indicate different snapshots.

which may influence nature of their "friend" relation. Meanwhile, *Emma* has no other interactions with *Victoria*, so their "friend" relation remains unchanged. Besides, both *Emma* and *Victoria* chat with *James*, but due to the different types of relations each has established with him, the semantics behind act of chatting may vary. The multiple interactions among this knowledge collaboratively create the unique semantics of a specific snapshot. (b) **Trans-snapshot evolutionary synergy.** During snapshot progression, different relations between entities influence each other and exhibit distinct states: **transient**, **high-stability**, or **evolving**. For example, see Fig. 1, *James* and *Emma* chat at snapshot 1 and visit at snapshot N, while consistently maintaining a sibling relation. In contrast, *James* and *Victoria* go on a date at snapshot 1 and fall in love at snapshot 2, while their relation evolves from friends to lovers. Clearly, chat, visit, and date belong to the category of transient relations; sibling belongs to high-stability relations; lover and spouse fall under evolving relations. Each snapshot encapsulates evolving semantics that, in concert, delineate the comprehensive structure of real-world knowledge. The above mechanisms collaboratively depict the underlying logic of temporal facts, highlighting urgent need to construct a high-quality unified framework for modeling.

Our primary goal is to deeply investigate above snapshot mechanisms and develop a framework, TeCES, that supports interpretable and well-justified modeling of evolving snapshots. Specifically, TeCES introduces 2-grade geometric algebra (GA) to map all elements of facts (i.e., entities, relations, and timestamps) into a unified system, leveraging multilevel structures such as scalars, vectors, and bivectors to capture multiple semantics among these elements. TeCES then attaches temporal information to each entity, enabling its mapping to the corresponding snapshot space. Meanwhile, TeCES reconfigures relations and timestamps into two composite GA representations, each carrying temporal and relational information in different proportions, and further applies the geometric product to enable multidimensional interactions between them, revealing relation state transitions driven by snapshot progression. Lastly, TeCES applies geometric product to combine the head entity at each specific snapshot with fused temporal-relational representation, achieving multilevel approximation of the target tail entity. Thus, TeCES can jointly model evolving fact snapshots within a well-founded and lightweight GA system.

- We leverage the concept of snapshots to deconstruct the static logic and temporal dynamic mechanisms inherent in real-world facts.
- We analyze and reconstruct facts via multilevel GA structures to jointly capture their static logic and temporal evolution in a lightweight framework.
- TeCES demonstrates significant and stable performance through a series of experiments.

2 Related Work

2.1 Static KGC models

We categorize static KGC models into four families. **Translation Family.** TransE interprets relations as a translation between head and tail entity to achieve its modeling objectives. Subsequent works in this line, such as TransD (Ji et al., 2015) and TransR (Lin et al., 2015). **Tensor Family.**

Tensor decomposition is effective for capturing latent semantics and logical rules in KGs. TuckER (Balazevic et al., 2019) uses Tucker decomposition to factorize a fact into a core tensor and three matrices. QuatE (Zhang et al., 2019) and DualE (Cao et al., 2021) model relations as rotations in quaternion and dual quaternion spaces. BDRI (Yu et al., 2023a) uses BTD structure and inverse patterns to enhance forward-inverse relation interactions. CompoundE (Ge et al., 2023) integrates multiple 3D compound transformations, enabling diverse variants to align with rich intrinsic characteristics. **Deep learning Family.** Some models using neural networks to accomplish static KGC task gets remarkable results. EIGAT (Zhao et al., 2022), HyperGatE (Fang et al., 2025), and RGAI (Shang et al., 2024a). Recently, using geometric rules to improve deep learning interpretability has gained attention. MGTCA (Shang et al., 2024b) integrates geometric features into a convolutional attention network. **LLMs Family.** LLMs adopt strategies such as prompt engineering, fine-tuning, and retrieval-augmented generation to achieve integrated knowledge representation. MKGL (Guo et al., 2024), KoPA (Zhang et al., 2024c), CSProm (Li et al., 2024a), and KGR3 (Li et al., 2025).

2.2 Temporal KGC models

We divide temporal KGC models into five families. **Translation Family.** TTransE (Jiang et al., 2016) extends TransE by modeling timestamps as translation vectors. Similarly, HYTE (Dasgupta et al., 2018) extends TransH by representing timestamps as a learned hyperplane. **Tensor Family.** TComplEx (Lacroix et al., 2020) and J-MTComplEx (Zhang et al., 2024b) model facts in complex space, while TeLM (Xu et al., 2021) and GeomE (Xu et al., 2023) represent facts within a GA system. EHPR (Guo et al., 2025a) encodes knowledge information as quaternion. MvTuckER (Wang et al., 2024) learns multi-view representations using Tucker decomposition. TuTR (Yue et al., 2025) uses a two-layer Tucker–Tensor Ring decomposition to enhance temporal interaction. TBDRI (Yu et al., 2023b) and CEC-BD (Yue et al., 2024), based on BTD, utilize different ways to represent facts, respectively. TCompoundE (Ying et al., 2024) integrates multiple 3D compound geometric transformation operations to align with rich intrinsic characteristics. TeRDy (Liu and Wang, 2025) learns temporal relaton embeddings via frequency decomposition. **Mathematical rules Family.** CDRGN-SDE (Zhang et al., 2024a) uses stochastic differential equations to integrate temporal and dimensional data. **Deep learning Family.** Some studies incorporate neural networks into temporal KGC, including TeMP-SA (Wu et al., 2020), DLGR (Xiao et al., 2024), SANe (Li et al., 2024d), MDRQS (Zhu et al., 2025), Neo-TKGC (Qiu et al., 2025), and GLARGCN (Wang et al., 2025). Moreover, research on enhancing deep learning interpretability via geometric spaces is emerging. HyGNet (Li et al., 2024c) integrates hyperbolic space into neural networks, enhancing the interpretability of networks. **LLMs Family.** LLMs capture spatiotemporal patterns through task reformulation, architecture, and temporal learning. COSIGN (Li et al., 2024b), GS-KGC (Yang et al., 2025), HFL (Xu et al., 2025), DuaTHP (Chen et al., 2025). These strategies enhance LLMs in dynamic modeling but still lack fine-grained modeling and interpretability.

3 Preliminaries

3.1 Problem Definition

Temporal knowledge graph $\mathcal{KG}$ consists of entities, relations, and timestamps. We define $\mathcal{E}$ as the set of all entities, $\mathcal{R}$ as the set of all relations, and $\mathcal{T}$ as the set of all timestamps. Formally, given a quadruple is represented as $(s, r, o, \tau) \subset \mathcal{E} \times \mathcal{R} \times \mathcal{E} \times \mathcal{T}$, where $s \in \mathcal{E}$ and $o \in \mathcal{E}$ denote head and tail entity respectively, $r \in \mathcal{R}$ denotes the relation between them, and $\tau \in T$ denotes timestamp. To effectively capture temporal semantic structures, we adopt the concept of "snapshots" by aggregating facts with identical timestamps τ_n into unified time slices, constructing structured semantic representations at each time point. Specifically, the snapshot at timestamp τ_n is defined as $\mathcal{KG}_{\tau_n} = (s, r, o, \tau_n) \in \mathcal{KG}$. That is, each snapshot corresponds to a time-specific relational graph. Temporal evolution is then modeled as transitions across successive snapshots. We use Π and $\Pi^- = \mathcal{E} \times \mathcal{R} \times \mathcal{E} \times \mathcal{T} - \Pi$ to denote set of observed quadruples and set of unobserved quadruples, respectively. Timestamp τ has multiple cases, such as time period $[\tau_b, \tau_e]$, missing beginning time period $[-, \tau_e]$, missing ending time period $[\tau_b, -]$ and time point τ.

3.2 Geometric Algebra Background

Geometric algebra (GA) (Hestenes, 1999) unifies linear algebra, vector algebra, complex numbers, quaternions, and exterior algebra into one system with clear geometric meaning and high computa-

tional efficiency.

2-grade GA space $\mathbb{G}^2$ is constructed over two-dimensional (2D) Euclidean vector space $\mathbb{R}^2$, with $\{e_1, e_2\}$ as an orthonormal basis. Elements of $\mathbb{G}^2$ are referred to as 2-grade multivectors $\mathcal{M}$, consisting of scalars (grade-0), vectors (grade-1), and bivectors (grade-2), i.e., $M_\alpha = \alpha_0 + \alpha_1 e_1 + \alpha_2 e_2 + \alpha_{12} e_1 e_2$, as shown in Fig. 2.

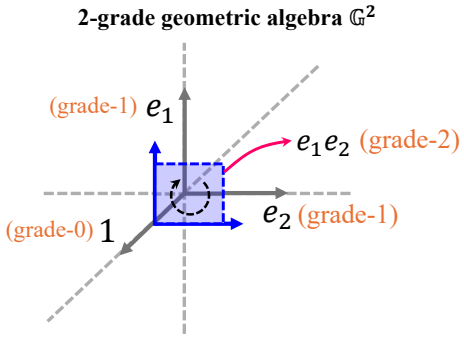

Figure 2: Conceptual diagram of geometric algebra.

Geometric product: The geometric product of two 2-grade multivectors involves the combination of scalar, vector, and bivector components. Given $M_\alpha = \alpha_0 + \alpha_1 e_1 + \alpha_2 e_2 + \alpha_{12} e_1 e_2$ and $M_\beta = \beta_0 + \beta_1 e_1 + \beta_2 e_2 + \beta_{12} e_1 e_2$, their geometric product expands as:

$$\begin{aligned} M_\alpha \times_2 M_\beta = {} & \alpha_0\beta_0 + \alpha_1\beta_1 + \alpha_2\beta_2 - \alpha_{12}\beta_{12} \\ & + \big(\alpha_0\beta_1 + \alpha_1\beta_0 \\ & \quad - \alpha_2\beta_{12} + \alpha_{12}\beta_2\big)e_1 \\ & + \big(\alpha_0\beta_2 + \alpha_1\beta_{12} \\ & \quad + \alpha_2\beta_0 - \alpha_{12}\beta_1\big)e_2 \\ & + \big(\alpha_0\beta_{12} + \alpha_1\beta_2 \\ & \quad - \alpha_2\beta_1 + \alpha_{12}\beta_0\big)e_1e_2 \end{aligned} \tag{1}$$

where $\times_2$ is a geometric product.

Inner product: For two 2-grade multivectors, their inner product reflects the degree of alignment between their geometric components. Given $M_\alpha = \alpha_0 + \alpha_1 e_1 + \alpha_2 e_2 + \alpha_{12} e_1 e_2$ and $M_\beta = \beta_0 + \beta_1 e_1 + \beta_2 e_2 + \beta_{12} e_1 e_2$, their inner product is defined as the scalar part of their geometric product: $M_\alpha \cdot M_\beta := \langle M_\alpha M_\beta \rangle_0 = \alpha_0\beta_0 + \alpha_1\beta_1 + \alpha_2\beta_2 - \alpha_{12}\beta_{12}$, where $\langle\cdot\rangle_0$ denotes the grade-0 projection of a multivector, which extracts scalar part of geometric product.

4 TeCES for temporal KGC

This section formally defines the notation and presents the implementation details of TeCES, with Fig. 3 illustrating its overall architecture.

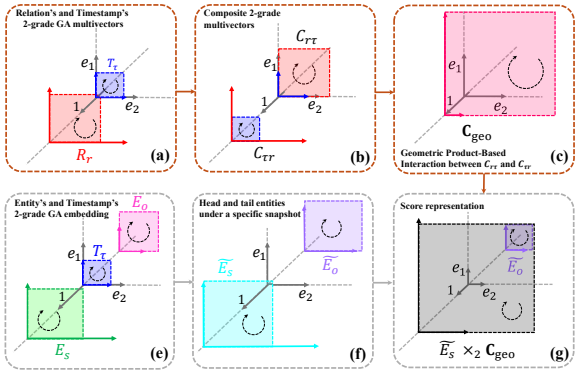

Figure 3: Illustrations of TeCES. Arrows of different colors indicate the process flows of different operations.

Symbol description. Suppose that $\mathcal{KG}$ consists of $\mathscr{E}$ entities, $\mathscr{R}$ relations, and $\mathscr{T}$ timestamps. We use $E \in \mathbb{G}^{2^{\mathscr{E}\times k}}$ to denote all entity 2-grade multivector embeddings, $R \in \mathbb{G}^{2^{\mathscr{R}\times k}}$ to denote all relation 2-grade multivector embeddings, $T \in \mathbb{G}^{2^{\mathscr{T}\times k}}$ to denote all timestamp 2-grade multivector embeddings, where each row is a k-dimensional embedding vector. Given a quadruple (s, r, o, τ), the head entity s, relation r, tail entity o, and timestamp τ are embedded as 2-grade multivectors in $\mathbb{G}^2$:

$$\begin{aligned} E_s &= s_0 + s_1 e_1 + s_2 e_2 + s_{12} e_1 e_2, \\ R_r &= r_0 + r_1 e_1 + r_2 e_2 + r_{12} e_1 e_2, \\ E_o &= o_0 + o_1 e_1 + o_2 e_2 + o_{12} e_1 e_2, \\ T_\tau &= \tau_0 + \tau_1 e_1 + \tau_2 e_2 + \tau_{12} e_1 e_2, \end{aligned} \tag{2}$$

where $s_0, s_1, s_2, s_{12} \in \mathbb{R}^k$; $r_0, r_1, r_2, r_{12} \in \mathbb{R}^k$; $o_0, o_1, o_2, o_{12} \in \mathbb{R}^k$; $\tau_0, \tau_1, \tau_2, \tau_{12} \in \mathbb{R}^k$.

Implementation details. TeCES embeds entities, relations, and timestamps into a 2-grade GA system, thereby leveraging multilevel structures to capture multiple semantics among these elements (see Fig. 3(a) and (e)). To ensure that entities can be stably mapped to different specific snapshots, TeCES attaches each grade component of timestamp's 2-grade multivector T_τ to corresponding components of head entity's and tail entity's 2-grade multivectors, E_s and E_o, respectively (see Fig. 3(f)):

$$\begin{aligned} \widetilde{E}_s = {} & f_0(s_0, \tau_0) + f_1(s_1, \tau_1)e_1 \\ & + f_2(s_2, \tau_2)e_2 + f_{12}(s_{12}, \tau_{12})e_1e_2, \\ \widetilde{E}_o = {} & g_0(o_0, \tau_0) + g_1(o_1, \tau_1)e_1 \\ & + g_2(o_2, \tau_2)e_2 + g_{12}(o_{12}, \tau_{12})e_1e_2. \end{aligned} \tag{3}$$

Here, $\widetilde{E}_s$ and $\widetilde{E}_o$ respectively represent the head and tail entities under a specific snapshot space;

$f_i(\cdot,\cdot)$ and $g_i(\cdot,\cdot)$ denote additive fusion operations applied to the head and tail entities with the timestamp, respectively, ensuring balanced integration and traceability of entity semantics within snapshots. Furthermore, to enable multidimensional and deep interactions between temporal and relational information, and to uncover relation states driven by snapshot progression, TeCES reconfigures the relation's 2-grade multivector $R_r = r_0 + r_1e_1 + r_2e_2 + r_{12}e_1e_2$ and the timestamp's 2-grade multivector $T_\tau = \tau_0+\tau_1e_1+\tau_2e_2+\tau_{12}e_1e_2$ into two composite 2-grade multivectors, namely $C_{r\tau} = r_0 + r_1e_1 + r_2e_2 + \tau_{12}e_1e_2$ and $C_{\tau r} = \tau_0 + \tau_1e_1 + \tau_2e_2 + r_{12}e_1e_2$. TeCES then uses geometric product $\times_2$ to realize interaction between $C_{r\tau}$ and $C_{\tau r}$ (see Fig. 3(b) and (c)):

$$\mathbf{C}_{\text{geo}} = C_{r\tau} \times_2 C_{\tau r}. \tag{4}$$

Finally, TeCES performs a geometric product $\times_2$ to interact $\widetilde{E}_s$ with $\mathbf{C}_{\text{geo}}$, and further applies an inner product with $\widetilde{E}_o$ to obtain final score representation of TeCES (see Fig. 3(g)):

$$\varphi(s,r,o,\tau) = (\widetilde{E}_s \times_2 \mathbf{C}_{\text{geo}}) \cdot \widetilde{E}_o, \tag{5}$$

where $\cdot$ is inner product. TeCES handles different forms of timestamps as follows: 1) For full timestamps (s, r, o, $[\tau_b, \tau_e]$), TeCES splits them into two single-timestamp facts and averages their scores:

$$\varphi(s,r,o,[\tau_b,\tau_e]) = \frac{\varphi(s,r,o,\tau_b) + \varphi(s,r,o,\tau_e)}{2}. \tag{6}$$

2) For missing timestamps, i.e., (s, r, o, $[\tau_b, -]$) or (s, r, o, $[-, \tau_e]$), the score of TeCES is taken directly from known timestamp:

$$\begin{aligned} \varphi(s,r,o,[\tau_b,-]) &= \varphi(s,r,o,\tau_b), \\ \varphi(s,r,o,[-,\tau_e]) &= \varphi(s,r,o,\tau_e). \end{aligned} \tag{7}$$

Optimization. TKGs can be regarded as fourth-order tensors; thus, we adopt tensor nuclear norm regularization and design the following linear temporal regularizer to improve quality and continuity of representations across adjacent timestamps:

$$\mathfrak{L}_a = \sum_{i=1}^{N} \|T_{\tau_{i+1}} - T_{\tau_i} - T_b\|_4^4, \tag{8}$$

where $\|\cdot\|_4^4$ denotes nuclear 4-norm. T_b is a linear time-constrained bias. This constraint encourages nearby timestamps to be close while allowing distant ones to differ, thereby promoting temporal smoothness.

Loss function. We regard temporal KGC as a multi-classification task, use cross-entropy loss function:

$$\begin{aligned} L = \sum_{(s,r,o,\tau)\in\Pi\cup\Pi'} (-log\frac{exp\,(\varphi(s,r,o,\tau))}{\sum_{s'\in\Pi'} exp\,(\varphi(s',r,o,\tau))} \\ - log\frac{exp\,(\varphi(s,r,o,\tau))}{\sum_{o'\in\Pi'} exp\,(\varphi(s,r,o',\tau))}) + \lambda_a\mathfrak{L}_a. \end{aligned} \tag{9}$$

Here Π' is sampled from set of unobserved quadruple Π^-. λ_a is temporal regularization weight.

5 Experiments and Results

5.1 Datasets and Experimental Setup

Datasets. ICEWS14, ICEWS05-15 and ICEWS18 are subsets of ICEWS (Boschee et al., 2015) collecting large-scale event data from multiple sources. ICEWS14, ICEWS05-15 and ICEWS18 cover events during *2014*, *2005–2015*, and *2018*, respectively. YAGO11k and Wikidata12k (Dasgupta et al., 2018) are subsets of YAGO3 and Wikidata. YAGO3 (Mahdisoltani et al., 2015) and Wikidata (Erxleben et al., 2014) are TKGs with time annotations in forms such as time points and intervals. GDELT (Trivedi et al., 2017) is a global database of political, economic, and environmental events from *2015-2016*. YAGO15K extends FB15k (Bordes et al., 2013) by adding timestamps to each fact, with incomplete temporal information increasing dataset's complexity. Besides, TeCES is successfully deployed on Enterprise Procurement Dataset (EPD) to evaluate its engineering applicability in real-world scenarios. The datasets and experimental details see Appendices A and F.

Experimental Setup. We implement TeCES in PyTorch and train it with Adagrad using a batch size of 1000. The optimal embedding dimension across all benchmarks is k = 100. The settings of the temporal regularization weight λ_a and the learning rate η, as well as corresponding hyperparameter sensitivity analysis, are provided in Appendix B.

5.2 Evaluation Protocol and Baselines

1) **Evaluation Metrics**. **Mean Reciprocal Rank (MRR)** measures the average inverse rank of correct answers. **Hits@n** is the proportion of correct answers ranked in top **n** (where $\mathbf{n} \in 1, 3, 10$).
2) **Selecting SOTA baselines.** We select SOTA

models as follows: TComplEx (2020), TNTComplEx (2020), TeLM (2021), TBDRI (2023), TGeomE++ (2023), DLGR (2024), CEC-BD (2024), J-MTComplEx (2024), SANe (2024), TCompoundE (2024), HFL (2025), DuaTHP (2025), EHPR (2025), GLARGCN (2025), MDRQS (2025), LTGQ (2025), TuTR (2025), TeRDy (2025).

5.3 Main Results

Tables 1, 2 and 3 list temporal KGC results on each dataset. All results are taken from original papers. **Dashes** indicate results not reported in the corresponding literature. The best results among all models are shown in **bold**, and the second-best results are underlined. Overall, TeCES outperforms all baseline models across datasets. Compared with EHPR (**tensor family**), TeCES improves MRR on all datasets with an average increase of 17.7 points. Compared with MDRQS (**deep learning family**), TeCES improves MRR on all datasets with an average increase of 32.2 points. Compared with DuaTHP (**LLMs family**), TeCES improves MRR on all datasets with an average increase of 37.8 points. For detailed information, see Appendix C. Compared with existing efficient models (e.g., EHPR and TuTR), TeCES's core advantage lies in its capability to simulate the temporal snapshot evolution mechanism of real world within a GA framework. This design enables TeCES to capture the diverse and dynamic characteristics of entities and relations both within and across snapshots, while adaptively accommodating relational pattern variations arising from differences inside and outside snapshots. Moreover, TeCES, with its lightweight GA system and simulated snapshot design, compensates for the shortcomings of deep learning models and LLMs in terms of interpretability and computational efficiency, laying a solid foundation for achieving interpretability and lightweight implementation of LLMs in specific knowledge domains. Finally, we apply **Friedman test** for overall comparison and **Wilcoxon signed-rank test** for pairwise significance analysis of MRR between TeCES and the baseline models on ICEWS14 and ICEWS05-15, demonstrating the statistical significance of TeCES. See Appendix B for details.

Fig. 4 shows horizontal and vertical distribution patterns across datasets, where different colors denote distinct datasets. Due to the relatively uniform fact densities and differing temporal ranges of ICEWS18, ICEWS05-15, and GDELT, we adopt temporal averaging to capture general

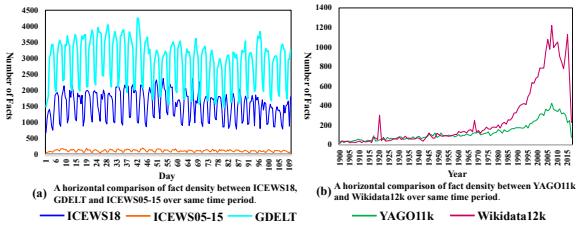

Figure 4: Fact distribution comparison visualization across similar datasets.

trends and reveal structural differences in data density (see Fig. 4(a)). Within the ICEWS series, TeCES shows substantial gains on ICEWS05-15, ICEWS18, and GDELT compared to SOTA models. Despite similar data types, ICEWS14, ICEWS05-15 and ICEWS18 differ in density and distribution. ICEWS05-15 spans over ten times the temporal range of ICEWS14, increasing dependence on cross-snapshot modeling, where TeCES performs stably. In contrast, ICEWS18 shares a similar time range with ICEWS14 but features denser intra-snapshot information, increasing interaction complexity, highlighting TeCES's robustness. GDELT further intensifies this trend with even higher density, reinforcing model's stability in dense settings. Moreover, we observe that on high-density datasets (e.g., GDELT and ICEWS18), TeCES exhibits a slight decrease on Hit@10 compared to EHPR. This phenomenon stems from TeCES's emphasis on modeling multi-faceted features of high-frequency entity–relation interactions in temporal snapshots, which improves prediction accuracy for frequent cases but partially overlooks low-frequency entities.

For Wikidata12k and YAGO11k, TeCES outperforms SOTA models significantly. Both datasets exhibit long-tail distributions, leading to fluctuating interaction loads across snapshots. TeCES effectively balances this heterogeneity. In particular, Wikidata12k has a more concentrated head-entity density than YAGO11k, underscoring TeCES's adaptability to load imbalance (see Fig. 4(b)). Finally, YAGO15K contains both static and temporal facts, where relation states evolve dynamically over time, making it more reflective of real-world knowledge evolution. As shown in Fig. 5, different relations exhibit diverse temporal distribution patterns (e.g., high-frequency, low-frequency, and transient relations), demonstrating strong dynamics and structural complexity, which significantly increases the difficulty of temporal KGC.

	ICEWS14				ICEWS05-15				ICEWS18			
	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
TeLM	62.5	54.5	67.3	77.4	67.8	59.9	72.8	82.3	-	-	-	-
TBDRI	65.2	55.2	69.7	78.5	70.9	64.6	75.7	82.1	-	-	-	-
TGeomE++	65.2	55.2	69.7	78.5	70.9	64.6	75.7	82.1	-	-	-	-
DLGR	46.7	36.6	51.6	-	-	-	-	-	35.4	25.1	40.0	-
CEC-BD	63.3	55.4	68.0	77.7	68.1	60.2	73.0	82.5	28.5	18.8	32.3	47.7
J-MTComplEx	63.6	55.6	68.1	78.1	68.3	60.1	73.6	83.2	-	-	-	-
SANe	63.8	55.8	68.8	78.2	68.3	60.5	73.4	82.3	-	-	-	-
HFL	37.9	27.7	42.7	57.3	38.7	28.5	43.0	59.5	27.1	17.8	30.4	45.5
DuaTHP	63.7	55.3	67.4	78.8	68.9	61.8	72.6	83.5	-	-	-	-
EHPR	71.5	65.0	75.6	83.7	76.0	69.5	80.3	87.7	46.2	40.1	47.5	**55.3**
GLARGCN	67.6	62.1	71.0	77.6	77.5	72.7	80.8	85.9	-	-	-	-
LTGQ	64.1	56.4	68.9	77.9	69.4	61.4	74.6	83.7	-	-	-	-
MDRQS	62.5	54.4	67.3	77.5	67.0	59.1	71.6	81.5	-	-	-	-
TCompoundE	64.4	56.1	68.2	78.2	69.2	61.2	74.3	83.7	-	-	-	-
TeRDy	64.8	56.6	69.7	79.9	69.7	61.4	74.9	84.7	-	-	-	-
TuTR	73.2	69.8	74.5	79.7	79.7	75.7	82.1	87.1	44.7	39.3	46.7	54.4
TeCES (ours)	**90.7**	**89.3**	**91.4**	**93.3**	**93.2**	**92.5**	**93.6**	**94.5**	**49.2**	**49.0**	**49.3**	49.7
APG (%)	17.5 ↑	19.5 ↑	15.8 ↑	9.6 ↑	13.5 ↑	16.8 ↑	11.5 ↑	6.8 ↑	3.0 ↑	8.9 ↑	1.8 ↑	5.6 ↓

Table 1: Experimental results on ICEWS14, ICEWS05-15 and ICEWS18. ↑ and ↓ denote performance gain and drop relative to baseline model, respectively.

	YAGO11k				Wikidata12k				GDELT			
	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
TComplEx	18.5	12.7	18.3	30.7	33.1	23.3	35.7	53.9	29.8	21.3	32.3	46.4
TeLM	19.1	12.9	19.4	32.1	33.2	23.1	36.0	54.2	-	-	-	-
CEC-BD	21.2	15.4	21.5	33.9	33.9	24.1	36.9	54.3	29.6	20.1	33.4	46.5
J-MTComplEx	22.2	15.8	22.9	35.6	36.0	27.0	39.3	54.1	-	-	-	-
SANe	25.0	18.0	26.6	40.1	43.2	33.1	48.3	64.0	30.1	21.2	32.6	47.6
DuaTHP	19.5	12.3	20.7	32.9	30.4	20.9	33.1	50.9	27.5	18.3	30.9	46.2
EHPR	34.6	22.7	40.0	58.4	43.9	33.1	47.7	67.4	76.5	71.1	79.0	**87.0**
GLARGCN	-	-	-	-	37.3	33.8	38.5	42.3	57.1	48.4	61.3	73.0
MDRQS	27.1	19.9	29.6	35.6	-	-	-	-	42.4	35.0	45.1	56.3
TCompoundE	-	-	-	-	-	-	-	-	43.3	34.7	46.9	59.5
LTGQ	25.1	18.5	26.3	39.3	41.3	30.9	46.5	62.2	-	-	-	-
TuTR	21.0	15.4	21.7	32.4	46.0	40.5	47.3	56.8	36.5	28.7	39.6	51.7
TeCES (ours)	**61.6**	**58.3**	**63.0**	**67.7**	**73.6**	**70.8**	**75.1**	**78.7**	**79.7**	**77.9**	**80.4**	83.1
APG (%)	27.0 ↑	35.6 ↑	33.0 ↑	9.3 ↑	27.6 ↑	30.3 ↑	27.4 ↑	11.3 ↑	3.2 ↑	6.8 ↑	1.4 ↑	3.9 ↓

Table 2: Experimental results on YAGO11k, Wikidata12k and GDELT. ↑ and ↓ denote performance gain and drop relative to baseline model, respectively.

	YAGO15K			
	MRR	H@1	H@3	H@10
TNTComplEx	37.0	29.0	39.0	54.0
TBDRI	36.8	30.1	37.2	55.4
TuckER-FA	36.5	28.2	39.2	54.3
EHPR	54.1	46.4	58.3	68.5
TeCES (ours)	**84.9**	**82.8**	**86.3**	**88.6**
APG (%)	30.8 ↑	36.4 ↑	28.0 ↑	20.1 ↑

Table 3: Experimental results on YAGO15K. ↑ and ↓ denote performance gain and drop relative to baseline model, respectively.

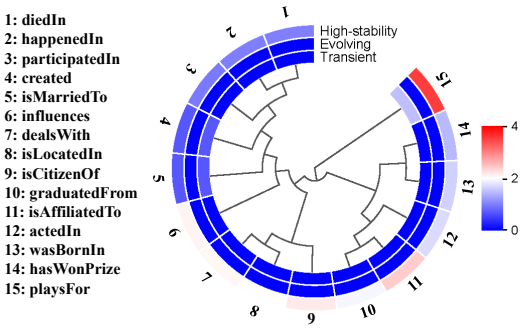

Figure 5: Visualization of high-frequency relation state distribution on YAGO15K.

Experimental results show that, despite the increased challenge posed by YAGO15K, TeCES still achieves more competitive performance compared to SOTA models (e.g., EHPR). This indicates that, by jointly modeling relations and temporal information, TeCES is capable of effectively capturing the dynamic evolution of relation states, improving performance in complex temporal KGC scenarios.

6 ANALYSIS

Fig. 6 lists performance of different component configurations across four datasets, highlighting the role of each module. **TeCES w/o TA**: Removes Timestamp Attachment (TA), so temporal information is no longer embedded in entity representations for snapshot-specific mapping. **TeCES w/o RTF**: Removes Relation-Time Fusion (RTF), so relations and timestamps are no longer reconfigured.

Base-GA: A basic GA model with RTF and TA excluded. We further adapt our framework to complex space to evaluate its portability and robustness. **CEFS**: Fully adapts TeCES framework to complex space. **CEFS w/o TA**: Removes TA from CEFS. **CEFS w/o RTF**: Removes RTF from CEFS. See Appendix G for details.

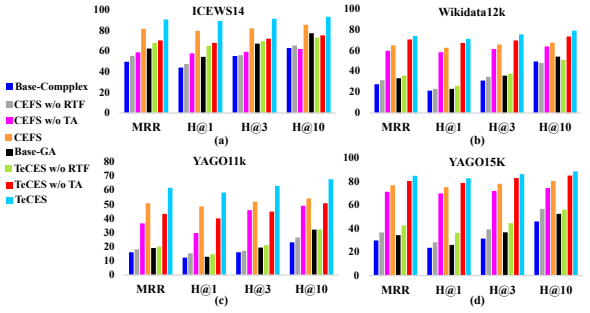

Figure 6: Component compatibility experiments across different datasets.

6.1 Ablation study on TA and RTF

TeCES and CEFS consistently get significant performance across all datasets, demonstrating portability of the framework. Furthermore, RTF consistently outperforms TA, mainly because TA maps entities into different snapshot spaces to support RTF in capturing intra-snapshot semantic coupling and relation states. In contrast, the long-tail effect in Wikidata12k and YAGO11k, along with incomplete temporal data in YAGO15K, limits TA's standalone modeling ability. Still, the joint modeling of TA and RTF consistently yields stable performance gains, highlighting their strong complementarity. Overall, the collaboration between TA and RTF leads to performance gains across all datasets. Besides, the TeCES series consistently outperforms the CEFS series, highlighting superior expressiveness of GA system over complex number system.

To understand TA and RTF, we visualize the trained fact embeddings. First, we extract and visualize all head and tail entity embeddings of *Obama* and *Merkel* with temporal information from Jan to Jul 2014 (see Fig. 7). We observe that *Obama*'s and *Merkel*'s embeddings across different months exhibit similar distribution patterns, which aligns with the motivation behind designing TA. Next, we extract facts formed by high-frequency relations between *North Korea* and *South Korea* from Jan to Mar 2014, and visualize the fused embeddings of head entity, relation, and timestamp learned by

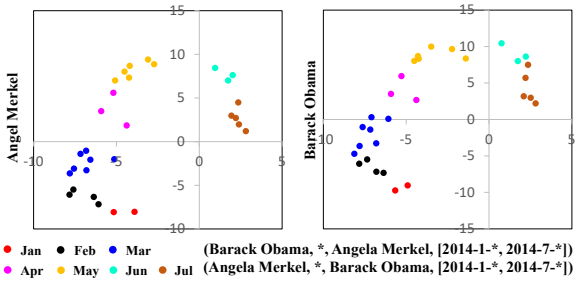

Figure 7: Visualization of learned head and tail entity embeddings carrying specific temporal information. Different colors represent entities from different months.

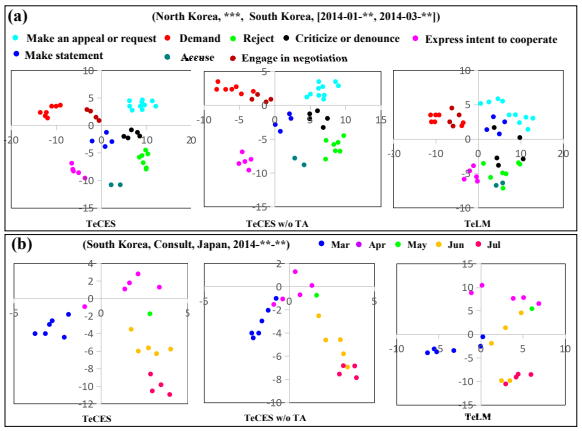

Figure 8: Visualization of fact embeddings learned on ICEWS14. (a) Fused head entity–relation–timestamp embeddings; (b) Temporal-relation embeddings across months.

different models (see Fig. 8 (a)). Visually, TeCES gets more effective embedding aggregation compared to both TeLM and TeCES w/o TA. Notably, TeCES w/o TA also outperforms TeLM in this regard. Lastly, we extract relation "consult" between *South Korea* and *Japan* from Mar to Jul and visualize learned temporal-relation embeddings (see Fig. 8 (b)). Visually, TeCES and TeCES w/o TA exhibit more pronounced aggregation effects compared to TeLM. Moreover, TeCES demonstrates clearer boundaries across different snapshots compared to TeCES w/o TA. These findings underscore the synergistic performance gains brought by the integration of TA and RTF. To further analyze the above phenomenon, we visualize the fact interactions of high-frequency entities in ICEWS14 during January, May, and November (see Fig. 9). Different colored frames denote different snapshots; nodes represent high-frequency entities, and edges indicate fact interactions between entities within the corresponding temporal snapshot. The interactions and evolving relational states of entities across dif-

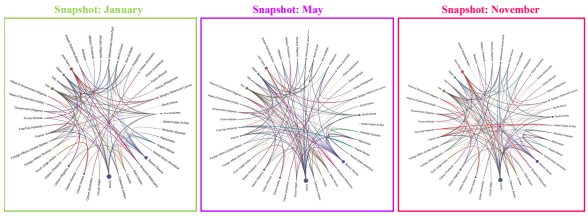

Figure 9: Snapshot evolution distribution of facts between high-frequency entities from ICEWS14.

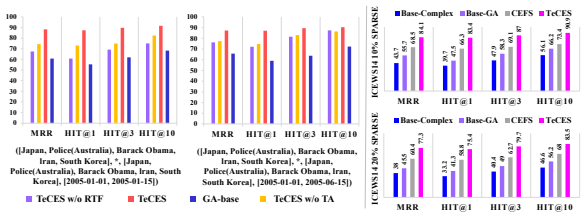

Figure 10: Strong constraints and sparse stress tests.

ferent snapshots strongly align with the motivation behind TA and RTF.

6.2 Strong constraints analysis of quadruples

We construct two separate strong-constraint datasets to simulate distinct scenarios: specific snapshots and evolving snapshots (see Fig. 10). We first extract facts involving *Japan*, *Police (Australia)*, *Obama*, *Iran*, and *South Korea* from Jan 1 to Jan 15, 2005, to simulate a specific snapshot scenario. We further extract facts involving same set of entities from Jan 1 to Jun 15, 2005, to simulate an evolving snapshot scenario. Empirically, TeCES consistently gets superior performance. Furthermore, TeCES w/o TA shows a larger performance improvement over TeCES w/o RTF in the specific snapshot scenario. This is because, under specific snapshots, TA maps all facts into the same space, which limits performance. In contrast, under the evolving snapshot scenario, TA and RTF work collaboratively to yield performance gains. We randomly remove 10% and 20% of ICEWS14 training set (denoted as ICEWS14 10% SPARSE and ICEWS14 20% SPARSE, respectively) to evaluate the robustness of TeCES under sparse data conditions (see Fig. 10). Empirically, the TeCES series models remain highly competitive in these sparse scenarios, validating the stability of GA system.

6.3 Training Time and Standard Deviation

Fig. 11 (a) lists per-epoch training time on NVIDIA RTX 3090 for TeLM, TeCES w/o TA, TeCES w/o RTF, and TeCES across multiple datasets. Specifically, compared to efficient models (e.g., TeLM), TeCES offers controllable computational overhead and greater competitiveness. Fig. 11 (b) shows the MRR standard deviations over 5 runs of TeCES on all datasets, highlighting its performance stability.

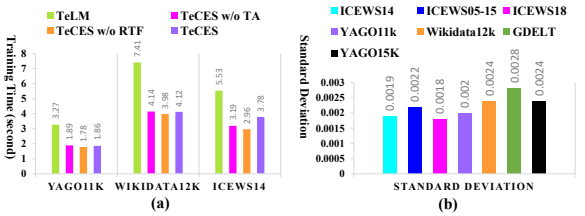

Figure 11: Efficiency and Stability Evaluation: (a) Training Time; (b) Standard Deviation

6.4 Conclusion

This paper uses snapshots to reveal the underlying mechanisms of real-world facts. Based on this, we design TA and RTF with GA to jointly model evolving snapshots. Visualization is used to simulate and validate TeCES in different scenarios. We also observe severe fact-density imbalance across snapshots, which makes the joint modeling of TA and RTF more challenging. In future work, we will study snapshot mechanisms under imbalance and develop specialized models.

Limitations

This paper leverages snapshots to deconstruct the underlying mechanisms of real-world facts. Based on this, we design TA and RTF modules using geometric algebra to jointly model the evolving snapshots. Furthermore, we observe that the fact density across snapshots is highly uneven, leading to significant variations in entity and relation interactions under different snapshots. Clearly, this objective challenge limits TeCES's ability to achieve more fine-grained knowledge representation. To further address the issue of imbalanced fact density across snapshots, we plan to introduce the concept of live snapshots and design a framework capable of dynamically integrating knowledge from adjacent snapshots to enhance KGC.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (No. 6250010550), the Natural Science Foundation of Xinjiang Uygur Autonomous Region (No. 2024D01B99), and the Introduction Program for Young Doctors (Tianchi Talents) of Xinjiang Uygur Autonomous Region.

References

Ivana Balazevic, Carl Allen, and Timothy M. Hospedales. 2019. Tucker: Tensor factorization for knowledge graph completion. 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 2019, Hong Kong, China, November 3-7, 2019*, pages 5184–5193. Association for Computational Linguistics.

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

Elizabeth Boschee, Jennifer Lautenschlager, Sean O'Brien, Steve Shellman, James Starz, and Michael Ward. 2015. Icews coded event data. *Harvard Dataverse*, 12.

Zongsheng Cao, Qianqian Xu, Zhiyong Yang, Xiaochun Cao, and Qingming Huang. 2021. Dual quaternion knowledge graph embeddings. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 6894–6902. AAAI Press.

K. Mani Chandy and Leslie Lamport. 1985. Distributed snapshots: Determining global states of distributed systems. *ACM Transactions on Computer Systems*, 3(1):63–75.

Yutong Chen, Xia Li, Yang Liu, and Tiangui Hu. 2025. Integrating transformer architecture and householder transformations for enhanced temporal knowledge graph embedding in duathp. *Symmetry*, 17(2):173.

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 (EMNLP 2018)*, pages 2001–2011.

Fredo Erxleben, Michael Günther, Markus Krötzsch, Julian Mendez, and Denny Vrandecic. 2014. Introducing wikidata to the linked data web. In *The Semantic Web – ISWC 2014: 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014, Proceedings, Part I*, Lecture Notes in Computer Science (LNCS), vol 8796, pages 50–65. Springer.

Yan Fang, Xiaodong Liu, Wei Lu, Witold Pedrycz, Qi Lang, and Jianhua Yang. 2025. Knowledge graph completion with low-dimensional gated hierarchical hyperbolic embedding. *Knowledge-Based Systems*, 309:112804.

Xiou Ge, Yun-Cheng Wang, Bin Wang, and C.-C. Jay Kuo. 2023. Compounding geometric operations for knowledge graph completion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 6947–6965. Association for Computational Linguistics.

Jiujiang Guo, Mankun Zhao, Jian Yu, Ruiguo Yu, Jianhang Song, Qifei Wang, Linying Xu, and Mei Yu. 2025a. Learning evolutionary hierarchy perception representation based on quaternion for temporal knowledge graph completion. *Information Sciences*, 688:121409.

Jiujiang Guo, Mankun Zhao, Wenbin Zhang, Tianyi Xu, Linying Xu, Yu Jian, Yu Mei, and Yu Ruiguo. 2025b. TeDS: Joint learning of diachronic and synchronic perspectives in quaternion space for temporal knowledge graph completion. In *Forty-second International Conference on Machine Learning*.

Lingbing Guo, Zhongpu Bo, Zhuo Chen, Yichi Zhang, Jiaoyan Chen, Yarong Lan, Mengshu Sun, Zhiqiang Zhang, Yangyifei Luo, Qian Li, Qiang Zhang, Wen Zhang, and Huajun Chen. 2024. MKGL: mastery of a three-word language. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

David Hestenes. 1999. *New Foundations for Classical Mechanics*. Kluwer Academic Publishers.

Guoliang Ji, Shizhu He, Liheng Xu, Kang Liu, and Jun Zhao. 2015. Knowledge graph embedding via dynamic mapping matrix. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 687–696. The Association for Computer Linguistics.

Tingsong Jiang, Tianyu Liu, Tao Ge, Lei Sha, Baobao Chang, Sujian Li, and Zhifang Sui. 2016. Towards time-aware knowledge graph completion. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING 2016)*, pages 1715–1724, Osaka, Japan.

Timothée Lacroix, Guillaume Obozinski, and Nicolas Usunier. 2020. Tensor decompositions for temporal knowledge base completion. In *8th International Conference on Learning Representations*. OpenReview.net.

Dawei Li, Zhen Tan, Tianlong Chen, and Huan Liu. 2024a. Contextualization distillation from large language model for knowledge graph completion. In *Findings of the Association for Computational Linguistics: EACL 2024, St. Julian's, Malta, March 17-22, 2024*, pages 458–477. Association for Computational Linguistics.

Jinpeng Li, Hang Yu, Xiangfeng Luo, and Qian Liu. 2024b. COSIGN: contextual facts guided generation for knowledge graph completion. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1669–1682. ACL.

Muzhi Li, Cehao Yang, Chengjin Xu, Xuhui Jiang, Yiyan Qi, Jian Guo, Ho fung Leung, and Irwin King. 2025. Retrieval, reasoning, re-ranking: A context-enriched framework for knowledge graph completion. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2025 – Long Papers)*, pages 4349–4363.

Yancong Li, Xiaoming Zhang, Ying Cui, and Shuai Ma. 2024c. Hyperbolic graph neural network for temporal knowledge graph completion. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy*, pages 8474–8486. ELRA and ICCL.

Yancong Li, Xiaoming Zhang, Bo Zhang, Feiran Huang, Xiaopeng Chen, Ming Lu, and Shuai Ma. 2024d. Sane: Space adaptation network for temporal knowledge graph completion. *Information Sciences*, 667:120430.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI 2015)*.

Jingping Liu, Mingchuan Zhang, Weichen Li, Chao Wang, Shuang Li, Haiyun Jiang, Sihang Jiang, Yanghua Xiao, and Yunwen Chen. 2024. Beyond entities: A large-scale multi-modal knowledge graph with triplet fact grounding. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 18653–18661. AAAI Press.

Ziyang Liu and Chaokun Wang. 2025. Terdy: Temporal relation dynamics through frequency decomposition for temporal knowledge graph completion. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 9611–9622. Association for Computational Linguistics.

Farzaneh Mahdisoltani, Joanna Biega, and Fabian M. Suchanek. 2015. YAGO3: a knowledge base from multilingual wikipedias. In *Seventh Biennial Conference on Innovative Data Systems Research (CIDR 2015)*, pages 18–26, Asilomar, California, USA. CIDR.

Zihan Qiu, Xiaoling Zhou, Chunyan An, Qiang Yang, and Zhixu Li. 2025. Neo-tkgc: Enhancing temporal knowledge graph completion with integrated node weights and future information. In *Proceedings of the 18th ACM International Conference on Web Search and Data Mining (WSDM)*, pages 373–381. ACM.

Bin Shang, Yinliang Zhao, and Jun Liu. 2024a. Knowledge graph representation learning with relation-guided aggregation and interaction. *Information Processing and Management*, 61(4):103752.

Bin Shang, Yinliang Zhao, Jun Liu, and Di Wang. 2024b. Mixed geometry message and trainable convolutional attention network for knowledge graph completion. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 8966–8974. AAAI Press.

Rakshit Trivedi, Hanjun Dai, Yichen Wang, and Le Song. 2017. Know-evolve: Deep temporal reasoning for dynamic knowledge graphs. In *Proceedings of the 34th International Conference on Machine Learning (ICML 2017)*, pages 3462–3471, Sydney, Australia. PMLR.

Hao Wang, Jing Yang, Laurence T. Yang, Yuan Gao, Jihong Ding, Xiaokang Zhou, and Huazhong Liu. 2024. Mvtucker: Multi-view knowledge graphs representation learning based on tensor tucker model. *Information Fusion*, 106:102249.

Ruijie Wang, Yuchen Yan, Jialu Wang, Yuting Jia, Ye Zhang, Weinan Zhang, and Xinbing Wang. 2018. Acekg: A large-scale knowledge graph for academic data mining. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, pages 1487–1490. ACM.

Shuo Wang, Shuxu Chen, and Zhaoqian Zhong. 2025. Global and local information-aware relational graph convolutional network for temporal knowledge graph completion. *Applied Intelligence*, 55(2):125.

Yiwei Wei, Shaozu Yuan, Hengyang Zhou, Longbiao Wang, Zhiling Yan, Ruosong Yang, and Meng Chen. 2024. G^2sam: Graph-based global semantic awareness method for multimodal sarcasm detection. In *Thirty-Eighth AAAI Conference on Artificial Intelligence*, pages 9151–9159. AAAI Press.

Jiapeng Wu, Meng Cao, Jackie Chi Kit Cheung, and William L. Hamilton. 2020. Temp: Temporal message passing for temporal knowledge graph completion. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5730–5746. Association for Computational Linguistics (ACL).

Yao Xiao, Guangyou Zhou, Zhiwen Xie, Jin Liu, and Jimmy Xiangji Huang. 2024. Learning dual disentangled representation with self-supervision for temporal knowledge graph reasoning. *Information Processing and Management*, 61(3):103618.

Chengjin Xu, Yung-Yu Chen, Mojtaba Nayyeri, and Jens Lehmann. 2021. Temporal knowledge graph completion using a linear temporal regularizer and multivector embeddings. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2021)*, pages 2569–2578.

Chengjin Xu, Mojtaba Nayyeri, Yung-Yu Chen, and Jens Lehmann. 2023. Geometric algebra based embeddings for static and temporal knowledge graph completion. *IEEE Transactions on Knowledge and Data Engineering*, 35(5):4838–4851.

Wenjie Xu, Ben Liu, Miao Peng, Zihao Jiang, Xu Jia, Kai Liu, Lei Liu, and Min Peng. 2025. Historical facts learning from long-short terms with language model for temporal knowledge graph reasoning. *Information Processing Management*, 62(3):104047.

Rui Yang, Jiahao Zhu, Jianping Man, Hongze Liu, Li Fang, and Yi Zhou. 2025. GS-KGC: A generative subgraph-based framework for knowledge graph completion with large language models. *Information Fusion*, 117:102868.

Rui Ying, Mengting Hu, Jianfeng Wu, Yalan Xie, Xiaoyi Liu, Zhunheng Wang, Ming Jiang, Hang Gao, Linlin Zhang, and Renhong Cheng. 2024. Simple but effective compound geometric operations for temporal knowledge graph completion. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 11074–11086. Association for Computational Linguistics.

Mei Yu, Jiujiang Guo, Jian Yu, Tianyi Xu, Mankun Zhao, Hongwei Liu, Xuewei Li, and Ruiguo Yu. 2023a. BDRI: block decomposition based on relational interaction for knowledge graph completion. *Data Mining and Knowledge Discovery*, 37(2):767–787.

Mei Yu, Jiujiang Guo, Jian Yu, Tianyi Xu, Mankun Zhao, Hongwei Liu, Xuewei Li, and Ruiguo Yu. 2023b. Tbdri: Block decomposition based on relational interaction for temporal knowledge graph completion. *Applied Intelligence*, 53(5):5072–5084.

Shaozu Yuan, Yiwei Wei, Hengyang Zhou, Qinfu Xu, Meng Chen, and Xiaodong He. 2025. Enhancing semantic awareness by sentimental constraint with automatic outlier masking for multimodal sarcasm detection. *IEEE Transactions on Multimedia*, 27:5376–5386.

Lupeng Yue, Yongjian Ren, Yan Zeng, Jilin Zhang, Kaisheng Zeng, Jian Wan, and Mingyao Zhou. 2024. Complex expressional characterizations learning based on block decomposition for temporal knowledge graph completion. *Knowledge-Based Systems*, 290:111591.

Lupeng Yue, Yongjian Ren, Yan Zeng, Jilin Zhang, Kaisheng Zeng, Mingyao Zhou, and Jian Wan. 2025. Two-layer tensor decomposition for temporal kge graph completion. *Expert Systems with Applications*, 287:127953.

Dong Zhang, Wenlong Feng, Zonghang Wu, Guanyu Li, and Bo Ning. 2024a. CDRGN-SDE: cross-dimensional recurrent graph network with neural stochastic differential equation for temporal knowledge graph embedding. *Expert Systems with Applications*, 247:123295.

Fu Zhang, Hongzhi Chen, Yuzhe Shi, Jingwei Cheng, and Jinghao Lin. 2024b. Joint framework for tensor decomposition-based temporal knowledge graph completion. *Information Sciences*, 654:119853.

Shuai Zhang, Yi Tay, Lina Yao, and Qi Liu. 2019. Quaternion knowledge graph embeddings. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 2731–2741.

Yichi Zhang, Zhuo Chen, Lingbing Guo, Yajing Xu, Wen Zhang, and Huajun Chen. 2024c. Making large language models perform better in knowledge graph completion. In *Proceedings of the 32nd ACM International Conference on Multimedia, MM 2024, Melbourne, VIC, Australia, 28 October 2024 - 1 November 2024*, pages 233–242. ACM.

Yu Zhao, Huali Feng, Han Zhou, Yanruo Yang, Xingyan Chen, Ruobing Xie, Fuzhen Zhuang, and Qing Li. 2022. EIGAT: incorporating global information in local attention for knowledge representation learning. *Knowledge-Based Systems*, 237:107909.

Jun Zhu, Jiahui Hu, Di Bai, Yan Fu, Junlin Zhou, and Duanbing Chen. 2025. Multi-dimension rotations based on quaternion system for modeling various patterns in temporal knowledge graphs. *Knowledge-Based Systems*, 311:113114.

A Datasets

1) Publicly Available Dataset. Table 4 summarizes statistics of temporal KGC datasets. ICEWS14, ICEWS05-15 and ICEWS18 are subsets of ICEWS. ICEWS acquires and processes millions of data from various sources to aid in monitoring to global events (e.g., (*Japan*, *Accuse*, *Korea*, *2014-3-3*)). ICEWS14, ICEWS05-15 and ICEWS18 separately collect events from *2014-1* to *2014-12*, *2005-1* to *2015-12*, and *2018-1* to *2018-12*. YAGO11k and Wikidata12k (Dasgupta et al., 2018) are subsets of YAGO3 and Wikidata. YAGO3 (Mahdisoltani et al., 2015) and Wikidata (Erxleben et al., 2014) are TKGs with time annotations in forms such as time

Dataset	Entities	Relations	Facts	Time Span
ICEWS14	6,869	230	90,730	2014
ICEWS05-15	10,488	251	461,329	2005-2015
ICEWS18	23,033	256	468,558	2018
YAGO11k	10,623	10	20,507	1513-2017
Wikidata12k	12,554	24	40,621	1526-2020
GDELT	500	20	3,419,607	2015-2016
YAGO15K	15,403	34	138,056	-

Table 4: Statistics for the various experimental datasets.

Dataset	Entities	Relations	Facts	Time Span
EPD21	51,785	320	120,907	2021
EPD22	258,455	325	747,166	2022
EPD23	1,077,558	336	3,602,870	2023
EPD24	210,889	323	394,973	2024

Table 5: Statistical summary of enterprise procurement dataset.

points (e.g., *Allan*, *BornIn*, *Launceston*, *2005-1-3*), start/end times (*Biden*, *Presidency*, *POTUS*, [*2021-01-02*, *####-#-#*]), and intervals (*Varick*, *Mayor*, *New York*, [*1789-3-2*, *1801-3-2*]). GDELT (Trivedi et al., 2017) is a global event database covering political, economic, and environmental events from *2015-4* to *2016-3*. YAGO15K extends FB15k (Bordes et al., 2013) by adding timestamps to each fact, with incomplete temporal information increasing the dataset's complexity.

2) Confidential Enterprise Procurement Dataset. EPD21, EPD22, EPD23, and EPD24 are subsets of Enterprise Procurement Dataset (EPD). The EPD is derived from annual procurement data of large enterprises and is designed to support research on procurement cost analysis, anomaly detection in transactions, and supply chain risk prediction (e.g., (*Company A*, *purchases*, *Material B*, *2021-05-08*) and (*Material B*, *price*, *1,375,442 CNY*)). Specifically, EPD21, EPD22, EPD23, and EPD24 cover events occurring from January to December of 2021, 2022, 2023, and 2024, respectively. The dataset details are shown in Table 5.

B Experimental Setup and Hyperparameter analysis

B.1 Experimental Setup

We implement TeCES in PyTorch and train it with Adagrad using a batch size of 1000. The temporal regularization weight λ_a and learning rate η. The temporal regularization weight λ_a is tuned within the range $\{0, 0.0025, 0.005, 0.0075, 0.01, 0.1, \ldots, 0.3\}$, while the learning rate η is searched within $\{0.3, 0.1, 0.05, 0.03, 0.01, \ldots, 0.001\}$. The best combinations of (λ_a, η) are: (0.01, 0.003) for ICEWS14, (0.1, 0.05) for ICEWS05-15, (0.1, 0.1) for ICEWS18, (0.03, 0.1) for YAGO11k, (0.025, 0.01) for Wikidata12k, (0.01, 0.1) for GDELT, and (0.025, 0.1) for YAGO15K. The optimal embedding dimension across all benchmarks is k = 100.

B.2 Hyperparameter Analysis

We conduct a comprehensive sensitivity analysis of hyperparameters, as shown in Fig. 12. Specifically, we adopt grid search on ICEWS14 to investigate the sensitivity of TeCES to the temporal regularization weight λ_a and learning rate η. The temporal regularization weight λ_a is tuned within the range $\{0, 0.0025, 0.005, 0.0075, 0.01, 0.1, \ldots, 0.3\}$, while the learning rate η is searched within $\{0.3, 0.1, 0.05, 0.03, 0.01, \ldots, 0.001\}$. Through observation, we find that the optimal result is achieved when (λ_a, λ_b) is set to (0.01, 0.003).

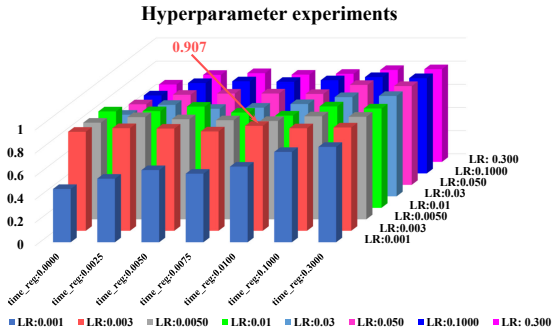

Figure 12: Hyperparameter experiments on ICEWS14.

B.3 Parameter Count Comparison

Fig. 13 presents a comparison of parameter sizes among TeCES, TeLM, and EHPR. Specifically, we report the parameter counts corresponding to each model's optimal performance. Compared with existing efficient models (TeLM and EHPR), TeCES consistently achieves more significant performance improvements across different datasets while maintaining a smaller parameter size. Moreover, as the dataset scale increases (from ICEWS14 to ICEWS05-15), the growth of TeCES's parameter size is slower and more stable, whereas EHPR and TeLM exhibit larger fluctuations. These findings indicate that TeCES provides a sustainable and stable solution for adaptive completion in large-scale KGs.

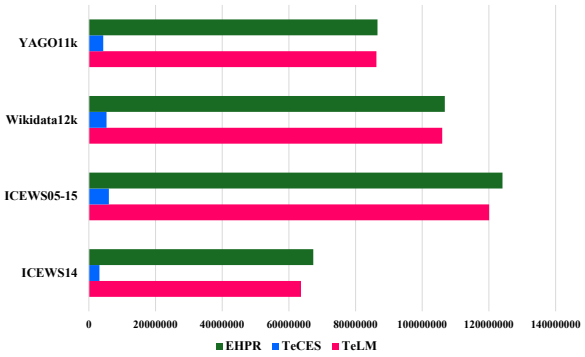

Figure 13: Parameter comparison among multiple efficient models.

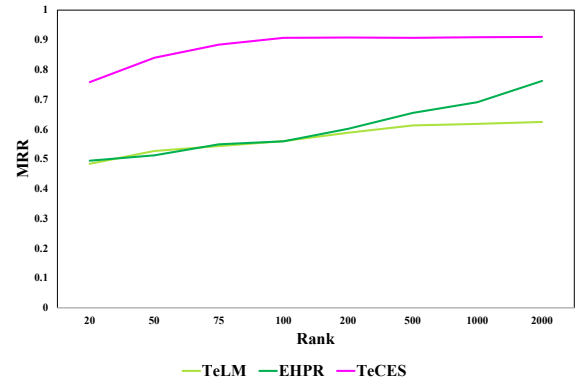

Figure 14: Results of different models with various embedding dimensions on ICEWS14.

B.4 Influence of embedding dimensionality

Fig. 14 illustrates MRR performance of TeLM, EHPR, and TeCES on ICEWS14 across different embedding dimensions $k = \{20, 50, 75, 100, 200, 500, 1000, 2000\}$. Among all models, TeCES consistently achieves the best or second-best performance across all embedding dimensions, verifying the effectiveness of the proposed framework and demonstrating its strong stability under low-rank conditions. Furthermore, when the embedding dimension $k \geq 100$, TeCES shows a gradual plateau in performance improvement and reaches its optimal point earlier, while EHPR and TeLM continue improving and peak at $k = 2000$. These results indicate that TeCES possesses superior parameter efficiency and generalization capability, making it particularly suitable for low-dimensional scenarios and more scalable and practical for large-scale TKGs.

C Main Detailed Results

Tables 1, 2 and 3 list temporal KGC results on each dataset. All results are taken from original papers. Compared with EHPR (**tensor family**), TeCES increases by 17.5 MRR points on ICEWS14, 13.5 MRR points on ICEWS05-15, 3.2 MRR points on GDELT, 3.0 MRR points on ICEWS18, 27.0 MRR points on YAGO11k, 29.7 MRR points on Wikidata12k, and 30.8 MRR points on YAGO15K. Compared with MDRQS (**deep learning family**), TeCES increases by 28.2 MRR points on ICEWS14, 26.2 MRR points on ICEWS05-15, 37.3 MRR points on GDELT, 37.3 MRR points on YAGO11k. Compared with DuaTHP (**LLMs family**), TeCES increases by 27.0 MRR points on ICEWS14, TeCES increases by 24.3 MRR points on ICEWS05-15, 52.2 MRR points on GDELT, 42.1 MRR points on YAGO11k, and 43.2 MRR points on Wikidata12k.

D Statistical Significance Analysis of Model Performance

Friedman Test Results. We use a Friedman test on the MRR values in Table 1. For the k models ($k = 18$) across n datasets ($n = 2$), the test yields a statistic of approximately 33.79. Under the F-distribution with $k-1$=17 and (k-1)(n-1)=17 degrees of freedom, the critical value at a significance level of α = 0.05 is approximately 2.49 (Iman–Davenport correction). Since the test statistic exceeds the critical value, we reject the null hypothesis and conclude that MRR performance of temporal KGC models differs significantly between the ICEWS14 and ICEWS05-15 datasets. Based on the average ranks, TeCES has the lowest average rank of 1.0, indicating the best performance; TuTR follows closely with a rank of 2.0; EHPR and GLARGCN have average ranks of 3.5, placing them in the second tier; while HFL and DLGR have the highest average ranks of 17.5, ranking at the bottom. This further confirms that the consistently top performance of TeCES is statistically meaningful and not due to random variation.

Nemenyi Test Results for Post-hoc Analysis. Building on the significant outcome of Friedman test, we further apply Nemenyi post-hoc test to identify which specific temporal KGC models differ significantly from one another. This test compares the average ranks of 18 models on ICEWS14 and ICEWS05-15 based on MRR, and resulting p-values are visualized in the heat-map shown in

Fig. 15. Experimentally, TeCES shows statistically significant differences from most SOTA models, as indicated by the darker blue cells (low p-values), demonstrating that it consistently outperforms the majority of baselines. In contrast, the reddish areas between models such as EHPR, GLARGCN, and TuTR indicate that their performance differences are not statistically significant. Overall, the heat map reveals that TeCES achieves statistically robust superiority over existing models under the Nemenyi test.

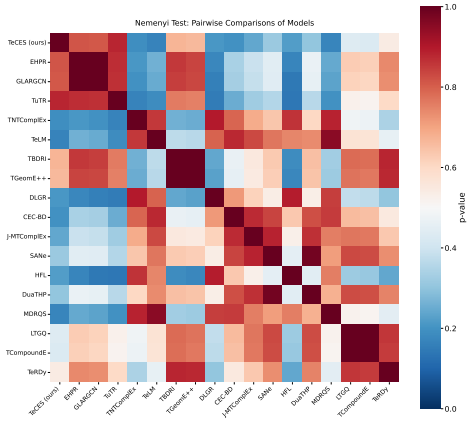

Figure 15: Nemenyi test results for multiple models on MRR metric.

Wilcoxon Test Results for Pairwise Comparisons. Building upon the significant outcome of the Friedman test, we further conduct the Wilcoxon signed-rank test to determine the statistical significance of pairwise MRR differences among models on ICEWS14 and ICEWS05-15. The heat map in Fig. 16 presents the two-tailed p-values for all model pairs. TeCES exhibits deep-blue cells in its row or column, indicating extremely low p-values (approximately 0.125) against other models. Therefore, TeCES is significantly superior to all 17 baseline models, including its strongest competitors, EHPR and GLARGCN. In contrast, the upper-right quadrant of the heat map shows light-orange blocks among lower-ranked models (such as DLGR), where p-values often exceed 0.5, suggesting that their MRR differences are not statistically significant. Overall, the Wilcoxon analysis further verifies that the superiority of TeCES is not only consistent but also statistically significant, while the remaining models form overlapping performance clusters with no clear pairwise advantage.

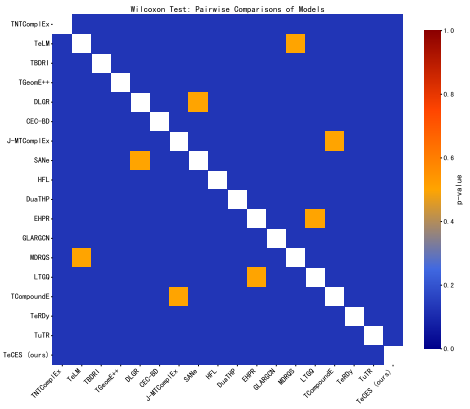

Figure 16: Wilcoxon test results for multiple models on MRR metric.

E Monthly Evaluation

Fig. 17 presents the monthly evaluation performance of TeLM, EHPR, and various variants of TeCES on ICEWS14. Overall, TeCES consistently outperforms existing baseline models and ablation variants across all months, demonstrating its strong robustness and temporal generalization capability. Moreover, TeCES w/o TA performs better than TeCES w/o RTF in most months, mainly because TA faces greater limitations in modeling across snapshots, whereas RTF is less constrained. The combination of RTF and TA further yields performance gains, indicating good compatibility and complementarity between the two components.

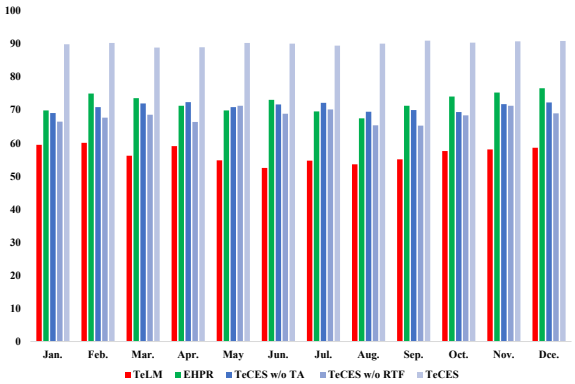

Figure 17: Monthly Evaluation across TeLM, EHPR, and TeCES Variants.

F Production-Level Deployment of Enterprise Procurement Data Inference

TeCES and CEFS are deployed for procurement data inference tasks in energy enterprises, as shown in Tables 6 and 7. Compared with existing efficient models (e.g., TeLM and EHPR), TeCES

and CEFS achieve the best and second-best performance, respectively, demonstrating the strong stability of our framework in real-world deployment. To further analyze the performance gains of TeCES, we visualize temporal distributions of facts across different datasets and conduct a longitudinal comparison, as shown in Fig. 18. Specifically, EPD exhibits a pronounced surge in data volume from February to August, while remaining relatively stable during the other months. Compared with EPD21, EPD22, and EPD24, EPD23 exhibits a pronounced high-density peak during this period. Nevertheless, TeCES and CEFS maintain strong performance, demonstrating their capability to model and accommodate the coupling between intra-snapshot relations and temporal semantics under high-density conditions. Meanwhile, the improvements of TeCES and CEFS on Hits@10 are relatively modest. This is mainly because the evolving snapshot framework prioritizes highly precise ranking for high-frequency entities, significantly boosting MRR and Hits@1, while yielding relatively limited gains within the broader top-10 candidate range. In addition, TeCES consistently achieves more pronounced performance gains than CEFS, further demonstrating the superior representational capacity of the geometric product over Hermitian operators in high-density factual scenarios. In summary, the TeCES framework has been successfully applied to supply-chain reasoning tasks in energy enterprise data systems. Looking ahead, we will further explore intrinsic characteristics of supply-chain data, such as seasonal skewed peaks, to achieve more efficient, robust, and scalable reasoning capabilities, better supporting real-world industrial environments.

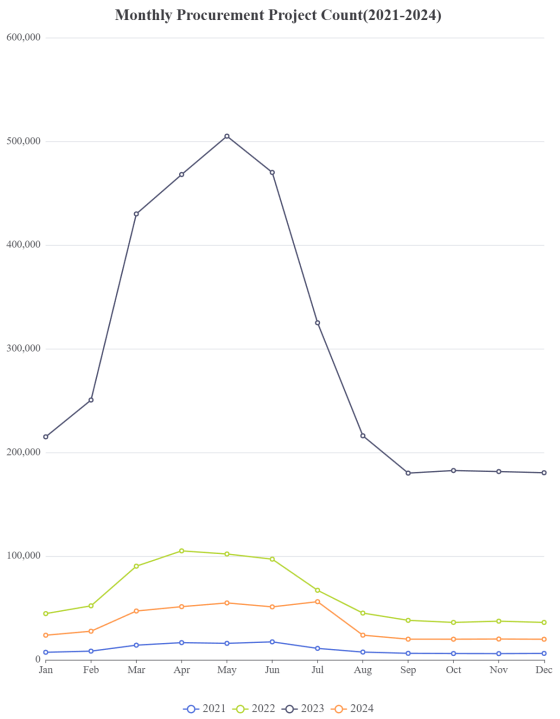

Figure 18: Fact distribution comparison visualization across similar datasets.

G Implementation Details of Compatibility Components

G.1 CEFS for temporal KGC

This section follows the principles upon which we built CEFS, we construct a model named CEFS based on complex embeddings. For CEFS, head entity s and tail entity o correspond to $C_s = \{ a_s + e_s\mathbf{i} : a_s, e_s \in \mathbb{R}^k \}$ and $C_o = \{ a_o + e_o\mathbf{i} : a_o, e_o \in \mathbb{R}^k \}$, respectively, while relation r and timestamp τ correspond to $C_r = \{ a_r + e_r\mathbf{i} : a_r, e_r \in \mathbb{R}^k \}$ and $C_\tau = \{ a_\tau + e_\tau\mathbf{i} : a_\tau, e_\tau \in \mathbb{R}^k \}$, respectively. The specific steps are as follows:

To ensure that entities can be stably mapped to different specific snapshots, CEFS attaches each grade component of the timestamp complex embedding C_τ respectively to the corresponding components of head entity complex embedding C_s and tail entity complex embedding C_o:

$$\widetilde{C}_s = f_0(a_s, a_\tau) + f_1(e_s, e_\tau)\mathbf{i}. \tag{10}$$

$$\widetilde{C}_o = g_0(a_o, a_\tau) + g_1(e_o, e_\tau)\mathbf{i}. \tag{11}$$

Here, $\widetilde{C}_s$ and $\widetilde{C}_o$ respectively represent head and tail entities under a specific snapshot space; $f_i(\cdot,\cdot)$ and $g_i(\cdot,\cdot)$ denote additive fusion operations applied to the head and tail entities with the timestamp, respectively, ensuring balanced integration and traceability of entity semantics within snapshots. Furthermore, CEFS reconfigures relation complex embedding $C_r = a_r + e_r\mathbf{i}$ and timestamp complex embedding $C_\tau = a_\tau + e_\tau\mathbf{i}$ into two composite complex embedding, namely $C_{r\tau} = a_r + e_\tau\mathbf{i}$ and $C_{\tau r} = a_\tau + e_r\mathbf{i}$. CEFS then uses hermitian operator to realize interaction between $C_{r\tau}$ and $C_{\tau r}$:

$$\mathbf{C}_{\text{geo}} = C_{r\tau} \times_2 C_{\tau r}. \tag{12}$$

Finally, CEFS performs a hermitian operator to interact $\widetilde{C}_s$ with $\mathbf{C}_{\text{geo}}$, and further applies an inner product with $\widetilde{C}_o$ to obtain final score representation of CEFS:

$$\varphi(s, r, o, \tau) = (\widetilde{C}_s\mathbf{C}_{\text{geo}}) \cdot \widetilde{C}_o, \tag{13}$$

	EPD21				EPD22			
	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
TComplEx†	20.7	18.1	23.4	29.0	24.4	20.3	25.2	30.8
TeLM†	25.3	23.9	29.4	35.1	32.2	27.1	34.0	40.6
EHPR†	31.6	29.7	36.5	<u>46.4</u>	40.8	38.7	46.1	<u>57.4</u>
CEFS (ours)	<u>33.2</u>	<u>30.3</u>	<u>37.1</u>	42.6	<u>43.7</u>	<u>40.5</u>	<u>46.4</u>	52.1
TeCES (ours)	**42.0**	**41.5**	**44.8**	**47.7**	**55.0**	**51.4**	**55.1**	**60.2**
APG (%)	8.8 ↑	11.2 ↑	9.4 ↑	5.1 ↑	11.3 ↑	10.9 ↑	8.7 ↑	8.1 ↑

Table 6: Experimental results on EPD21 and EPD22. † All results reported are reproduction results. The best results among all models are written **bold**. The second-best results among all models are written <u>underlined</u>. ↑ and ↓ denote performance gain and drop relative to baseline model, respectively.

	EPD23				EPD24			
	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
TComplEx†	21.5	17.7	25.1	33.7	35.1	29.6	37.2	49.1
TeLM†	23.4	21.1	27.2	35.0	33.2	26.1	36.0	54.2
EHPR†	29.0	24.5	35.0	<u>44.4</u>	40.1	35.7	43.4	53.4
CEFS (ours)	<u>35.2</u>	<u>33.4</u>	<u>38.6</u>	<u>44.4</u>	<u>48.7</u>	<u>45.1</u>	<u>49.4</u>	<u>55.0</u>
TeCES (ours)	**44.1**	**41.7**	**45.6**	**49.3**	**60.9**	**58.7**	**62.4**	**66.7**
APG (%)	8.9 ↑	8.3 ↑	7.0 ↑	4.9 ↑	12.2 ↑	13.6 ↑	13.0 ↑	11.7 ↑

Table 7: Experimental results on EPD23 and EPD24. † All results reported are reproduction results. The best results among all models are written **bold**. The second-best results among all models are written <u>underlined</u>. ↑ and ↓ denote performance gain and drop relative to baseline model, respectively.

where $\cdot$ is inner product.

G.2 CEFS w/o TA

For CEFS w/o TA, we remove Timestamp Attachment (TA), so temporal information is no longer embedded in entity representations for snapshot-specific mapping. Specifically, head entity s and tail entity o correspond to $C_s = \{ a_s + e_s\mathbf{i} : a_s, e_s \in \mathbb{R}^k \}$ and $C_o = \{ a_o + e_o\mathbf{i} : a_o, e_o \in \mathbb{R}^k \}$, respectively, while relation r and timestamp τ correspond to $C_r = \{ a_r + e_r\mathbf{i} : a_r, e_r \in \mathbb{R}^k \}$ and $C_\tau = \{ a_\tau + e_\tau\mathbf{i} : a_\tau, e_\tau \in \mathbb{R}^k \}$, respectively. The specific steps are as follows:

First, CEFS w/o TA reconfigures relation complex embedding $C_r = a_r + e_r\mathbf{i}$ and timestamp complex embedding $C_\tau = a_\tau + e_\tau\mathbf{i}$ into two composite complex embedding, namely $C_{r\tau} = a_r + e_\tau\mathbf{i}$ and $C_{\tau r} = a_\tau + e_r\mathbf{i}$. CEFS w/o TA then uses hermitian operator to realize interaction between $C_{r\tau}$ and $C_{\tau r}$:

$$\mathbf{C}_{\text{geo}} = C_{r\tau} \times_2 C_{\tau r}. \tag{14}$$

Then, CEFS w/o TA performs a hermitian operator to interact C_s with $\mathbf{C}_{\text{geo}}$, and further applies an inner product with C_o to obtain final score representation of CEFS w/o TA:

$$\varphi(s, r, o, \tau) = (C_s\mathbf{C}_{\text{geo}}) \cdot C_o, \tag{15}$$

where $\cdot$ is inner product.

G.3 CEFS w/o RTF

For CEFS w/o RTF, we remove the interaction between the relation and the timestamp. Specifically, head entity s and tail entity o correspond to $C_s = \{ a_s + e_s\mathbf{i} : a_s, e_s \in \mathbb{R}^k \}$ and $C_o = \{ a_o + e_o\mathbf{i} : a_o, e_o \in \mathbb{R}^k \}$, respectively, while relation r and timestamp τ correspond to $C_r = \{ a_r + e_r\mathbf{i} : a_r, e_r \in \mathbb{R}^k \}$ and $C_\tau = \{ a_\tau + e_\tau\mathbf{i} : a_\tau, e_\tau \in \mathbb{R}^k \}$, respectively. The specific steps are as follows:

First, to ensure that entities can be stably mapped to different specific snapshots, CEFS w/o RTF attaches each grade component of the timestamp complex embedding C_τ respectively to the corresponding components of head entity complex embedding C_s and tail entity complex embedding C_o:

$$\widetilde{C}_s = f_0(a_s, a_\tau) + f_1(e_s, e_\tau)\mathbf{i}. \tag{16}$$

$$\widetilde{C}_o = g_0(a_o, a_\tau) + g_1(e_o, e_\tau)\mathbf{i}. \tag{17}$$

Here, $\widetilde{C}_s$ and $\widetilde{C}_o$ respectively represent head and tail entities under a specific snapshot space; $f_i(\cdot, \cdot)$

and $g_i(\cdot,\cdot)$ denote additive fusion operations applied to the head and tail entities with the timestamp, respectively, ensuring balanced integration and traceability of entity semantics within snapshots. CEFS w/o RTF then uses hermitian operator to realize interaction between C_r and C_τ:

$$\mathbf{C}_{\text{geo}} = C_r \times_2 C_\tau. \tag{18}$$

Finally, CEFS w/o RTF performs a hermitian operator to interact $\widetilde{C}_s$ with $\mathbf{C}_{\text{geo}}$, and further applies an inner product with $\widetilde{C}_o$ to obtain final score representation of CEFS w/o RTF:

$$\varphi(s,r,o,\tau) = (\widetilde{C}_s \mathbf{C}_{\text{geo}}) \cdot \widetilde{C}_o, \tag{19}$$

where $\cdot$ is inner product.

G.4 TeCES for temporal KGC

For TeCES, the head entity s, relation r, tail entity o, and timestamp τ are embedded as $E_s = \{\, s_0 + s_1e_1 + s_2e_2 + s_{12}e_1e_2 : s_0, s_1, s_2, s_{12} \in \mathbb{R}^k \,\}$, $R_r = \{\, r_0 + r_1e_1 + r_2e_2 + r_{12}e_1e_2 : r_0, r_1, r_2, r_{12} \in \mathbb{R}^k \,\}$, $E_o = \{\, o_0 + o_1e_1 + o_2e_2 + o_{12}e_1e_2 : o_0, o_1, o_2, o_{12} \in \mathbb{R}^k \,\}$, and $T_\tau = \{\, \tau_0 + \tau_1e_1 + \tau_2e_2 + \tau_{12}e_1e_2 : \tau_0, \tau_1, \tau_2, \tau_{12} \in \mathbb{R}^k \,\}$, respectively. The specific steps are as follows:

To ensure that entities can be stably mapped to different specific snapshots, TeCES attaches each grade component of the timestamp's 2-grade multivector T_τ respectively to the corresponding components of head entity's 2-grade multivector E_s and tail entity's 2-grade multivector E_o :

$$\begin{aligned}\widetilde{E}_s = {} & f_0(s_0,\tau_0) + f_1(s_1,\tau_1)e_1 \\ & + f_2(s_2,\tau_2)e_2 + f_{12}(s_{12},\tau_{12})e_1e_2.\end{aligned} \tag{20}$$

$$\begin{aligned}\widetilde{E}_o = {} & g_0(o_0,\tau_0) + g_1(o_1,\tau_1)e_1 \\ & + g_2(o_2,\tau_2)e_2 + g_{12}(o_{12},\tau_{12})e_1e_2.\end{aligned} \tag{21}$$

Here, $\widetilde{E}_s$ and $\widetilde{E}_o$ respectively represent the head and tail entities under a specific snapshot space; $f_i(\cdot,\cdot)$ and $g_i(\cdot,\cdot)$ denote additive fusion operations applied to the head and tail entities with the timestamp, respectively, ensuring balanced integration and traceability of entity semantics within snapshots. Furthermore, to enable multidimensional and deep interactions between temporal and relational information, and to uncover relation states driven by snapshot progression, TeCES reconfigures the relation's 2-grade multivector R_r = $r_0 + r_1e_1 + r_2e_2 + r_{12}e_1e_2$ and the timestamp's 2-grade multivector T_τ = $\tau_0 + \tau_1e_1 + \tau_2e_2 + \tau_{12}e_1e_2$ into two composite 2-grade multivectors, namely $C_{r\tau}$ = $r_0 + r_1e_1 + r_2e_2 + \tau_{12}e_1e_2$ and $C_{\tau r}$ = $\tau_0 + \tau_1e_1 + \tau_2e_2 + r_{12}e_1e_2$. TeCES then uses geometric product to realize interaction between $C_{r\tau}$ and $C_{\tau r}$:

$$\mathbf{C}_{\text{geo}} = C_{r\tau} \times_2 C_{\tau r}. \tag{22}$$

Finally, TeCES performs a geometric product to interact $\widetilde{E}_s$ with $\mathbf{C}_{\text{geo}}$, and further applies an inner product with $\widetilde{E}_o$ to obtain final score representation of TeCES:

$$\varphi(s,r,o,\tau) = (\widetilde{E}_s \times_2 \mathbf{C}_{\text{geo}}) \cdot \widetilde{E}_o, \tag{23}$$

where $\cdot$ is inner product.

G.5 TeCES w/o TA

For TeCES w/o TA, we remove Timestamp Attachment (TA), so temporal information is no longer embedded in entity representations for snapshot-specific mapping. Specifically, the head entity s, relation r, tail entity o, and timestamp τ are embedded as $E_s = \{\, s_0 + s_1e_1 + s_2e_2 + s_{12}e_1e_2 : s_0, s_1, s_2, s_{12} \in \mathbb{R}^k \,\}$, $R_r = \{\, r_0 + r_1e_1 + r_2e_2 + r_{12}e_1e_2 : r_0, r_1, r_2, r_{12} \in \mathbb{R}^k \,\}$, $E_o = \{\, o_0 + o_1e_1 + o_2e_2 + o_{12}e_1e_2 : o_0, o_1, o_2, o_{12} \in \mathbb{R}^k \,\}$, and $T_\tau = \{\, \tau_0 + \tau_1e_1 + \tau_2e_2 + \tau_{12}e_1e_2 : \tau_0, \tau_1, \tau_2, \tau_{12} \in \mathbb{R}^k \,\}$, respectively. The specific steps are as follows:

First, TeCES w/o TA reconfigures the relation's 2-grade multivector R_r = $r_0 + r_1e_1 + r_2e_2 + r_{12}e_1e_2$ and the timestamp's 2-grade multivector T_τ = $\tau_0 + \tau_1e_1 + \tau_2e_2 + \tau_{12}e_1e_2$ into two composite 2-grade multivectors, namely $C_{r\tau}$ = $r_0 + r_1e_1 + r_2e_2 + \tau_{12}e_1e_2$ and $C_{\tau r}$ = $\tau_0 + \tau_1e_1 + \tau_2e_2 + r_{12}e_1e_2$. TeCES w/o TA then uses geometric product to realize interaction between $C_{r\tau}$ and $C_{\tau r}$:

$$\mathbf{C}_{\text{geo}} = C_{r\tau} \times_2 C_{\tau r}. \tag{24}$$

Finally, TeCES w/o TA performs a geometric product to interact E_s with $\mathbf{C}_{\text{geo}}$, and further applies an inner product with E_o to obtain final score representation of TeCES w/o TA:

$$\varphi(s,r,o,\tau) = (E_s \times_2 \mathbf{C}_{\text{geo}}) \cdot E_o, \tag{25}$$

where $\cdot$ is inner product.

G.6 TeCES w/o RTF

For TeCES w/o RTF, we remove the interaction between relations and the timestamps. Specifically,

the head entity s, relation r, tail entity o, and timestamp τ are embedded as $E_s = \{ s_0 + s_1e_1 + s_2e_2 + s_{12}e_1e_2 : s_0, s_1, s_2, s_{12} \in \mathbb{R}^k \}$, $R_r = \{ r_0 + r_1e_1 + r_2e_2 + r_{12}e_1e_2 : r_0, r_1, r_2, r_{12} \in \mathbb{R}^k \}$, $E_o = \{ o_0 + o_1e_1 + o_2e_2 + o_{12}e_1e_2 : o_0, o_1, o_2, o_{12} \in \mathbb{R}^k \}$, and $T_\tau = \{ \tau_0 + \tau_1e_1 + \tau_2e_2 + \tau_{12}e_1e_2 : \tau_0, \tau_1, \tau_2, \tau_{12} \in \mathbb{R}^k \}$, respectively. The specific steps are as follows:

To ensure that entities can be stably mapped to different specific snapshots, TeCES w/o RTF attaches each grade component of timestamp's 2-grade multivector T_τ respectively to the corresponding components of head entity's 2-grade multivector E_s and tail entity's 2-grade multivector E_o :

$$\widetilde{E}_s = f_0(s_0, \tau_0) + f_1(s_1, \tau_1)e_1 + f_2(s_2, \tau_2)e_2 + f_{12}(s_{12}, \tau_{12})e_1e_2. \tag{26}$$

$$\widetilde{E}_o = g_0(o_0, \tau_0) + g_1(o_1, \tau_1)e_1 + g_2(o_2, \tau_2)e_2 + g_{12}(o_{12}, \tau_{12})e_1e_2. \tag{27}$$

Here, $\widetilde{E}_s$ and $\widetilde{E}_o$ respectively represent the head and tail entities under a specific snapshot space; $f_i(\cdot,\cdot)$ and $g_i(\cdot,\cdot)$ denote additive fusion operations applied to the head and tail entities with the timestamp, respectively, ensuring balanced integration and traceability of entity semantics within snapshots. TeCES w/o RTF then uses hermitian operator to realize interaction between $R_r = r_0 + r_1e_1 + r_2e_2 + r_{12}e_1e_2$ and $T_\tau = \tau_0 + \tau_1e_1 + \tau_2e_2 + \tau_{12}e_1e_2$:

$$\mathbf{C}_{\text{geo}} = R_r \times_2 T_\tau. \tag{28}$$

Finally, TeCES w/o RTF performs a geometric product to interact $\widetilde{E}_s$ with $\mathbf{C}_{\text{geo}}$, and further applies an inner product with $\widetilde{E}_o$ to obtain final score representation of TeCES w/o RTF:

$$\varphi(s, r, o, \tau) = (\widetilde{E}_s \times_2 \mathbf{C}_{\text{geo}}) \cdot \widetilde{E}_o, \tag{29}$$

where $\cdot$ is inner product.

G.7 Training Algorithm of CEFS and TeCES

To further illustrate our framework and implementation details, we additionally provide reproducible pseudocode to demonstrate the algorithmic process and facilitate experimental replication. Algorithm 1 outlines the training procedure for the proposed CEFS. Algorithm 2 illustrates the training pipeline of the proposed TeCES.

Algorithm 1 CEFS training algorithm.

Input: (1) A TKG $\mathcal{KG}$; (2) Entities $\mathscr{E}$; (3) Relations $\mathscr{R}$; (4) Timestamps $\mathscr{T}$; (5) Hyperparameters *max_epochs*, rank d, λ_a, λ_b

Output: C_s, C_r, C_o, C_τ for $\forall\, s, o \in \mathcal{E}$, $\forall\, r \in \mathcal{R}$, $\forall\, \tau \in \mathcal{T}$

Initialize embeddings:
  $C_s, C_o \sim \mathcal{U}(-1, 1)^{d\times 2}$ for $\forall s, o \in \mathcal{E}$
  $C_r \sim \mathcal{U}(-1, 1)^{d\times 2}$ for $\forall r \in \mathcal{R}$
  $C_\tau \sim \mathcal{U}(-1, 1)^{d\times 2}$ for $\forall \tau \in \mathcal{T}$
for *epoch* = 1 to *max_epochs* **do**
  $L_{\text{total}} \leftarrow 0$
  for $(s, r, o, \tau) \in \mathcal{G}$ **do**
    Decompose $C_s, C_o \in \mathbb{C}^{d\times 2}$ as: $C_s \leftarrow (a_s, e_s)$, $C_o \leftarrow (a_o, e_o)$
    Decompose $R_r \in \mathbb{C}^{d\times 2}$ as: $C_r \leftarrow (a_r, e_r)$
    Decompose $T_\tau \in \mathbb{C}^{d\times 2}$ as: $T_\tau \leftarrow (a_\tau, e_\tau)$
    $\widetilde{C}_s \leftarrow (a_s + a_\tau, e_s + e_\tau)$
    $\widetilde{C}_o \leftarrow (a_o + a_\tau, e_o + e_\tau)$
    $C_{r\tau} \leftarrow (a_r, e_r)$
    $C_{\tau r} \leftarrow (a_\tau, e_\tau)$
    $\mathbf{C}_{\text{geo}} \leftarrow C_{r\tau} C_{\tau r}$
    $\varphi(s, r, o, \tau) \leftarrow (\widetilde{C}_s \mathbf{C}_{\text{geo}}) \cdot \widetilde{C}_o$
    $L_{\text{cross\_entropy}} \leftarrow 0$
    for $(s', r, o, \tau) \in \Pi'$ **do**
      $L_{s'} \leftarrow \log \frac{\exp(\varphi_{(s,r,o,\tau)})}{\sum_{s'\in\Pi'} \exp(\varphi_{(s',r,o,\tau)})}$
      $L_{\text{cross\_entropy}} \leftarrow L_{\text{cross\_entropy}} - L_{s'}$
    end for
    for $o' \in \Pi'$ **do**
      $L_{o'} \leftarrow \log \frac{\exp(\varphi_{(s,r,o,\tau)})}{\sum_{o'\in\Pi'} \exp(\varphi_{(s,r,o',\tau)})}$
      $L_{\text{cross\_entropy}} \leftarrow L_{\text{cross\_entropy}} - L_{o'}$
    end for
    $L_{\text{total}} \leftarrow L_{\text{total}} + L_{\text{cross\_entropy}}$
    $L_{\text{total}} \leftarrow L_{\text{total}} + \lambda_a \mathfrak{L}_a$
  end for
end for
Return C_s, C_r, C_o, C_τ

Algorithm 2 TeCES training Algorithm.

Input: (1) A TKG $\mathcal{KG}$; (2) Entities $\mathscr{E}$; (3) Relations $\mathscr{R}$; (4) Timestamps $\mathscr{T}$; (5) Hyperparameters *max_epochs*, rank d, λ_a, λ_b

Output: E_s, R_r, E_o, T_τ for $\forall\, s, o \in \mathcal{E}$, $\forall\, r \in \mathcal{R}$, $\forall\, \tau \in \mathcal{T}$

1: Initialize embeddings:
  1.1: $E_s, E_o \sim \mathcal{U}(-1,1)^{d\times 4}$ for $\forall s, o \in \mathcal{E}$
  1.2: $R_r \sim \mathcal{U}(-1,1)^{d\times 4}$ for $\forall r \in \mathcal{R}$
  1.3: $T_\tau \sim \mathcal{U}(-1,1)^{d\times 4}$ for $\forall \tau \in \mathcal{T}$
2: **for** *epoch* = 1 to *max_epochs* **do**
3: $L_{\text{total}} \leftarrow 0$
4: **for** $(s, r, o, \tau) \in \mathcal{G}$ **do**
5: Decompose $E_s, E_o \in \mathbb{R}^{d\times 4}$ as: $E_s \leftarrow (s_0, s_1, s_2, s_{12})$, $E_o \leftarrow (o_0, o_1, o_2, o_{12})$
6: Decompose $R_r \in \mathbb{R}^{d\times 4}$ as: $R_r \leftarrow (r_0, r_1, r_2, r_{12})$
7: Decompose $T_\tau \in \mathbb{R}^{d\times 4}$ as: $T_\tau \leftarrow (\tau_0, \tau_1, \tau_2, \tau_{12})$
7: $\widetilde{E}_s \leftarrow (s_0+\tau_0, s_1+\tau_1, s_2+\tau_2, s_{12}+\tau_{12})$
7: $\widetilde{E}_o \leftarrow (o_0+\tau_0, o_1+\tau_1, o_2+\tau_2, o_{12}+\tau_{12})$
8: $C_{r\tau} \leftarrow (r_0, r_1, r_2, \tau_{12})$
9: $C_{\tau r} \leftarrow (\tau_0, \tau_1, \tau_2, r_{12})$
10: $\mathbf{C}_{\text{geo}} \leftarrow C_{r\tau} \times_2 C_{\tau r}$
10: $\varphi(s, r, o, \tau) \leftarrow (\widetilde{E}_s \times_2 \mathbf{C}_{\text{geo}}) \cdot \widetilde{E}_o$
17: $L_{\text{cross\_entropy}} \leftarrow 0$
18: **for** $(s', r, o, \tau) \in \Pi'$ **do**
19: $L_{s'} \leftarrow \log \frac{\exp(\varphi_{(s,r,o,\tau)})}{\sum_{s'\in\Pi'} \exp(\varphi_{(s',r,o,\tau)})}$
20: $L_{\text{cross\_entropy}} \leftarrow L_{\text{cross\_entropy}} - L_{s'}$
21: **end for**
22: **for** $o' \in \Pi'$ **do**
23: $L_{o'} \leftarrow \log \frac{\exp(\varphi_{(s,r,o,\tau)})}{\sum_{o'\in\Pi'} \exp(\varphi_{(s,r,o',\tau)})}$
24: $L_{\text{cross\_entropy}} \leftarrow L_{\text{cross\_entropy}} - L_{o'}$
25: **end for**
26: $L_{\text{total}} \leftarrow L_{\text{total}} + L_{\text{cross\_entropy}}$
27: $L_{\text{total}} \leftarrow L_{\text{total}} + \lambda_a \mathfrak{L}_a$
28: **end for**
29: **end for**
30: **Return** E_s, R_r, E_o, T_τ