HGE: Embedding Temporal Knowledge Graphs in a Product Space of Heterogeneous Geometric Subspaces

Jiaxin Pan,[1] **Mojtaba Nayyeri,** [1] **Yinan Li** [1] **Steffen Staab** [1,2]

[1] University of Stuttgart, Stuttgart, Germany
[2] University of Southampton, Southampton, United Kingdom
jiaxin.pan@ki.uni-stuttgart.de, mojtaba.nayyeri@ki.uni-stuttgart.de,
yinan9721@gmail.com, steffen.staab@ki.uni-stuttgart.de

Abstract

Temporal knowledge graphs represent temporal facts (s, p, o, τ) relating a subject s and an object o via a relation label p at time τ, where τ could be a time point or time interval. Temporal knowledge graphs may exhibit static temporal patterns at distinct points in time and dynamic temporal patterns between different timestamps. In order to learn a rich set of static and dynamic temporal patterns and apply them for inference, several embedding approaches have been suggested in the literature. However, as most of them resort to single underlying embedding spaces, their capability to model all kinds of temporal patterns was severely limited by having to adhere to the geometric property of their one embedding space. We lift this limitation by an embedding approach that maps temporal facts into a product space of several heterogeneous geometric subspaces with distinct geometric properties, i.e. Complex, Dual, and Split-complex spaces. In addition, we propose a temporal-geometric attention mechanism to integrate information from different geometric subspaces conveniently according to the captured relational and temporal information. Experimental results on standard temporal benchmark datasets favorably evaluate our approach against state-of-the-art models.

1 Introduction

Knowledge Graphs (KGs) (Hogan et al. 2021) model facts in real-world applications as directed edge-labeled graphs. Temporal KGs (TKGs) include timestamps to their facts in order to model the temporal validity of facts. Depending on the representational model, timestamps may represent time points or time intervals. For instance, a quadruple *(Boris Johnson, IsPrimeministerOf, UK, [2019, 2022])* in a TKG represents the fact that Boris Johnson is the prime minister of UK between 2019 and 2022.

Relations in temporal knowledge graphs may exhibit various structural temporal patterns. In the left part of Figure 1, *(Charles III, marriedWith, Camilla, 2005)* and *(Camilla, marriedWith, Charles III, 2005)* forms a symmetrical structure in time. In the middle part, at first *(Elizabeth Bowes-Lyon, hasChild, Elizabeth II, 1926)* and then *(Elizabeth II, hasChild, Charles III, 1948)*. The transition of *hasChild* relation through *Elizabeth II* forms a hierarchy structure in
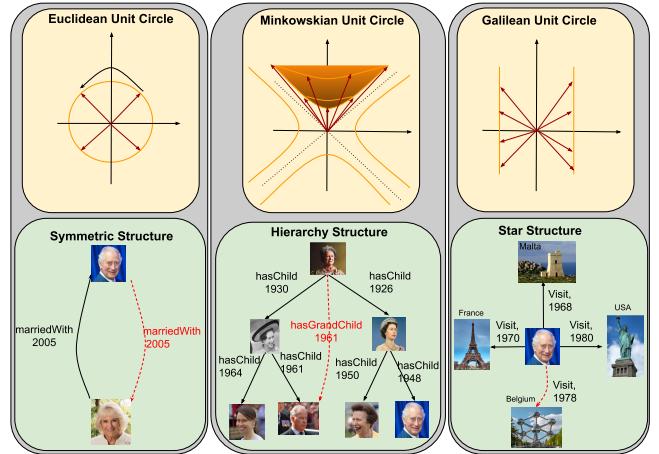
Figure 1: Unit spheres in their corresponding spaces. All points on the orange hyperplanes have the same distance to their origin. Different spaces favor different temporal patterns: **Left:** Unit circle represented in Complex space (top) is suitable for representing periodicities and for inferencing with 'periodic' logical temporal patterns, e.g. symmetry (bottom). **Middle:** Minkowskian unit circle in Split-complex space (top) is suitable for representing a temporal hierarchy formed by *Make statement*. **Right:** Galilean unit circle represented in Dual space (top) is suitable for representing temporal star patterns (bottom).

TKGs. In the right part, *Charles III*, *Visit Malta*, *France, Belgium*, *USA* etc at different timestamps, forming a star structure over time. Moreover, as *Charles III* shows, the structures which entities are involved in temporal knowledge graphs may evolve over time. How to preserve different relational structural patterns and how to capture evolving temporal patterns for entities is a fundamental challenge in TKGEs.

Existing embedding approaches such as TeRO, Rotate-QVS, and TLT-KGE(Xu et al. 2020; Chen et al. 2022; Zhang et al. 2022) resorted to single underlying embedding spaces, such as Complex space or Quaternion space to model symmetric patterns by the rotations on a unit hypersphere. Other works (Chami et al. 2020; Balazevic, Allen, and Hospedales

2019; Montella, Barahona, and Heinecke 2021; Han et al. 2020) use hyperbolic space to preserve hierarchical patterns in temporal KGs. However, their capability to model all kinds of structural patterns was severely limited by having to adhere to the geometric properties of their one embedding space. (Han et al. 2020) has shown the advantage of using multiple geometric subspaces (spherical, hyperbolic, etc) in different dimensions to preserve heterogeneous structural patterns in temporal KGs. However, it ignores the evolution of structural patterns between entities and requires a manual selection of subspaces dimension. How to integrate suitable subsets of geometries to model different relational structural patterns as well as capturing evolutionary temporal patterns between entities remain an open problem in these approaches.

In this paper, we address these problems by introducing a new product space covering various geometric subspaces namely a) complex, b) split-complex and c) dual spaces with a *temporal relational attention mechanism* and a *temporal geometric attention mechanism* to model both structural and evolutionary temporal patterns. Figure 1 illustrates the spaces and some corresponding patterns. a) Consider the left part of Figure 1: In the complex space, Euclidean unit circles are induced by circular rotations. Thus, points on the circle establish periodicities and various logical temporal patterns, e.g. relations that are symmetry in time (Xu et al. 2020). Circular rotations are modeled by circular sine and cosine functions in the complex space. b) Consider the middle part of Figure 1: In the split-complex space, a Minkowskian unit circle is induced through hyperbolic rotation, where points on the circle can be mapped using hyperbolic sine and cosine. Thus, the split-complex space can capture a temporal hierarchy, e.g. children must be born after their parents. c) Consider the right part of Figure 1: In the dual space, a Galilean unit circle is induced by the rotation that maps points on the circle using Galilean sine and cosine. Points on the induced circle (two parallel lines) are equidistant to the center, making it useful for modeling star-shaped subgraphs.

The combination of these three spaces together with their geometries and corresponding operators allows for capturing diverse logical and structural patterns such as relational symmetry in time, temporal hierarchy patterns, and temporal star patterns. Which geometry should be preferred in a specific case, however, needs to be learned. For this purpose, we provide a temporal geometric attention mechanism to select the preferred geometries for a given relation and time. Moreover, to deal with the evolution of patterns between entities, we propose the temporal-relational attention mechanism to balance static embedding and time-evolving embedding. We compare our TKGE model, heterogeneous geometric embedding (HGE), to TKGE methods in Complex space such as TComplEx (Lacroix, Obozinski, and Usunier 2020), TeRo (Xu et al. 2020), TLT-KGE (Zhang et al. 2022) and find that our model obtains better results for link prediction tasks in TKGs. In summary, the key contributions of this paper are as follows:

- We extend state-of-the-art Temporal Knowledge Graph Embedding (TKGE) models that use Complex spaces to

a new method, HGE. By utilizing multiple heterogeneous geometries, HGE embeds temporal facts in a product space of Complex, Split-complex, and Dual subspaces.

- Our theoretical analysis shows that our embedding method can capture a range of various structural and logical temporal patterns by utilizing the rotation operations acting on Euclidean, Minkowskian, and Galilean unit circles. These theoretical considerations are supported by experiments and ablation studies on pre-existing benchmark datasets.

- Two novel kinds of attention mechanisms, temporal-relational attention, and temporal-geometric attention allow for representing relation changing frequencies and suitable geometries, respectively.

- Experimental results on benchmark datasets show that HGE uniformly improves several state-of-the-art TKGE models. Subsequent ablation studies verify the general benefit of the attention-based product space models over the Complex space.

2 Preliminaries

Definition 1 (Time Interval). *Let $\mathbb{T}$ be the set of closed intervals on the real line $\mathbb{R}$. For a time interval $\tau = [m, n] \in \mathbb{T}, \tau \subseteq \mathbb{R}$, with $m, n \in \tau, m \leq n$ it holds that $\forall t \in \mathbb{R} : m \leq t \leq n \Rightarrow t \in \tau$.*

Definition 2 (Temporal Knowledge Graph). *Let V be a set of vertices, R be a set of relation labels, $\mathbb{T}$ be the set of all time intervals, $G \subseteq V \times R \times V \times \mathbb{T}$, then a temporal fact $(s, p, o, \tau) \in G$ with subject s, object o and relation label p is valid during time interval τ. A temporal knowledge graph $TKG = (V, R, G)$ defines a set of temporal facts. In addition, we denote G_i as i-th snapshot of the TKG*

We re-use Allen's interval calculus to express relations between time intervals (Allen 1983). It defines 13 possible relations between two time intervals such that these relations are exhaustive and pairwise disjoint. For example, Allen relation Contains(τ_1, τ_2) holds between two time intervals $\tau_1 = [m_1, n_1], \tau_2 = [m_2, n_2]$ if $m_1 < m_2 < n_2 < n_1$. Following (Singh et al. 2023), we refer to the 13 relations of Allen interval calculus as *Allen relations* and the relation in temporal knowledge graphs as *KG relations*. Appendix A describes the details of 13 Allen relations.

3 Embedding Model in Heterogeneous Geometric Subspaces

To capture heterogeneous structural and logical patterns in a temporal KG, we propose the HGE model which extends the complex space adopted by existing models(Zhang et al. 2022; Lacroix, Obozinski, and Usunier 2020) to an attention-based product space. We introduce the key components of our temporal knowledge graph embedding method, HGE, in the following order: a) *embedding space*, b) *temporal-relational attention*, c) *temporal-geometric attention*. Figure 2 shows the structure of our proposed HGE model.
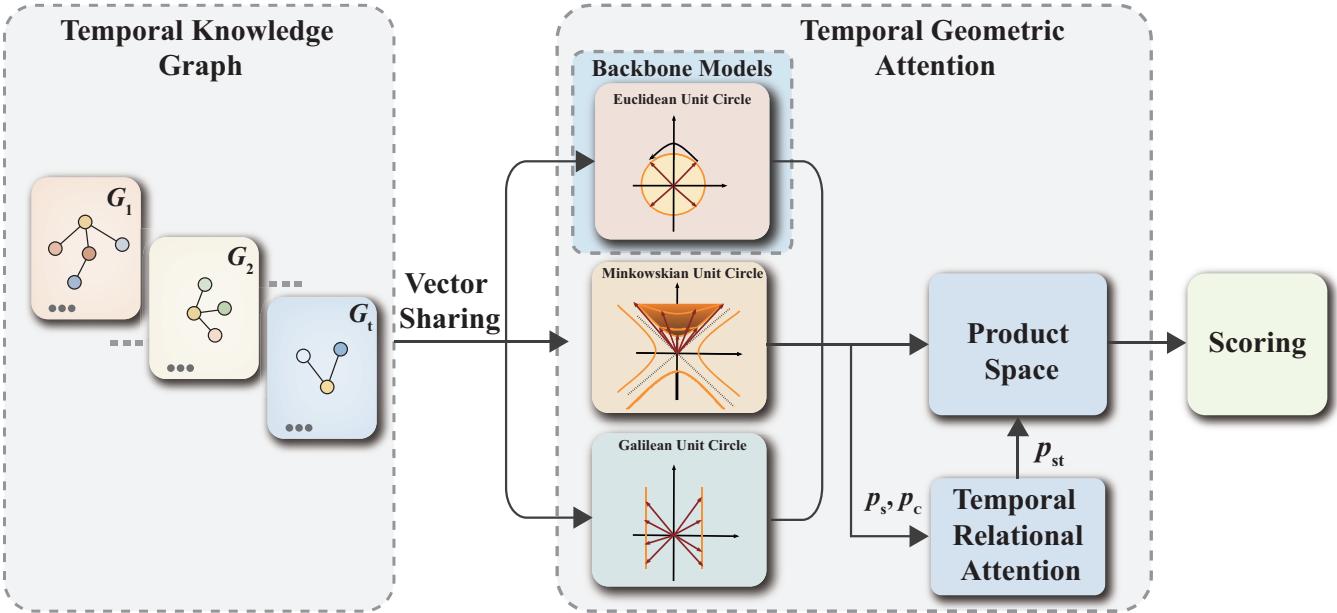
Figure 2: An illustration for the HGE. At first, entities, relations and timestamps in temporal knowledge graphs are represented in heterogeneous geometric subspaces: 1) complex space, 2) split-complex space, 3) dual space respectively. Based on the static relation embedding p_s, and dynamic relation embedding p_c, temporal relational attention learns hybrid relation embedding p_{st} based on each relation's changing frequencies. Temporal geometric attention incorporates embeddings in geometric subspaces into a product space by p_{st}, which decides the suitable geometry for each relation. Finally, the scoring function is performed on the embeddings learned in the product space.

3.1 Embeddings in Geometric Subspaces

We aim to embed the elements of a temporal knowledge graph (entities, relations, and times) into a d dimensional product space $\mathcal{M} = \mathbb{M}_1 \times \ldots \times \mathbb{M}_d$ where each $\mathbb{M}_i$ is a Complex, Dual or Split-complex space, i.e. $\mathbb{M}_i \in \{\mathbb{C}, \mathbb{S}, \mathbb{D}\}$. For a given fact $(s, p, o, \tau) \in G$, we use the mappings $f_e : \mathcal{E} \to \mathbb{M}_i, f_r : \mathcal{R} \to \mathbb{M}_i, f_\tau : \mathcal{T} \to \mathbb{M}_i$ to assign d dimensional vectors to each element of a TKG as $s_{\mathbb{M}_i}, p_{\mathbb{M}_i}, o_{\mathbb{M}_i}, \tau_{\mathbb{M}_i}$ respectively.

We introduce the three fundamental parts of the product space for developing our model, namely Complex, Split-complex and Dual spaces together with their geometric interpretations. Given a quadratic formula $k^2 + g = 0, g = \{-1, 1, 0\}$, we have the three number systems based on the value of g:

Complex Vector Space Complex numbers (Harkin and Harkin 2004; Helzer 2000) allow for solving the quadratic formula $k^2 + 1 = 0$, by defining a new number $k = i$ where $i^2 = -1$. i is used to define the set of Complex numbers $\mathbb{C} = \{q = a + bi | a, b \in \mathbb{R}, i^2 = -1\}$, where a is the real and b the imaginary part. The multiplication of two Complex numbers $q_1 = a + bi, q_2 = c + di$ is defined by $q_1 * q_2 = (ac - bd) + (ad + bc)i$. It has been proved by previous works (Zhang et al. 2022; Lacroix, Obozinski, and Usunier 2020; Xu et al. 2020) to represent temporal knowledge graphs effectively. Following their work, we represent s, p, o, τ in complex space as:

$$s_{\mathbb{C}} = s_{\mathbb{C}a} + s_{\mathbb{C}b}i, p_{\mathbb{C}} = p_{\mathbb{C}a} + p_{\mathbb{C}b}i, \\ o_{\mathbb{C}} = o_{\mathbb{C}a} + o_{\mathbb{C}b}i, \tau_{\mathbb{C}} = \tau_{\mathbb{C}a} + \tau_{\mathbb{C}b}i \tag{1}$$

where $s_{\{.\}}, p_{\{.\}}, o_{\{.\}}, \tau_{\{.\}} \in \mathbb{R}^d$. $\{.\}_a$ represents the real part of each element and $\{.\}_b$ represents the imaginary part.

Split-complex Vector Space Dealing with quadratic formula $k^2 - 1 = 0$, a split-complex number (Harkin and Harkin 2004; Helzer 2000) is defined as $p = a + jb$, where $k = j, j^2 = 1, j \neq 1, -1$. Formally the space of split-complex number is defined as $\mathbb{S} = \{q = a + bj | a, b \in \mathbb{R}, j^2 = 1, j \neq 1, -1\}$. a, b are real and split parts, respectively. The multiplication of two Split-Complex numbers $q_1 = a + bj, q_2 = c + dj$ is defined by $q_1 * q_2 = (ac + bd) + (ad + bc)j$. We represent s, p, o, τ in split-complex space as:

$$s_{\mathbb{S}} = s_{\mathbb{S}a} + s_{\mathbb{S}b}j, p_{\mathbb{S}} = p_{\mathbb{S}a} + p_{\mathbb{S}b}j, \\ o_{\mathbb{S}} = o_{\mathbb{S}a} + o_{\mathbb{S}b}j, \tau_{\mathbb{S}} = \tau_{\mathbb{S}a} + \tau_{\mathbb{S}b}j, \tag{2}$$

Dual Vector Space Dual numbers (Angeles 1998; Helzer 2000) are similar to Complex numbers, but their imaginary ϵ is defined such that $\epsilon^2 = 0, \epsilon \neq 0$. The dual space is then defined as $\mathbb{D} = \{q = a + b\epsilon | a, b \in \mathbb{R}, \epsilon^2 = 0, \epsilon \neq 0\}$ where a, b are real and dual components of the dual numbers. The multiplication of two Dual numbers $q_1 = a + b\epsilon, q_2 = c + d\epsilon$ is defined by $q_1 * q_2 = (ac) + (ad + bc)\epsilon$. We represent s, p, o, τ in dual space as:

$$s_{\mathbb{D}} = s_{\mathbb{D}a} + s_{\mathbb{D}b}\epsilon, p_{\mathbb{D}} = p_{\mathbb{D}a} + p_{\mathbb{D}b}\epsilon,$$
$$o_{\mathbb{D}} = o_{\mathbb{D}a} + o_{\mathbb{D}b}\epsilon, \tau_{\mathbb{D}} = \tau_{\mathbb{D}a} + \tau_{\mathbb{D}b}\epsilon \tag{3}$$

3.2 Embeddings in Attention-based Product Space

How to fuse information from different subspaces into a product space efficiently remains a challenging task in the knowledge graph embedding task. Existing work (Han et al. 2020) assigns different dimensions d_i for each subspace $\mathbb{M}_i$, where $\sum d_i = d$, and calculates their individual loss which will be aggregated subsequently to a total loss. Such a stacking strategy requires the manual selection of suitable d_i numbers for every new task and consumes huge computation resources to reach optimal d_i decision. To capture suitable geometries from various subspaces efficiently, we introduce an attention-based product space. Rather than stacking ad hoc vectors for each subspace, our method reuses vectors for every subspace and aggregates *Scoring Vectors* of subspaces by relational and temporal information.

Real and Imaginary Vector Sharing   Existing methods (Han et al. 2020) assigns different vectors for each subspaces. However, pre-experiments in Appendix C illustrate that although their geometric interpretations are diverse, real and imaginary vectors in different subspaces are almost unanimous when trained to optimal settings with the same embedding sizes. Accordingly, we share the real and imaginary vectors between all subspaces as follows:

$$\{.\}_{\mathbb{C}a} = \{.\}_{\mathbb{S}a} = \{.\}_{\mathbb{D}a}$$
$$\{.\}_{\mathbb{C}b} = \{.\}_{\mathbb{S}b} = \{.\}_{\mathbb{D}b} \tag{4}$$

where $\{.\} \in \{s, p, o, \tau\}$. With the reusing strategy, our method avoids the manual selection of subspace dimensions and saves embedding space. If not specified, we use $s = [s_a, s_b]$, $p = [p_a, p_b]$, $o = [o_a, o_b]$ and $\tau = [\tau_a, \tau_b]$ to represent embeddings a generic geometric subspace in the following section for simplicity.

Temporal-relational Attention   Relations in TKGs may exhibit different frequencies of change varying from fully static to quickly changing behavior (Lacroix, Obozinski, and Usunier 2020). For example, the relation *capitalOf* is not changing often over time, while the relation *isPresidentOf* exhibits more frequent changes. Therefore, for each relation p, we provide two vectors $p_s, p_c \in \mathcal{M}$. The first captures the static behavior and the second captures the dynamic behavior by multiplication with time embedding τ_τ. We provide a temporal attention mechanism to emphasize static or dynamic behavior depending on the characteristics of the relation:

$$p_{s\tau} = \alpha_\tau (p_c * \tau_\tau) + \alpha_s p_s$$
$$(\alpha_\tau, \alpha_s) = \text{Softmax} (\mathbf{w}_p (p_c * \tau_\tau), \mathbf{w}_p p_s) \tag{5}$$

where $\mathbf{w}_p$ is the relation-specific weight.

Scoring Vectors from Subspaces   We take all values in each subspace for entities, relations, and times $s_i, p_{s\tau i}, o_i \in \mathbb{M}_i$ and compute $c_i = \langle s_i, p_{s\tau i}, o_i \rangle^1$, where $\langle , , \rangle$ is the product in Complex, Split-complex and Dual spaces computed as follows:

$$c_{\mathbb{C}} = \langle (s_a p_{s\tau a} - s_b p_{s\tau b}) + (s_a p_{s\tau b} + s_b p_{s\tau a})i, o_a + io_b \rangle$$
$$= (s_a p_{s\tau a} o_a - s_b p_{s\tau b} o_a - s_a p_{s\tau b} o_b - s_b p_{s\tau a} o_b) +$$
$$(s_a p_{s\tau a} o_b - s_b p_{s\tau b} o_b + s_a p_{s\tau b} o_a + s_b p_{s\tau a} o_a)i,$$
$$c_{\mathbb{S}} = \langle (s_a p_{s\tau a} + s_b p_{s\tau b}) + (s_a p_{s\tau b} + s_b p_{s\tau a})j, o_a + jo_b \rangle$$
$$= (s_a p_{s\tau a} o_a + s_b p_{s\tau b} o_a + s_a p_{s\tau b} o_b + s_b p_{s\tau a} o_b) +$$
$$(s_a p_{s\tau a} o_b + s_b p_{s\tau b} o_b + s_a p_{s\tau b} o_a + s_b p_{s\tau a} o_a)j,$$
$$c_{\mathbb{D}} = \langle (s_a p_{s\tau a}) + (s_a p_{s\tau b} + s_b p_{s\tau a})\epsilon, o_a + \epsilon o_b \rangle$$
$$= (s_a p_{s\tau a} o_a) + (s_a p_{s\tau a} o_b + s_a p_{s\tau b} o_a + s_b p_{s\tau a} o_a)j. \tag{6}$$

Temporal-geometric Attention   Scoring vectors represent distinctive geometric information captured by each subspace. We propose a temporal-geometric attention mechanism to integrate them based on current relational and time information.

$$\beta_i = \text{Softmax} (p_{s\tau} c_i), \ i \in \{\mathbb{C}, \mathbb{D}, \mathbb{S}\}. \tag{7}$$

It emphasizes the most suitable geometry for each query via the augmented relation embedding p_{st}. As the changing frequencies of relations could be reflected by p_{st}, HGE could model the static and dynamic logical and structural patterns in TKGs. The overall score aggregates the inner product in all subspaces:

$$S_{\mathcal{M}}(s, r, o, \tau) = \sum_{i=1}^{d} \beta_i c_i, \tag{8}$$

It's worth noting that new geometric subspaces could be easily incorporated into Equation 8 given shared real and imaginary vectors and appropriate scoring vectors.

4   Theoretical Analysis on Temporal Patterns

Knowledge graphs exhibit *patterns*. A *structural pattern* is a regularity in the graph, e.g. a tree as given in the middle of Figure 1, that may or may not allow for logical conclusions, but which may be hard to represent in some embedding methods. A *logical pattern* represents a rule that allows for concluding new facts when applied to given facts. For instance, *(Charles,marriedWith,Camilla)* implies *(Camilla,marriedWith,Charles)* because *marriedWith* is symmetric.

Embeddings for temporal knowledge graphs must account for temporal facts including time components and express corresponding *temporal patterns*. Four kinds of logical patterns, *symmetric, inverse, asymmetric* and *evolve* are mostly considered and studied in existing TKGE models (Chen et al. 2022; Xu et al. 2020). However, their definitions either neglect time information or merely consider patterns

---

[1]Similar to previous work(Xu et al. 2020; Lacroix, Obozinski, and Usunier 2019), we adopt conjugate on o_i to increase the performance in experiments.

when facts happen at the same time. We generalize and go beyond these approaches and consider *static temporal patterns* and *dynamic temporal patterns*. If a structural or a logical temporal pattern holds *regardless of time information* as in traditional knowledge graphs, we call it a *static temporal pattern*. If a structural or a logical temporal pattern represents or draws conclusions *using time information*, we call it a *dynamic temporal pattern*.

In the following, we will formally define a few temporal patterns. For simplicity, we only illustrate the occasion when τ is a time interval. However, it's convenient to extend the following definitions when τ is a time point. Examples of each definition are indicated after "//".

4.1 Static Logical Temporal Patterns

Definition 3. *A temporal relation p is symmetric at all points in time iff $\forall s, o, \tau : (s, p, o, \tau) \rightarrow (o, p, s, \tau)$.* // *marriedWith*
A temporal relation p is anti-symmetric at all points in time iff $\forall s, o, \tau : (s, p, o, \tau) \rightarrow \neg(o, p, s, \tau)$. // *locatedIn*

Definition 4. *A temporal relation p_1 is the inverse of temporal relation p_2 at all points in time iff $\forall s, o, \tau : (s, p_1, o, \tau) \rightarrow (o, p_2, s, \tau)$.* // *advises, advisedBy*

4.2 Dynamic Logical Temporal Patterns

Definition 5. *A temporal relation p is temporal symmetric iff $\forall s, o, \tau_1 : \exists \tau_2 : (s, p, o, \tau_1) \rightarrow (o, p, s, \tau_2)$.* // *consults*
A temporal relation p is temporal anti-symmetric iff $\forall s, o : \exists \tau_1 : (s, p, o, \tau_1) \rightarrow \forall \tau_2 \neg(o, p, s, \tau_2)$. // *arrest*

Definition 6. *A relation p_1 at time τ_1 is the temporal inverse of relation p_2 at time τ_2 iff $\forall s, o : \exists \tau_1, \tau_2 : (s, p_1, o, \tau_1) \rightarrow (s, p_2, o, \tau_2)$.* // *invitesToVisit, Visit*

Definition 7. *Relation p_1 evolves into relation p_2 iff $\forall s, o : \exists \tau_1, \tau_2 : Precedes(\tau_1, \tau_2) \ \& \ (s, p_1, o, \tau_1) \rightarrow (s, p_2, o, \tau_2)$.* // *engagedWith, marriedWith*

Definition 8. *Relation p is temporary in time iff $\forall s, o, \tau_1 : (s, p, o, \tau_1) \rightarrow \exists \tau_0, \tau_2 : Precedes(\tau_0, \tau_1) \ \& \ Precedes(\tau_1, \tau_2) \ \& \ \neg(s, p, o, \tau_0) \ \& \ \neg(s, p, o, \tau_2)$.* //- *worksFor*

4.3 Modeling Temporal Patterns

We present a theoretical analysis corresponding to the ability of our method in modeling various temporal patterns introduced in 4 as follows: (See details in Appendix F)

Proposition 1. *HGE can model (anti-)symmetry and temporal (anti-)symmetry in Definitions 3 and 5.*

Proposition 2. *HGE can model inverse and temporal inverse patterns in Definitions 4 and 6.*

Proposition 3. *HGE can model evolves pattern in Definition 7.*

Proposition 4. *HGE can model temporary relations in Definition 8.*

5 Experiments

5.1 Experimental Settings

Dataset To evaluate the effectiveness of the proposed attention-based product space embedding, we perform the link prediction task on four popular temporal knowledge graph benchmark datasets, i.e. ICEWS14 (Garcia-Duran, Dumančić, and Niepert 2018), ICEWS05-15 (Garcia-Duran, Dumančić, and Niepert 2018), GDELT (Trivedi et al. 2017) and Wikidata12k (Lacroix, Obozinski, and Usunier 2020). ICEWS14 and ICEWS05-15 are two subset datasets from the Integrated Conflict EarlyWarning System (ICEWS)(Lautenschlager, Shellman, and Ward 2015), which contain news facts in 2014 and between 2005 and 2015 respectively. The Global Database of Events, Language, and Tone (GDELT) is a large knowledge graph that describes facts about human behaviors. We adopt the same data subset as (Gao et al. 2020), which uses the subset of facts from April 1, 2015 to March 31, 2016. Compared to other datasets, GDELT contains fewer temporal relations but more quadruples, which makes it the densest dataset concerning temporal information. Wikidata12k is a subset of wikidata dump (Erxleben et al. 2014). It represents the time information $\tau \in \mathcal{T}$ as time intervals, in which m or n could be empty, referring to intervals $(-\infty, n]$ or $[m, \infty)$. Table 5 summarises the statistics of four datasets.

Backbone and Baseline Models Our proposed model, HGE, aims to generalize complex-space-based TKGE models to an attention-based product space of heterogeneous geometric subspaces. Hence, we choose several state-of-the-art complex-space-based TKGE models as HGE's backbone models to validate its effectiveness. **TeRo** (Xu et al. 2020) defines the evolution of entity embeddings from the initial state to the current time as a rotation in complex vector space. **TComplEx** and **TNTComplEx** (Lacroix, Obozinski, and Usunier 2020) models temporal knowledge graph completion as an order 4 tensor completion problem. **TLT-KGE** (Zhang et al. 2022) models semantic information and temporal information as different parts of complex space or quaternion space. Complex or quaternion operations exchange information between different parts.

To give a comprehensive overview, we also compare our model with non-complex space temporal knowledge graph embedding baselines TTransE (Garcia-Duran, Dumančić, and Niepert 2018), TA-DistMult (Leblay and Chekol 2018), RotateQVS(Chen et al. 2022), BoxTE (Messner, Abboud, and Ceylan 2022), and LCGE(Niu and Li 2023)[2].

Evaluation Metrics We adopt the link prediction task to evaluate our proposed model. Link prediction infers the missing entities for incomplete facts. During the test step, we follow the procedure of (Xu et al. 2020) to generate candidate quadruples. From a test quadruple (s, p, o, τ), we replace s with $\bar{s} \in \mathcal{E}$ and o with $\bar{o} \in \mathcal{E}$ to get candidate quadruples $(s, p, \bar{o}, \tau) \cup (\bar{s}, p, o, \tau)$. If τ is a time interval $[m, n]$, we sample a time point (appearing in the dataset) uniformly at random, in the range $[m, n]$ as (Lacroix, Obozin-

---

[2]We notice some inconsistent inference issues in LCGE's original code. Please refer to Appendix J for detailed discussions.

ski, and Usunier 2019). When m or n is empty, we set it as the first or last time point of the dataset. All candidate quadruples will be ranked by their scores using a time-aware filtering strategy (Goel et al. 2020). We evaluate our models with four metrics: Mean Reciprocal Rank (MRR), the mean of the reciprocals of predicted ranks of correct quadruples, and Hits@(1/3/10), the percentage of ranks not higher than 1/3/10. For all experiments, the higher the better.

To have a fair comparison, we set entity and relation embedding dimension sizes as reported in the original papers. For TeRo-based models, we set the dimension size of d as 500 on four benchmark datasets. For TComplEx-based, TNTComplEx-based, and TLT-KGE-based models, we set the dimension size of d as 1200, 1200, 1500 and 2000 on ICEWS14, ICEWS05-15, GDELT and Wikidata12k respectively. The training epoch is set to 200. We adopt the same regularizer, loss function, and negative sampling size as reported in the original papers[3].

5.2 HGE's Performance Comparison

We evaluate HGE's performance gain on four datasets. Table 1 shows the performances of the original backbones and backbones plugged with HGE on time point datasets ICEWS14, ICEWS05-15, and GDELT. From Table 1, we have the following observations:

(i) HGE can provide significant improvements over chosen backbones consistently on all datasets, which verifies the effectiveness of the proposed HGE module.

(ii) We observe the proposed method is more effective on the dense dataset GDELT. GDELT provides more instances for each relation-timestamp pair. We conjecture it benefits the temporal-geometric attention mechanism, in which fine-grained geometric attention is influenced by both relational and temporal information. Conversely, ICEWS05-15 is the sparsest dataset. As a result, HGE dose not greatly improve the performance of backbones on ICEWS05-15 and even decreases TeRo's performance.

(iii) We find that HGE achieves greater performance gains for TNTComplEx and TComplEx than for TLT-KGE. As the TLT-KGE model provides interactions between time information and relation information in complex numbers, we believe it substitutes the function of the temporal-relation attention mechanism to some degree. However, Table 7 in Appendix presents that TNTComplEx+HGE reaches comparable results as TLT-KGE with only half parameter numbers, which demonstrates the proposed temporal-relation attention mechanism is more efficient to combine time and relation information.

Table 2 shows link prediction results on the time interval dataset. With HGE, all metrics get improvement, reflecting HGE could boost the performance of backbones on different kinds of TKGs.

5.3 Ablation Study

We conduct ablation study experiments on backbone TNT-ComplEx to investigate the effectiveness of each compo-

---

[3]The code, details of training and appendix are provided in https://github.com/NacyNiko/HGE
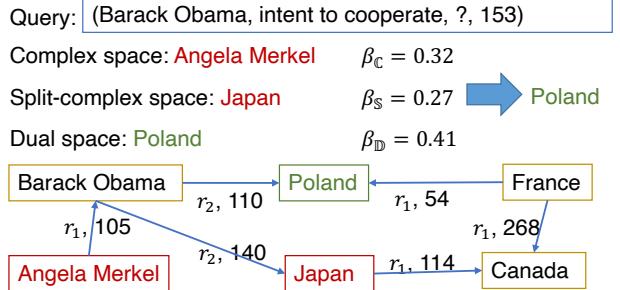
Figure 3: A case study of HGE model. We omit some entities connected to France by relation r_1 which forms a temporal star structure for brevity. r_1 stands for *intent to cooperate* relation and r_2 stands for *consult* relation. Time information is shown in ids.

nent. From Table 3, we have the following observations:

(i) Our proposed subspace integration strategy achieves higher performance than the stacking strategy introduced by (Han et al. 2020). We find out individual loss for each subspace in TNTComplEx+stack becomes unbalanced during training time. We conjecture the model may pay too much attention to optimizing the unsuitable geometry subspaces for certain facts and hamper further improvement.

(ii) We observe that the temporal-relation attention mechanism contributes more performance gain on GDELT. GDELT is a dense dataset and has more facts for the enumeration of objects of relation types and timestamps than other datasets. We conjecture it benefits from the fine-grained geometric attention mechanism in which the attention weights are influenced by both relation type and timestamps.

(iii) We find that the temporal-geometric attention mechanism is more effective on ICEWS14 and ICEWS05-15 datasets. Compared to GDELT, they contain more relation types and thus provide a wider variety of relational structural patterns in the datasets. This illustrates the importance of introducing heterogeneous geometric spaces in HGE to represent the diverse structure in temporal knowledge graphs.

5.4 Case Study

Intent to cooperate relation forms a temporal-star structure in TKGs as the head entity could express this attitude to multiple tail entities. In Figure 3, on account of the query (Barack Obama, intent to cooperate, ?, 153), complex space predicts the wrong answer *Angela Merkel* as it supposes a symmetric instance exists for *(Angela Merkel, r_1, Barack Obama, 105)*. Split-complex space predicts the wrong answer *Japan* to form a hierarchy path between *Angela Merkel, Barack Obama* and *Japan*. Dual space predicts the correct answer *Poland* as it has been the object entity in the temporal star structure formed by *France*. Given that *Barack Obama* consults *Japan* recently, HGE chooses the correct answer *Poland* with the help of the temporal-geometric attention mechanism.

Table 1: Link prediction results on ICEWS14, ICEWS05-15, and GDELT. The best results among all models are in **bold**. Additionally, we <u>underline</u> the best results among models with the same backbone model.

Model	ICEWS14				ICEWS05-15				GDELT			
	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10
TTransE	25.5	7.4	-	60.1	27.1	8.4	-	61.6	11.5	0	16.0	31.8
TADistMult	47.7	36.3	-	68.6	47.4	34.6	-	72.8	20.6	12.4	21.9	36.5
RotateQVS	59.1	50.7	64.2	75.4	63.3	52.9	70.9	81.3	27.0	17.5	29.3	45.8
BoxTE(k=2)	61.5	53.2	66.7	76.7	66.4	57.6	72.0	82.2	33.9	25.1	36.6	50.7
LCGE	61.6	53.2	66.7	77.5	61.8	51.4	68.1	81.2	-	-	-	-
TeRo	56.2	46.8	62.1	73.2	<u>58.6</u>	<u>46.9</u>	<u>66.8</u>	79.5	23.2	14.5	24.9	30.9
TeRo**+HGE**	<u>58.6</u>	<u>49.5</u>	<u>64.5</u>	<u>74.9</u>	57.8	45.3	66.5	<u>80.4</u>	<u>23.4</u>	<u>14.7</u>	<u>25.2</u>	<u>40.5</u>
△ Improve	4.3%	5.8%	3.9%	1.4%	-1.5%	-3.4%	-0.1%	1.1%	0.9%	1.4%	1.2%	31.1%
TComplEx	61.9	54.2	66.1	76.7	66.5	58.3	71.6	81.1	34.6	25.9	37.2	51.5
TComplEx**+HGE**	<u>62.6</u>	<u>54.7</u>	<u>67.2</u>	<u>77.4</u>	<u>67.2</u>	<u>59.3</u>	<u>72.0</u>	<u>81.7</u>	<u>36.8</u>	<u>27.4</u>	<u>40.1</u>	<u>55.3</u>
△ Improve	1.1%	0.9%	1.7%	0.9%	1.1%	1.7%	0.6%	0.7%	5.2%	5.8%	7.8%	7.4%
TNTComplEx	60.7	51.9	65.9	77.2	66.6	58.3	71.8	81.7	34.1	25.2	36.8	51.5
TNTComplEx**+HGE**	<u>63.0</u>	**55.1**	<u>67.5</u>	<u>78.0</u>	<u>68.1</u>	<u>60.1</u>	<u>72.9</u>	<u>82.9</u>	**37.1**	**28.3**	<u>40.0</u>	<u>54.1</u>
△ Improve	3.7%	6.2%	2.4%	0.6%	2.3%	3.1%	1.5%	1.5%	8.8%	12.3%	8.7%	5.0%
TLT-KGE	63.0	54.9	67.8	77.7	68.6	60.7	73.5	83.1	35.6	26.7	38.5	53.2
TLT-KGE**+HGE**	**63.4**	<u>55.0</u>	**68.5**	**78.8**	**68.8**	**60.8**	**74.0**	**83.5**	**37.1**	<u>27.7</u>	**40.2**	**55.6**
△ Improve	0.6%	0.1%	1.0%	1.4%	0.3%	0.2%	1.4%	0.5%	4.2%	3.7%	4.4%	3.0%

Table 2: Link Prediction results on Wikidata12k.

Model	MRR$[a,b]$	MRR$[a,\infty)$	MRR$(-\infty,b]$
TNTComplEx	27.4	37.8	51.7
TNTComplEx +HGE	28.4	37.8	57.0
TLT-KGE	27.0	36.0	48.0
TLT-KGE +HGE	27.4	37.7	51.7

Table 3: MRR performance of HGE components. +tra stands for merely using temporal-relational attention mechanism. +tga stands for merely using temporal-geometric attention mechanism. +stack stands for integrating subspaces with the stacking strategy in (Han et al. 2020)

Model	ICEWS14	ICEWS05-15	GDELT
TNTComplEx+HGE	63.0	68.1	37.1
TNTComplEx	60.7	66.6	34.1
TNTComplEX+stack	62.0	67.3	35.6
TNTComplEx+tra	62.0	67.4	36.9
TNTComplEx+tga	62.6	67.5	36.4

6 Related Works

TKGE models incorporate time information in different ways. TTransE (Leblay and Chekol 2018) and TA-DistMult (Garcia-Duran, Dumančić, and Niepert 2018) insert the time information into different score functions as another element. TeRo (Xu et al. 2020) defines the temporal evolution of entity embeddings as a rotation from the initial time to the current time in complex vector space.

T(NT)ComplEx (Lacroix, Obozinski, and Usunier 2019) is a semantic matching approach that models temporal knowledge graph completion as an order 4 tensor completion problem. TeLM (Xu et al. 2021) also performs 4th-order tensor factorization on temporal knowledge graphs but adds a bias component between the neighboring temporal embeddings in the temporal regularizer. Moreover, it adopts multi-vector embeddings for entities, relations, and timestamps.

Inspired by TeRo (Xu et al. 2020), RotateQVS (Chen

et al. 2022) embeds entities in quaternion space and temporal changes are represented as rotations. BoxTE (Messner, Abboud, and Ceylan 2022) extends BoxE (Abboud et al. 2020) by including relation-specific time embeddings. TLT-KGE (Zhang et al. 2022) models semantic information and temporal information as different parts of complex space or quaternion space. Complex or quaternion operations exchange information between different parts. LCGE (Niu and Li 2023) use temporal rules to regularize entity embedding and adopts commonsense reasoning as the extra learning task. Most of the reviewed TKGE approaches model temporal patterns by using a single geometry, and do not present multiple geometries to capture diverse temporal patterns.

Several manifold-based TKGE models have been proposed in (Montella, Barahona, and Heinecke 2021; Han et al. 2020). (Montella, Barahona, and Heinecke 2021) is an extension of AttH (Chami et al. 2020) to temporal KGEs which use hyperbolic manifolds as embedding space. It only uses a single geometry for embedding space. (Han et al. 2020) embeds TKGs into a product space of several manifolds to model multiple structural patterns. However, it does not select the most suitable manifold depending on structural patterns existing in TKGs but chooses it manually.

7 Conclusion

We present HGE, a new temporal KGE model that utilizes multiple geometries. HGE extends state-of-the-art TKGEs from a Complex space to the product space that embeds temporal facts in Complex, Split-complex, and Dual subspaces via two temporal attention mechanisms. The temporal-relational attention mechanism captures relations with varying change frequencies. The temporal geometric attention mechanism fuses information from different geometries according to the captured relational and temporal information. Extensive experiments on benchmark datasets validate that our model uniformly improves several state-of-the-art Complex-based TKGE models. In the future, we plan to include more types of heterogeneous geometric spaces.

Acknowledgments

References

Abboud, R.; Ceylan, I.; Lukasiewicz, T.; and Salvatori, T. 2020. Boxe: A box embedding model for knowledge base completion. *Advances in Neural Information Processing Systems*, 33: 9649–9661.

Allen, J. F. 1983. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11): 832–843.

Angeles, J. 1998. The application of dual algebra to kinematic analysis. In *Computational methods in mechanical systems*, 3–32. Springer.

Balazevic, I.; Allen, C.; and Hospedales, T. 2019. Multi-relational poincaré graph embeddings. *Advances in Neural Information Processing Systems*, 32.

Chami, I.; Wolf, A.; Juan, D.-C.; Sala, F.; Ravi, S.; and Ré, C. 2020. Low-Dimensional Hyperbolic Knowledge Graph Embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 6901–6914.

Chen, K.; Wang, Y.; Li, Y.; and Li, A. 2022. RotateQVS: Representing Temporal Information as Rotations in Quaternion Vector Space for Temporal Knowledge Graph Completion. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 5843–5857.

Erxleben, F.; Günther, M.; Krötzsch, M.; Mendez, J.; and Vrandečić, D. 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 13*, 50–65. Springer.

Gao, C.; Sun, C.; Shan, L.; Lin, L.; and Wang, M. 2020. Rotate3d: Representing relations as rotations in three-dimensional space for knowledge graph embedding. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, 385–394.

Garcia-Duran, A.; Dumančić, S.; and Niepert, M. 2018. Learning Sequence Encoders for Temporal Knowledge Graph Completion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 4816–4821.

Goel, R.; Kazemi, S. M.; Brubaker, M.; and Poupart, P. 2020. Diachronic embedding for temporal knowledge graph completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 3988–3995.

Han, Z.; Chen, P.; Ma, Y.; and Tresp, V. 2020. DyERNIE: Dynamic Evolution of Riemannian Manifold Embeddings for Temporal Knowledge Graph Completion. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 7301–7316.

Harkin, A. A.; and Harkin, J. B. 2004. Geometry of generalized complex numbers. *Mathematics magazine*, 77(2): 118–129.

Helzer, G. 2000. Special relativity with acceleration. *The American Mathematical Monthly*, 107(3): 219–237.

Hogan, A.; Blomqvist, E.; Cochez, M.; de Melo, G.; Gutierrez, C.; Kirrane, S.; Labra Gayo, J. E.; Navigli, R.; Neumaier, S.; Ngonga Ngomo, A.-C.; et al. 2021. Knowledge Graphs. *ACM Computing Surveys*, 54(4): 1–37.

Lacroix, T.; Obozinski, G.; and Usunier, N. 2019. Tensor Decompositions for Temporal Knowledge Base Completion. In *International Conference on Learning Representations*.

Lacroix, T.; Obozinski, G.; and Usunier, N. 2020. Tensor Decompositions for temporal knowledge base completion.

Lautenschlager, J.; Shellman, S.; and Ward, M. 2015. Icews event aggregations. *Harvard Dataverse*, 3(595): 28.

Leblay, J.; and Chekol, M. W. 2018. Deriving validity time in knowledge graph. In *Companion Proceedings of the The Web Conference 2018*, 1771–1776.

Messner, J.; Abboud, R.; and Ceylan, I. I. 2022. Temporal knowledge graph completion using box embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 7779–7787.

Montella, S.; Barahona, L. M. R.; and Heinecke, J. 2021. Hyperbolic Temporal Knowledge Graph Embeddings with Relational and Time Curvatures. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 3296–3308.

Niu, G.; and Li, B. 2023. Logic and Commonsense-Guided Temporal Knowledge Graph Completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 4569–4577.

Ren, X.; Bai, L.; Xiao, Q.; and Meng, X. 2023. Hierarchical Self-Attention Embedding for Temporal Knowledge Graph Completion. In *Proceedings of the ACM Web Conference 2023*, 2539–2547.

Singh, I.; Kaur, N.; Gaur, G.; and Mausam. 2023. NeuSTIP: A Novel Neuro-Symbolic Model for Link and Time Prediction in Temporal Knowledge Graphs. arXiv:2305.11301.

Trivedi, R.; Dai, H.; Wang, Y.; and Song, L. 2017. Know-evolve: Deep temporal reasoning for dynamic knowledge graphs. In *international conference on machine learning*, 3462–3471. PMLR.

Xu, C.; Chen, Y.-Y.; Nayyeri, M.; and Lehmann, J. 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*, 2569–2578.

Xu, C.; Nayyeri, M.; Alkhoury, F.; Yazdi, H. S.; and Lehmann, J. 2020. TeRo: A Time-aware Knowledge Graph Embedding via Temporal Rotation. In *Proceedings of the 28th International Conference on Computational Linguistics*, 1583–1593.

Zhang, F.; Zhang, Z.; Ao, X.; Zhuang, F.; Xu, Y.; and He, Q. 2022. Along the Time: Timeline-traced Embedding for

Temporal Knowledge Graph Completion. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2529–2538.

A Allen's Relations

The definitions of 13 Allen's Relations between two-time intervals $\tau_1 = [m_1, n_1], \tau_2 = [m_2, n_2]$ are defined in Figure 4.
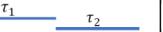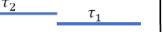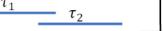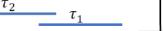
Allen's Relations	Pictoral Example	Chronological Sequence
Precedes(τ_1, τ_2)		$n_1 < m_2$
Preceded_by(τ_1, τ_2)		$n_2 < m_1$
Meets(τ_1, τ_2)		$n_1 = m_2$
Met_by(τ_1, τ_2)		$n_2 = m_1$
Overlaps(τ_1, τ_2)		$m_1 < m_2 < n_1 < n_2$
Overlapped_by(τ_1, τ_2)		$m_2 < m_1 < n_2 < n_1$
Starts(τ_1, τ_2)		$m_1 = m_2 < n_1 < n_2$
Started_by(τ_1, τ_2)		$m_1 = m_2 < n_2 < n_1$
During(τ_1, τ_2)		$m_2 < m_1 < n_1 < n_2$
Contains(τ_1, τ_2)		$m_1 < m_2 < n_2 < n_1$
Finishes(τ_1, τ_2)		$m_1 < m_2 < n_1 = n_2$
Finished_by(τ_1, τ_2)		$m_2 < m_1 < n_1 = n_2$
Equal(τ_1, τ_2)		$m_1 = m_2 < n_1 = n_2$

Figure 4: 13 relations in Allen algebra calculus.

B Extending Backbones to HGE methods

TeRo TeRo (Xu et al. 2020) represents s, p, o, τ in Complex space as Equation 1. Similarly, we represent s, p, o, τ in Split-complex space and Dual space as Equation 2 and 3. Following (Xu et al. 2020), we represent s_t and o_t as:

$$s_{t\mathbb{M}_i} = s_{\mathbb{M}_i} \circ t_{\mathbb{M}_i}, o_{t\mathbb{M}_i} = o_{\mathbb{M}_i} \circ t_{\mathbb{M}_i} \qquad (9)$$

For temporal relational attention, we set $p = p_s = p_c$, so the dynamic relational information is captured by $p * t$.

TLT-KGE TLT-KGE (Zhang et al. 2022) represents s, p, o, τ in Complex space as:

$$s_{\mathbb{C}} = e_s + t_\tau^e i, p_{\mathbb{C}} = r_p + t_\tau^r i, o_{\mathbb{C}} = e_o + t_\tau^e i, \qquad (10)$$

Similarly, we represent s, p, o, τ in Dual space and Split-complex space as:

$$\begin{aligned} s_{\mathbb{D}} = e_s + t_\tau^e \epsilon, p_{\mathbb{C}} = r_p + t_\tau^r \epsilon, o_{\mathbb{C}} = e_o + t_\tau^e \epsilon, \\ s_{\mathbb{S}} = e_s + t_\tau^e j, p_{\mathbb{S}} = r_p + t_\tau^r j, o_{\mathbb{S}} = e_o + t_\tau^e j, \end{aligned} \qquad (11)$$

For temporal relational attention, we adopt r_p and r_{compr} in Equation 12 of original paper as p_s, and $p_c * \tau_\tau$ respectively, where $r_{compr} = r_p * t_{compr}$.

C Embeddings in Complex, Dual and Split-Complex Subpaces

Keeping other settings fixed, we train 3 model variants TNTComplEx+complex, TNTComplEx+split, TNTComplEx+dual which use a single geometric space to optimal MRR scores on ICEWS14. We randomly select 100

entities from the entity set and analyze the similarity of their embeddings on different geometric spaces by cosine similarity:

$$S_C(A, B) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^n A_i B_i}{\sqrt{\sum_{i=1}^n A_i^2 \cdot \sum_{i=1}^n B_i^2}} \qquad (12)$$

We concat the real part and imaginary part of one entity when calculating the cosine similarity. From Figure 5, we could find out that in every sub-graph, the values on the diagonal, which represent the cosine similarity between entity embeddings of the same entity on different geometric subspace, are always the highest in a row and exceed 0.95. Therefore, although their geometric interpretations are diverse, real and imaginary vectors in different subspaces are almost unanimous when trained to optimal settings.

D Dataset Overview

Dataset statistics are described in Table 5.

E Temporal Pattern Statistics

We calculate the occurrence of each temporal pattern introduced in Section 4 to give an overview distribution of the temporal patterns. Table 4 shows the statistics on ICEWS14, ICEWS05-15 and GDELT. If a group of quadruples, such as the examples shown in the 2nd column in Table 4, meets the definition in Section 4, we calculate it as one occurrence.

F Modeling Various Temporal Patterns

Proposition 5. *HGE can model (anti-)symmetry patterns introduced in Definitions 3 and 5.*

Proof. Let p be a relation with temporal symmetry. One condition for modeling this pattern is $S(s, p, \bar{o}, \tau) = S(o, p, \bar{s}, \tau)$. For simplicity of representation, we use $p_t = p_{s\tau}$. Without loss of generality, we assume that we have only a one-dimensional split-complex vector. Therefore, we have the following equality to fulfill temporal symmetry:

$$\begin{aligned} \left(s_a p_{ta} o_a + s_b p_{tb} o_a - s_a p_{tb} o_b - s_b p_{ta} o_b \right) = \\ \left(o_a p_{ta} s_a + o_b p_{tb} s_a - o_a p_{tb} s_b - o_b p_{ta} s_b \right). \end{aligned}$$

This leads to the following equality $s_b p_{tb} o_a = s_a p_{tb} o_b$. To hold this equality, we need to have either $p_{tb} = 0$ or $s_b o_a = s_a o_b$. So far, we show for a given grounded quadruple (s, p, o, τ), if our model learns (s, p, o, τ) to be true, it can also hold its temporal symmetry (o, p, s, τ) as true. To generalize this to the universal quantifier (every grounded quadruple), we can add one extra dimension to model temporal symmetry for the extra pair of entities. In the extended dimension for the new pair (s,o), we should have $p_{tb} = 0$ or $s_b o_a = s_a o_b$ to hold temporal symmetry. In this way, all pairs (s, o) which are connected by temporal symmetry relation will be held as true by the model. A similar procedure can be done for Dual and ComplEx spaces. Therefore, there exist assignments for embeddings of entities and relations that fulfill the encoding of the temporal symmetric pattern. $\square$

Table 4: Real examples and statistics of each pattern in the train set of ICEWS14, ICEWS05-15, GDELT.

Patterns	Examples	ICEWS14	ICEWS05-15	GDELT
static symmetric	(Iraq, sign formal agreement, Iran, 2014-04-06) (Iran, sign formal agreement, Iraq, 2014-04-06)	6,506	36,537	366,830
static inverse	(Fiji, host a visit, Julie Bishop, 2014-11-04), (Julie Bishop, make a visit, Fiji, 2014-11-04)	10,361	63,092	552,280
dynamic symmetric	(France, engage in negotiation, Poland, 2014-04-04) (Poland, engage in negotiation, France, 2014-02-20)	78,473	3,817,343	17,265,293
dynamic inverse	(Angela Merkel, discuss by telephone, Ukraine, 2014-03-14), (Ukraine, consult, Angela Merkel, 2014-03-27)	768,586	48,641,730	104,909,248
dynamic evolve	(South Korea, demand, Japan, 2014-07-15), (South Korea, reject judicial cooperation, Japan, 2014-07-18)	971,055	63,733,447	112,653,245

Proposition 6. *HGE can model inverse patterns introduced in Definitions 4 and 6.*

Proof. Let temporal relation p_1 be the inverse of the temporal relation p_2 at all time points (4). One condition to model this pattern is to fulfill $S(s, p_1, \bar{o}, \tau) = S(o, p_2, \bar{s}, \tau)$. Without loss of generality, we assume that we have only a one-dimensional split-complex vector. Therefore, we have the following equality to fulfill temporal inverse relationships:

$$\left(s_a p_{1ta} o_a + s_b p_{1tb} o_a - s_a p_{1tb} o_b - s_b p_{1ta} o_b\right) =$$
$$\left(o_a p_{2ta} s_a + o_b p_{2tb} s_a - o_a p_{2tb} s_b - o_b p_{2ta} s_b\right).$$

If we set $p_{1ta} = p_{2ta}$, $p_{1tb} = -p_{2tb}$, the above equality holds. This means there exist assignments for embeddings of entities, relations, and times that fulfill the encoding of temporal inverse patterns. Our proof can be generalized to d dimensional product space by adding one dimension per each grounded atom. For the pattern in 6, the proof procedes likewise. The only difference is that the time embedding will be different at the two times τ_1, τ_2 to hold $p_{1ta} = p_{2ta}$, $p_{1tb} = -p_{2tb}$. $\square$

Proposition 7. *Let us assume that relation p_1 evolve to relation p_2 as formalized in 7. HGE can model this pattern.*

Proof. Given that p_1 evolves to p_2, and also given the two times τ_1 and τ_2 with $\tau_1 \prec\preceq \tau_2$, to model the pattern, we need to have $S(s, p_1, \bar{o}, \tau_1) = S(s, p_2, \bar{o}, \tau_2)$. Without loss of generality, we assume that we have only a one-dimensional split-complex vector. Then, we must fulfill the following equality:

$$\left(s_a p_{1t_1 a} o_a + s_b p_{1t_1 b} o_a - s_a p_{1t_1 b} o_b - s_b p_{1t_1 a} o_b\right) =$$
$$\left(s_a p_{2t_2 a} o_a + s_b p_{2t_2 b} o_a - s_a p_{2t_2 b} o_b - s_b p_{2t_2 a} o_b\right).$$

For this equality to hold, it must be the case that $p_{1t_1 a} = p_{2t_2 a}$, $p_{1t_1 b} = p_{2t_2 b}$. Note that these equality conditions do not necessarily mean that the embedding of static and temporal relations in Equation 5 should be the same because different convex combinations can create the same vector for temporal relations. Considering the universal quantifier, we can add one extra dimension for each grounded atom to fulfill equality. A similar consideration can be applied to Dual and ComplEx spaces. Therefore, there exist assignments for embeddings of entities and relations that encode the patterns. $\square$

Table 5: Statistics for ICEWS14, ICEWS05-15, GDELT and Wikidata12k.

Dataset	ICEWS14	ICEWS05-15	GDELT	Wikidata12k
Entities	7,128	10,488	500	12,554
Relations	230	251	20	24
Times	365	4017	366	1,726
Train	72,826	386,962	2,735,685	32,497
Validation	8,941	46,275	341,961	4,062
Test	8,963	46,092	341,961	4,062

Proposition 8. *Let p be a temporary relation in time as defined in 8. HGE can model this relation.*
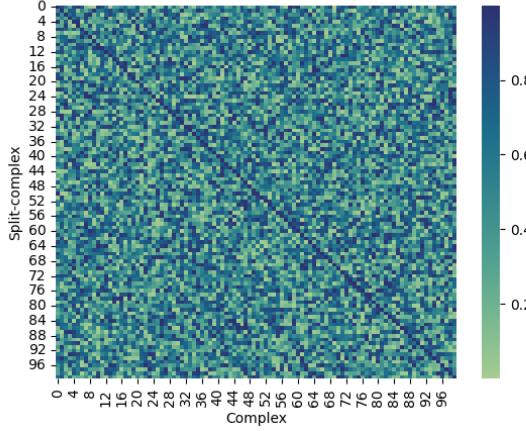
Proof. Let p be a temporary relation as in 8. To follow this pattern in the embedding space, for a given grounded atom (s, p, o, τ_1), there exist τ_0, τ_2 and also the embedding vectors for $s, p, o, \tau_0, \tau_1, \tau_2$ such that we have $S(s, p, \bar{o}, \tau_1) \neq S(s, p, \bar{o}, \tau_2)$ and $S(s, p, \bar{o}, \tau_1) \neq S(s, p, \bar{o}, \tau_0)$ as one possible condition to fulfill the pattern. Similar to the previous proofs, let us assume that we have only a one-dimensional split-complex vector. To fulfill the first condition (the second one will be similar), we have

$$\left(s_a p_{t_1 a} o_a + s_b p_{t_1 b} o_a - s_a p_{t_1 b} o_b - s_b p_{t_1 a} o_b\right) \neq$$
$$\left(s_a p_{t_2 a} o_a + s_b p_{t_2 b} o_a - s_a p_{t_2 b} o_b - s_b p_{t_2 a} o_b\right).$$
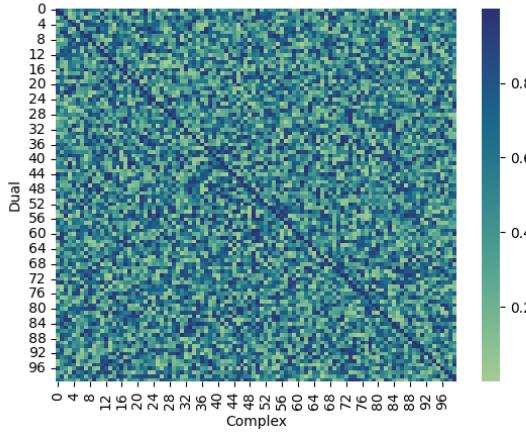
This can be simply fulfilled if we set $p_{t_1 a} \neq p_{t_2 a}, p_{t_1 b} \neq p_{t_2 b}$. In addition, we can have a large value for $S(s, p, \bar{o}, \tau_1)$ and a small value for $S(s, p, \bar{o}, \tau_2)$ (or vice versa) by properly setting the temporal relation close to zero at time τ_1 and high value at time τ_2 (and vice versa). A similar calculation can be done for Dual and ComplEx spaces. Therefore, there exist assignments for embeddings of entities and relations that encode the pattern. $\square$

G   Experiment Details
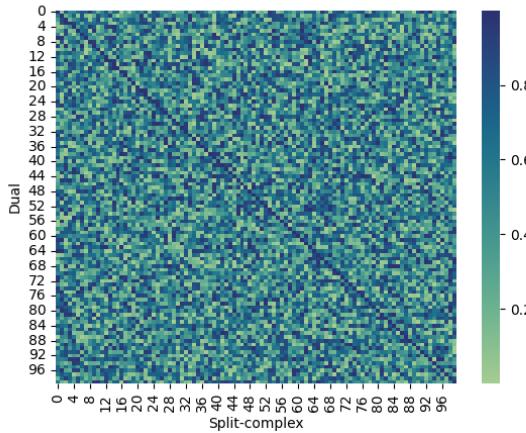
All experiments in the paper were run on the same NVIDIA A100 GPU device(40G GPU/100G CPU) with Ubuntu system 22.0. We implement a grid search to select the best regularizer weight from [5e-4, 3e-3, 5e-3, 3e-3, 1e-3, 3e-2, 1e-2, 1e-1]. A detailed list of hyperparamters is provided in hyperparamter.pdf file in the code folder of supplement material.

(a) Cosine similarity score between trained entity embeddings in Complex space and Split-complex space.

(b) Cosine similarity score between trained entity embeddings in Complex space and Split-complex space.

(c) Cosine similarity score between trained entity embeddings in Split-complex space and Dual space.

Figure 5: Cosine similarity scores between entity embeddings from different geometric space. x-axis and y-axis show the entity id on relevant geometric space

H Temporal Structural Patterns on Geometric subspaces

We consider symmetric patterns belonging to structural patterns too and define two other types of temporal structural patterns:

Definition 9. *Relation p forms a temporal star of size $n \in \mathbb{N}$ iff $\forall s$: $\exists o_1, \tau_1 \dots, o_n, \tau_n$: $Precedes(\tau_1, \tau_2)$ & $\dots$ & $Precedes(\tau(n-1), \tau_n)$ & (s, p, o_1, τ_1) & (s, p, o_2, τ_2) & $\dots$ & (s, p, o_n, τ_n).*

Definition 10. *A relation p forms a temporal hierarchy iff $\forall v_1, v_2, v_3, \tau_1, \tau_2 : (v_1, p, v_2, \tau_1)$ & $(v_2, p, v_3, \tau_2) \to \tau_1 \prec \tau_2$*

We investigate if heterogeneous geometric subspaces could represent different kinds of structural patterns. We extract 3 subsets for static symmetry, temporal hierarchy, and temporal star structural patterns from the test set of ICEWS14 and ICEWS05-15. Four variants of TNTComplEx+HGE model are tested in these subsets: 1) complex: only complex space is used. 2) split-complex: only split-complex space is used. 3) dual: only dual space is used. 4) HGE: the full model with three heterogeneous subspaces.

Table 6 shows that models using complex space perform best on static symmetric structural patterns. Models using split-complex space performs best on temporal hierarchy pattern while models using dual space perform best on temporal star pattern. This observation supports our core assumption that multiple geometric spaces may benefit temporal knowledge graph representation. Moreover, TNTComplEx+HGE performs better than all variants with single geometric spaces, demonstrating that the proposed product space with temporal geometric attention mechanism could integrate the advantages of individual subspaces.

I HGE's Time and Space Usage

As HGE reuses vectors for different geometric subspaces, the increased parameters to implement an HGE module will be $2|\mathcal{R}| * d$, which is the attention weights for two proposed attention mechanisms. We demonstrate the HGE's efficiency by comparing the number of parameters and running times of the original backbone with HGE-extended backbones. All models are trained with 200 epochs and we calculate the average running time of training epochs for each model. From Table 7, we observe that with the same embedding dimension d=1200 for entities and relations, the increased number of parameters and running time are rather moderate for HGE-extensions. Specifically, when TNTComplEx is extended by HGE, its performance is comparable to TLT-KGE with only half as many parameters and a shorter running time. Even if we decrease d of TNTComplEx+HGE to 1100, it still outperforms backbone TNTComplEx(d=1200) with fewer parameter numbers. This demonstrates that HGE's improvements do not come from the increased number of parameters, but rather from its representational approach.

J Baseline Selection

LCGE We found out the commonsense reasoning score introduced in equation 11 of LCGE(Niu and Li 2023) was

Table 6: MRR performance of heterogeneous geometric spaces on diverse structural pattern subsets.

Datasets	Structural Patterns	Statistics	TNTComplEx	complex	split-complex	dual	HGE
	static symmetric	1352	98.8	**99.5**	99.3	98.3	**99.5**
ICEWS14	temporal hierarchy	1193	69.5	70.4	**71.8**	71.0	**71.8**
	temporal star	6197	70.5	71.6	71.9	<u>72.9</u>	**73.0**
	static symmetric	7240	99.7	**99.8**	99.7	99.6	**99.8**
ICEWS05-15	temporal hierarchy	16703	72.7	72.8	<u>73.7</u>	72.5	**74.3**
	temporal star	39724	73.8	73.8	72.4	<u>74.7</u>	**75.4**

Table 7: Parameter number and average runtime for original backbones and backbones extended by HGE.

Datasets	Model	Rank(d)	Parameter number	Average epoch time(s)	MRR
	TNTComplEx	1200	20,191,200	1.80	60.7
ICEWS14	TLT-KGE	1200	38,693,400	2.25	63.0
	TNTComplEx+HGE	1100	19,520,600	2.10	62.9
	TNTComplEx+HGE	1200	21,295,200	2.19	63.0
	TNTComplEx	1200	37,221,600	11.79	66.6
ICEWS05-15	TLT-KGE	1200	81,360,600	13.91	68.6
	TNTComplEx+HGE	1100	35,224,200	11.52	67.7
	TNTComplEx+HGE	1200	38,426,400	12.13	68.1

Table 8: Results of LCGE in original paper and by our implementation

Model	ICEWS14				ICEWS05-15			
	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10
LCGE(Niu and Li 2023)	92.5	91.6	92.9	93.7	91.2	90.3	91.6	92.5
LCGE	61.6	53.2	66.7	77.5	61.8	51.4	68.1	81.2

considered during the training time but missed during the test time, which causes bias to final scores and rankings. We re-implemented the codes and attached our implementation in the supplementary material's code/LCGE_new folder. Table 8 shows the comparison of reported results in (Niu and Li 2023) and results by our implementation.

DyERNIE We do not include the baseline of Dy-ERNIE(Han et al. 2020) since this paper reports the results using the static filtered setting. Moreover, the code released by the authors is not complete to implement hyperbolic spaces, making it hard to report time-aware filtering results.

HSAE HSAE(Ren et al. 2023) adopts a hierarchy self-attention mechanism to incorporate information from different time shots. We do not include the baseline of HSAE because the author does not publish the codes.