TENSOR DECOMPOSITIONS FOR TEMPORAL KNOWLEDGE GRAPH COMPLETION WITH TIME PERSPECTIVE

Anonymous authors
Paper under double-blind review

ABSTRACT

Facts in the real world are often tied to time, such as the spread of diseases, and the state of military affairs. Therefore, knowledge graphs combined with temporal factors have gained growing attention. In the temporal knowledge graph, most researchers focus on the original facts and pay attention to their changes over time. The temporal factors are only used as auxiliary information for representation learning. In this paper, we try to observe from the perspective of time and find some interesting properties of temporal knowledge graph: (1) Simultaneousness. Various facts occur at the same time; (2) Aggregation. The facts may aggregately occur for a certain individual, organization, or location; (3) Associativity. Some specific relations tend to occur at specific times, such as celebrations at festivals. Based on the above three properties, we add a simple time-aware module to the existing tensor decomposition-based temporal knowledge graph model TComplEx Lacroix et al. (2020), which obtains impressive improvements and achieves state-of-the-art results on four standard temporal knowledge graph completion benchmarks. Specifically, in terms of mean reciprocal rank (MRR), we advance the state-of-the-art by +21.8% on ICEWS14, +16.9% on ICEWS05-15, +20.7% on YAGO15k, and 13.1% on GDELT.

1 INTRODUCTION

Knowledge graphs represent informative knowledge or facts of the real world as structured triples (head entity, relation, tail entity) also known as (subject, predicate, object). They have gained widespread attention for their successful usage in various applications (*e.g.*, question answering Saxena et al. (2020), bioinformatics Zitnik et al. (2018), and recommendation systems Mezni et al. (2021)). Although millions of entities and billions of facts exist in large-scale knowledge graphs, they still suffer from the incompleteness problem. Therefore, knowledge graph completion (also known as link prediction) methods are proposed to predict missing links among entities based on the known triples Bordes et al. (2013); Yang et al. (2015); Trouillon et al. (2016). Furthermore, in the real world, many facts are inherently tied to a specific time. For example, “Barack Obama is the president of USA” is only valid for the time period “2009 - 2017”. The task of temporal knowledge graph completion is to find missing links in graphs at precise points in time Leblay & Chekol (2018); Goel et al. (2020).

In temporal knowledge graph, these facts have some temporal metadata attached. The corresponding triples are transformed into quadruples (head entity, relation, tail entity, timestamp). Compared with timestamp, entities and relations have more interpretable interactions, such as symmetric/antisymmetric, inversion, *etc*. Trouillon et al. (2016); Sun et al. (2019). Therefore, when extending static knowledge graphs to temporal knowledge graphs, most knowledge graph embedding methods still mainly focus on triple facts Leblay & Chekol (2018); Garcia-Duran et al. (2018); Xu et al. (2020); Chen et al. (2022) and pay attention to their changes over time. While the temporal factors are only used as auxiliary information for representation learning, which leads to a lack of deeper mining for temporal factors.

In this work, we try to observe the temporal knowledge graph from a new perspective - **Time Perspective**, and explore some general properties from it. We then improve the current tensor decomposition model based on the observed properties. Our contributions can be summarized as follows:

Firstly, we observe the temporal knowledge graph from time perspective, and obtain three important properties, (1) Simultaneousness. Various facts occur at the same time; (2) Aggregation. The facts may aggregately occur for a certain individual, organization, or location; (3) Associativity. Some specific relations tend to occur at specific times, such as celebrations at festivals.

Then, we analyze the existing temporal tensor decomposition model TComplEx, and find that TComplEx degenerates when the inverse relations occur. The degenerated model cannot fully represent the other facts on the current timestamp, that is, TComplEx cannot satisfy the property simultaneousness. In addition, there is no suitable solution for the properties aggregation and associativity.

Finally, we propose three modules for three properties of time perspective for TComplEx, named as TPComplEx, and simplify to add the relative timestamp bias for the corresponding entities. Experimental results show that TPComplEx outperforms current state-of-the-art methods on four standard temporal knowledge graph completion benchmarks. Specifically, in terms of mean reciprocal rank (MRR), we advance the state-of-the-art by +21.8% on ICEWS14, +16.9% on ICEWS05-15, +20.7% on YAGO15k, and 13.1% on GDELT.

2 Related Work

2.1 Static Knowledge Graph Embedding Model

Roughly speaking, we can divide static knowledge graph embedding models into translational distance models and semantic matching models. The former use distance-based score functions, while the latter use similarity-based ones. For translational distance models, TransE Bordes et al. (2013) is the most widely used translation distance constraint model. It assumes that entities and relations satisfy $head+relation \approx tail$. However, TransE cannot handle 1-1-N, N-1-1, and N-1-N relations well Wang et al. (2014). TransH Wang et al. (2014) projects entities onto relation-specific hyperplanes to compensate for the shortcomings of TransE. TransR Lin et al. (2015) introduces relation-specific spatial transformations instead of hyperplanes. Moreover, RotatE Sun et al. (2019) defines each relation as a rotation from the source entity to the target entity in a complex vector space, which can represent various relation patterns. For semantic matching models, RESCAL Nickel et al. (2011) is a tensor factorization model which represents each relation as a full-rank matrix and obtains score function by matrix multiplication. DistMult Yang et al. (2015) simplifies RESCAL by restricting relation matrices to be diagonal. However, Distmult assumes that all relations are symmetric. ComplEx Trouillon et al. (2016) extends DistMult to complex space, and uses conjugate-transpose to model asymmetric relations. QuatE Zhang et al. (2019) extends the complex space into the quaternion space with two rotating surfaces. ConvE Dettmers et al. (2018) and InteractE Vashishth et al. (2020) employ convolutional neural networks to build score functions.

2.2 Temporal Knowledge Graph Embedding Model

Most static completion models cannot acquire temporal information when learning the embeddings of knowledge graphs, and perform poorly on temporal knowledge graphs Garcia-Duran et al. (2018); Leblay & Chekol (2018). TTransE Leblay & Chekol (2018) models the transition between time-aware relations of two adjacent facts by imposing temporal order constraints on the geometry of the embedding space. HyTE Dasgupta et al. (2018) associates temporal information with entities and relations by projecting them onto a temporal hyperplane. Both of them are constrained by the translational distance score function. TA-TransE and TA-DistMult Garcia-Duran et al. (2018) utilize recurrent neural networks to learn time-aware representations of relations and use standard scoring functions from TransE and DistMult. Motivated by diachronic word embeddings, DE-SimplE Goel et al. (2020) combines the diachronic entity embedding function with the static model SimplE and handles point-in-time event facts well. ATiSE Xu et al. (2019) regards the temporal evolution of entity and relation embeddings as combinations of trend component, seasonal component and random component. Inspired by the canonical decomposition of order-4 tensor, TComplEx Lacroix et al. (2020) introduces an extension of ComplEx for temporal Knowledge graph completion by adding additional complex temporal embeddings. ChronoR Sadeghian et al. (2021) propose chronological rotation embedding to capture rich interaction between the temporal and multi-relational characteristics. TeRo Xu et al. (2020) defines the temporal evolution of entity embedding as a rotation

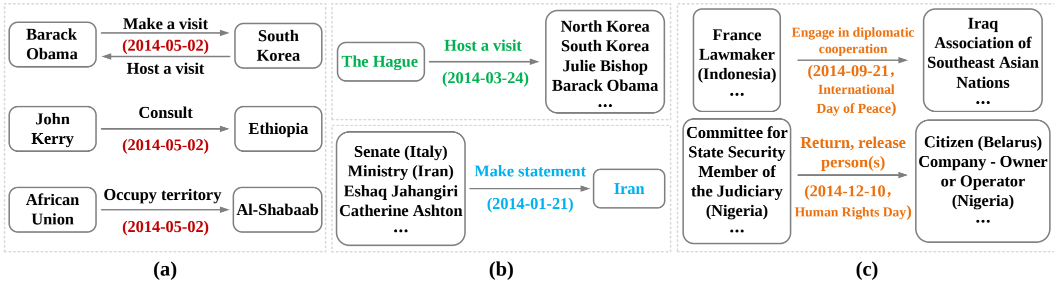

Figure 1: Examples of three properties of temporal knowledge graph extracted from the ICEWS14 knowledge graph. (a) Various facts occur on "2014-05-02"; (b) The facts that aggregately occur in the Hague and Iran; (c) More cooperative relations on the "2014-09-21, International Day of Peace", and more incidents of releasing citizens on the "2014-12-10, Human Rights Day".

in the complex vector space. RotateQVS Chen et al. (2022) further represents temporal entities as rotations in quaternion vector space and obtains better performance.

3 Observation from Time Perspective

In this section, we introduce the Observation for temporal knowledge graph from fact perspective and time perspective. Then, we introduce TCompEx, one of the current tensor decomposition-based temporal knowledge graph embedding model, and discuss its limitations.

Fact Perspective. For the additional temporal factor in temporal knowledge graph, most models only focus on the fact itself Leblay & Chekol (2018); Garcia-Duran et al. (2018); Xu et al. (2020); Chen et al. (2022). And pay attention to how facts change over time, *e.g.* (Barack Obama, Express intent to meet or negotiate, Abdel Fattah Al-Sisi, 2014-06-06), (Barack Obama, Discuss by telephone, Abdel Fattah Al-Sisi, 2014-06-11), (Adli Mansour, Accede to demands for change in leadership, Abdel Fattah Al-Sisi, 2014-06-14), (Abdel Fattah Al-Sisi, Make optimistic comment, Other Authorities / Officials (Egypt), 2014-06-24). These models usually pre-operate entity embeddings with additional temporal embeddings, such as weighted Goel et al. (2020), rotated Xu et al. (2020); Chen et al. (2022). Or pre-operate on relation embeddings, such as translation Leblay & Chekol (2018), LSTM Garcia-Duran et al. (2018). Finally, the transformed entities or relations are used to construct the scoring function following the static knowledge graph embedding models. The advantage of such operation is that it is simple and intuitive to introduce the temporal factors into the classic knowledge graph embedding models, but it lacks deeper mining of the temporal factors.

Time Perspective. For the observation from time perspective, we pay more attention to the facts that happened at a certain timestamp or time period. And it is easier to find some commonalities in time by jumping out of the specific entities of concern. Here we summarize the findings obtained from time perspective into three properties: (1) Simultaneousness. For example, at the timestamp "2014-05-02", there are facts (Barack Obama, Make a visit, South Korea), (John Kerry, Consult, Ethiopia), (African Union, Occupy territory, Al-Shabaab), etc. There may not be any correlation between these facts, that is, the embedding of time should have a greater generalization and can exist in all kinds of facts at the same time. (2) Aggregation. For example, "The Hague" "host a visit" at "2014-03-24" for "North Korea, South Korea, Julie Bishop, Barack Obama, *etc*." A lot of facts aggregately occur for a certain entity. (3) Associativity. For example, in the timestamp "2014-09-21" (International Day of Peace), more cooperation relations appear, such as "Engage in diplomatic cooperation, Express intent to engage in diplomatic cooperation (such as policy support)". See Figure 1 for more examples.

Symbol Description. Suppose that we have a temporal knowledge graph $\mathcal{G}$. We use $\mathcal{E}$ to denote the set of entities, $\mathcal{R}$ to denote the set of relations, and $\mathcal{T}$ to denote the set of timestamps. Then, the temporal knowledge graph $\mathcal{G}$ can be defined as a collection of quadruples, noted as (s, r, o, t), where a relation $r \in \mathcal{R}$ holds between a head entity $s \in \mathcal{E}$ and an tail entity $o \in \mathcal{E}$ at time t. The actual time t is represented by a timestamp $\tau \in \mathcal{T}$.

Limitations of TComplEx. TComplEx Lacroix et al. (2020) is an extension of ComplEx Trouillon et al. (2016) for temporal knowledge graph. The specific method is to add additional complex temporal embeddings to the Hermitian product. The scoring function is defined as

$$\begin{aligned}\phi(s,r,o,t) &= \mathrm{Re}\left(\left\langle C_s, C_r, \overline{C}_o, C_t\right\rangle\right) \\ &= \mathrm{Re}\left(\langle (a_s + b_s\mathbf{i}), (a_r + b_r\mathbf{i}), (a_o - b_o\mathbf{i}), (a_t + b_t\mathbf{i})\rangle\right),\end{aligned} \tag{1}$$

where C_* represents the complex embedding. $\mathbf{i}$ represents the imaginary unit. $a_*, b_* \in \mathbb{R}^k$ are the real and imaginary parts, respectively, and k is the embedding rank. Inverse relations (refer to Definition 3) widely exist in temporal knowledge graphs, such as (Barack Obama, Make a visit, South Korea, 2014-05-02) and (South Korea, Host a visit, Barack Obama, 2014-05-02). For such combination $(C_s, C_{r1}, C_o, C_t), (C_o, C_{r2}, C_s, C_t)$, we can get (refer to Appendix A.1 for more details)

$$\begin{cases} a_t = 0 \\ a_{r2} = -a_{r1} \\ b_{r2} = b_{r1} \end{cases} \quad \text{or} \quad \begin{cases} b_t = 0 \\ a_{r2} = a_{r1} \\ b_{r2} = -b_{r1}. \end{cases} \tag{2}$$

It means that the temporal complex embedding of TComplEx will degenerate to the real or imaginary part ($C_t = a_t$ or $C_t = b_t\mathbf{i}$) if there is an inversion relation on the timestamp. And the degenerated model cannot fully represent the other facts on the current timestamp, that is, TComplEx cannot satisfy the property 1 Simultaneousness - various facts occur at the same time. In Appendix A.1, we visualize the phenomenon of Equation (2) when the inversion relation exists, and the performance on link prediction decreases when the TComlEx temporal embedding has only real or imaginary parts. In addition, TComplEx also lacks suitable solutions for properties 2 Aggregation and properties 3 Associativity.

4 Proposed Model

To alleviate the problem of TComplEx degenerating when the inversion relation occurs, the intuitive solution is to add extra temporal embeddings like $\phi(s,r,o,t) = \mathrm{Re}\left(\left\langle C_s, C_r, \overline{C}_o, C_{t1}\right\rangle\right) + G(C_{t*})$, where C_{t*} represents the extra temporal complex embedding. To address property 2 Aggregation (examples can be seen in Figure 1 b), we define $G_1(C_{t2}) = \mathrm{Re}\left(\left\langle C_r, \overline{C}_o, C_{t1}, C_{t2}\right\rangle\right)$ for tail entity aggregation, and $G_2(C_{t3}) = \mathrm{Re}\left(\left\langle C_s, C_r, C_{t1}, \overline{C}_{t3}\right\rangle\right)$ for head entity aggregation. To address property 3 Associativity (examples can be seen in Figure 1 c), we define $G_3(C_{t4}) = \mathrm{Re}\left(\langle C_r, C_{t1}, C_{t4}\rangle\right)$ for the associated relation and timestamp. Then, the new scoring function is defined as

$$\begin{aligned}\phi(s,r,o,t) &= \mathrm{Re}\left(\left\langle C_s, C_r, \overline{C}_o, C_{t1}\right\rangle\right) + G_1(C_{t2}) + G_2(C_{t3}) + G_3(C_{t4}) \\ &= \mathrm{Re}\left(\left\langle C_s, C_r, \overline{C}_o, C_{t1}\right\rangle\right) + \mathrm{Re}\left(\left\langle C_r, \overline{C}_o, C_{t1}, C_{t2}\right\rangle\right) \\ &+ \mathrm{Re}\left(\left\langle C_s, C_r, C_{t1}, \overline{C}_{t3}\right\rangle\right) + \mathrm{Re}\left(\langle C_r, C_{t1}, C_{t4}\rangle\right).\end{aligned} \tag{3}$$

To reduce the temporal embedding vector, we denote the fourth temporal embedding vector as $C_{t4} = \left\langle C_{t2}, \overline{C}_{t3}\right\rangle$. Then, we can obtain the final scoring function

$$\begin{aligned}\phi(s,r,o,t) &= \mathrm{Re}\left(\left\langle C_s, C_r, \overline{C}_o, C_{t1}\right\rangle\right) + \mathrm{Re}\left(\left\langle C_r, \overline{C}_o, C_{t1}, C_{t2}\right\rangle\right) \\ &+ \mathrm{Re}\left(\left\langle C_s, C_r, C_{t1}, \overline{C}_{t3}\right\rangle\right) + \mathrm{Re}\left(\left\langle C_r, C_{t1}, C_{t2}, \overline{C}_{t3}\right\rangle\right) \\ &= \mathrm{Re}\left(\left\langle C_s + C_{t2}, C_r, \overline{C_o + C_{t3}}, C_{t1}\right\rangle\right).\end{aligned} \tag{4}$$

Compared with TComplEx, TPComplEx adds two additional temporal embedding biases in head entity and tail entity, respectively. We keep the total embedding parameters of the two models consistent by controlling the embedding rank.

Loss Function. Following Lacroix et al. (2018), for each of the train quadruples (s, r, o, t), the instantaneous multiclass loss is

$$\mathcal{L} = -\phi(s, r, o, t) + \log\left(\sum_{o' \neq o \cup o' \in \mathcal{E}} \exp\left(\phi(s, r, o', t)\right)\right) + \Omega(s, r, o, t), \tag{5}$$

where $\Omega(s, r, o, t)$ is regularization. We take the entities with temporal bias as the input of regularization, and adopt N3 regularization Lacroix et al. (2018), which is defined as

$$\Omega(s, r, o, t) = \lambda_1 \left(\|C_s + C_{t2}\|_3^3 + \|C_r\|_3^3 + \|C_o + C_{t3}\|_3^3\right) + \lambda_2 \|C_{t1}\|_3^3, \tag{6}$$

where λ_1 and λ_2 are the regularization weights of entity-relation embedding and temporal embedding, respectively.

4.1 Modeling the Properties of Time Perspective

For property 1 Simultaneousness, when there is an inverse relation pattern, $C_{t1} = a_{t1} + b_{t1}\mathbf{i}$ will degenerate into $C_{t1} = a_{t1}$, or $C_{t1} = b_{t1}\mathbf{i}$ (refer to Appendix A.4 for details). While C_{t2}, C_{t3} can still retain the value of the real and imaginary parts at the same time. We test the degraded model in Appendix A.4, and the experiments show that TPComplEx gets a smaller decrease than TComplEx when the temporal complex embedding C_{t1} has only real or imaginary parts.

For property 2 Aggregation, we take head entity aggregation as an example. An aggregated head entity will exist that $\forall i \in \{1, ..., m\}$, (s, r, o_i, t) can hold in temporal knowledge graphs simultaneously. For the head entity aggregation part of TPComplEx $G_2(s, r, o, t)$, we always have

$$G_2(s, r, o_1, t) = \operatorname{Re}\left(\left\langle C_s, C_r, C_{t1}, \overline{C}_{t3}\right\rangle\right) = ... = G_2(s, r, o_m, t) \tag{7}$$

Similarly, the same conclusion can be obtained in the tail entity aggregation part of TPComplEx.

For property 3 Associativity, we define the relation r and timestamp t are associated if $\forall i \in \{0, ..., m\}$, (s_i, r, o_i, t) can hold in temporal knowledge graphs simultaneously. For this part of TPComplEx $G_3(s, r, o, t)$, we always have

$$G_3(s_1, r, o_1, t) = \operatorname{Re}\left(\left\langle C_r, C_{t1}, C_{t2}, \overline{C}_{t3}\right\rangle\right) = ... = G_2(s_m, r, o_m, t) \tag{8}$$

The relation here mainly refers to having a specific meaning and existing on a specific timestamp, such as “Engage in diplomatic cooperation” with “2014-09-21 (International Day of Peace)”, *etc*. But there are also relations like “Make statement”. These generalized relations exist on almost all timestamps, and their associations are not strong.

4.2 Modeling Various Relation Patterns.

Definition 1 *A relation r is symmetric, if $\forall s, o, t$, $r(s, o, t) \wedge r(o, s, t)$ holds True*

Definition 2 *A relation r is asymmetric, if $\forall s, o, t$, $r(s, o, t) \wedge \neg r(o, s, t)$ holds True*

Definition 3 *Relation r_1 is the inverse of r_2, if $\forall s, o, t$, $r_1(s, o, t) \wedge r_2(o, s, t)$ holds True*

Similar to TComplEx, TPComplEx can also represent various relational patterns. In addition, in a specific relation pattern, the additional temporal bias has some properties, such as $C_{t2} = C_{t3}$ in both the symmetric relation and inverse relation patterns. The specific lemmas and proofs are as follows:

Lemma 1 *TPComplEx can infer the symmetry pattern. (See proof in Appendix A.2)*

Lemma 2 *TPComplEx can infer the antisymmetry pattern. (See proof in Appendix A.3)*

Lemma 3 *TPComplEx can infer the inversion pattern. (See proof in Appendix A.4)*

5 EXPERIMENTS

5.1 EXPERIMENTAL SETTING

Benchmark Datasets: To evaluate our proposed TPComplEx, we perform link prediction task on four commonly used temporal knowledge graph benchmark datasets, namely ICEWS14, ICEWS05-15, YAGO15k Garcia-Duran et al. (2018) and GDELT Trivedi et al. (2017). Table 1 summarises the details of the four datasets.

- ICEWS datasets are samplings from the Integrated Conflict Early Warning System (ICEWS) Lautenschlager et al. (2015), which is a repository containing political events with a specific timestamp. ICEWS14 and ICEWS05-15 Garcia-Duran et al. (2018) are two subsets of ICWES corresponding to facts in 2014 and facts between 2005 and 2015.
- Yago15K Garcia-Duran et al. (2018) is a modification of FB15k Bordes et al. (2013) and YAGO Hoffart et al. (2013) which adds "occursSince" and "occursUntil" timestamps to each quadruples. In the evaluation setting of Garcia-Duran et al. (2018), the incomplete triples to complete are of the form (subject, predicate, ?, occursSince | occursUntil, timestamp) (with tensors of order 5). Following the setting of Lacroix et al. (2020), we choose to unfold the (occursSince, occursUntil) and the predicate mode (using reciprocal setting) together, multiplying its size by two.
- GDELT Trivedi et al. (2017) is a subset of Global Database of Events, Language, and Tone Leetaru & Schrodt (2013), consisting of the facts from April 1, 2015 to March 31, 2016. We take the same pretreatment of the train, validation and test sets as Goel et al. (2020), to make the problem into a Temporal Knowledge Graph Completion rather than an extrapolation problem.

Dataset	#Entities	#Relations	#Timestamps	#train	#validation	#test
ICEWS14	7,128	230	365	72,826	8,941	8,963
ICEWS05-15	10,488	251	4,017	386,962	46,275	46,092
YAGO15k	15,403	34	198	110,441	13,815	13,800
GDELT	500	20	366	2,735,685	341,961	341,961

Table 1: Statistics of four experimented datasets.

Evaluation Protocol: For each quadruple (s, r, o, t) in the test dataset, we replace either the head entity s or the tail entity o with the total list of the embedding entities. Then, we base the score function to rank the candidate entities in descending order. The filtered setting is used to remove some correct results that appear in the training set or validation set but not in test set. We choose Mean Reciprocal Rank (MRR) and Hits at N (H@N) as the evaluation metrics. MRR is the average inverse rank for correct entities. Hit@n measures the proportion of correct entities in the top n entities. Higher MRR or H@N indicates better performance.

Baselines: We compare with both SOTA static and temporal knowledge graph embedding models. For static models, we reporte TransE Bordes et al. (2013), DistMult Yang et al. (2015), ComplEx Trouillon et al. (2016), SimplE Kazemi & Poole (2018a). For temporal models, we reporte TTransE Leblay & Chekol (2018), HyTE Dasgupta et al. (2018), TA-DistMult Garcia-Duran et al. (2018), TComplEx Lacroix et al. (2020), DE-SimplE Goel et al. (2020), TeRo Xu et al. (2020), ChronoR Sadeghian et al. (2021), RotateQVS Chen et al. (2022). Note that TComplEx Lacroix et al. (2020) is our main baseline. Our TPComplEx is improved based on this model, and both have the same number of embedding parameters (refer to Table 8).

Implementation Details: We implement our method based on the PyTorch library Paszke et al. (2019), and run on a single NVIDIA RTX 2080 Ti. We tune our model using grid search to select the optimal hyperparameters based on the performance of the validation dataset. The ranges of the hyperparameters of regularization rates λ_1 and λ_2 are adjusted in $\{1.0, 0.01, 0.001, 0.0001, 0.00001\}$. In order to obtain comparable results, we use Table 8 and dataset statistics Table 1 to compute the rank for each (model, dataset) pair that matches the number of parameters used in TComplEx Lacroix et al. (2020). Our model is optimized with Adagrad Duchi et al. (2011), with a learning rate of 0.1, and a batch-size of 1000 for all datasets. For specific embedding ranks, please refer to Table 8, and *our code will be released*.

	ICEWS14				ICEWS05-15			
	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
TransE Bordes et al. (2013)	28.0	9.4	-	63.7	29.4	9.0	-	66.3
DistMult Yang et al. (2015)	43.9	32.3	-	67.2	45.6	33.7	-	69.1
ComplEx Trouillon et al. (2016)	47	35	54	71	49	37	55	73
SimplE Kazemi & Poole (2018b)	45.8	34.1	51.6	68.7	47.8	35.9	53.9	70.8
TTransE Leblay & Chekol (2018)	25.5	7.4	-	60.1	27.1	8.4	-	61.6
HyTE Dasgupta et al. (2018)	29.7	10.8	41.6	65.5	31.6	11.6	44.5	68.1
TA-DistMult Garcia-Duran et al. (2018)	47.7	36.3	-	68.6	47.4	34.6	-	72.8
TComplEx Lacroix et al. (2020) (B/L)	61	53	66	77	66	59	71	80
DE-SimplE Goel et al. (2020)	52.6	41.8	59.2	72.5	51.3	39.2	57.8	74.8
TeRo Xu et al. (2020)	56.2	46.8	62.1	73.2	58.6	46.9	66.8	79.5
ChronoR Sadeghian et al. (2021)	<u>62.5</u>	<u>54.7</u>	<u>66.9</u>	<u>77.3</u>	<u>67.5</u>	<u>59.6</u>	<u>72.3</u>	<u>82.0</u>
RotateQVS Chen et al. (2022)	59.1	50.7	64.2	75.4	63.3	52.9	70.9	81.3
TPComplEx	**84.3**	**79.5**	**87.5**	**92.7**	**84.4**	**79.2**	**88.0**	**93.2**

Table 2: Evaluation results on ICEWS14 and ICEWS05-15 datasets. The best score is in **bold** and second best score is <u>underlined</u>.

	YAGO15k				GDELT			
	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
TransE Bordes et al. (2013)	29.6	22.8	-	46.8	11.3	0.0	15.8	31.2
DistMult Yang et al. (2015)	27.5	21.5	-	43.8	19.6	11.7	20.8	34.8
ComplEx Trouillon et al. (2016)	36	29	36	<u>54</u>	23.1	14.4	24.8	40.3
SimplE Kazemi & Poole (2018b)	-	-	-	-	20.6	12.4	22.0	36.6
TTransE Leblay & Chekol (2018)	32.1	23.0	-	51.0	11.5	0.0	16.0	31.8
HyTE Dasgupta et al. (2018)					11.8	0.0	16.5	32.6
TA-DistMult Garcia-Duran et al. (2018)	29.1	21.6	-	47.6	20.6	12.4	21.9	36.5
TComplEx Lacroix et al. (2020) (B/L)	36	28	<u>38</u>	<u>54</u>	26.8	<u>17.8</u>	28.7	43.1
DE-SimplE Goel et al. (2020)	-	-	-	-	23.0	14.1	24.8	40.3
TeRo Xu et al. (2020)	-	-	-	-	24.5	15.4	26.4	42.0
ChronoR Sadeghian et al. (2021)	<u>36.6</u>	<u>29.2</u>	37.9	53.8	-	-	-	-
RotateQVS Chen et al. (2022)	-	-	-	-	<u>27.0</u>	17.5	<u>29.3</u>	<u>45.8</u>
TPComplEx	**57.3**	**51.5**	**60.0**	**68.4**	**40.1**	**32.1**	**42.6**	**55.6**

Table 3: Evaluation results on YAGO15k and GDELT datasets. The best score is in **bold** and second best score is <u>underlined</u>.

5.2 Main Results

The experimental results on ICEWS14 and ICEWS05-15 datasets are reported in Table 2, results on YAGO15k and GDELT datasets are reported in Table 3. Overall, temporal knowledge graph embedding models are better than static knowledge graph embedding models, which demonstrates the effectiveness of incorporating temporal information. For the proposed TPComplEx, we observe that our model substantially outperforms all the baseline models over the four datasets across all metrics consistently. Especially compared to TComplEx, TPComplEx has the same number of embedding parameters, but achieves a large improvement with an average MRR +19.1% on the four datasets.

For the ICEWS14 and ICEWS05-15 datasets, both of which are datasets recording political events. Their simultaneousness and associativity are obvious from time perspective, so they have been greatly improved in TPComplEx. YAGO15K dataset is constructed based on FB15k, which contains large 1-N and N-1 relations Wang et al. (2014), so it satisfies the aggregation from time perspective and TPComplEx is suitable for this dataset. GDELT dataset is a comprehensive record of time facts, but due to its small number of entities and relations (500+20), and relatively large number of timestamps (366), our model (using $3 \times C_t$ for timestamp) has smaller embedding rank under the same number of embedding parameters. Therefore, TPComplEx has limited improvement compared to the other three datasets.

For the training time, we take the GDELT dataset as an example. Under the same number of embedding parameters, we obtain a speed of 57.5k triples per second, leading to experiments time of 1.58 hours. Correspondingly, TComplEx is 64.4k triples per second and 1.42 hours. When ten times the embedding rank, TPComplEx is 6.3k triples per second and 14.42 hours, and TComplEx is 6.9k triples per second and 13.22 hours. Overall, the increase in training time is acceptable compared to the improvement in results.

Quadruples in test set	TP	TComplEx			TPComplEx		
		MRR	H@1	H@10	MRR	H@1	H@10
(*, *, *, 2014-05-02)	P1	73.6	69.6	82.1	**85.8**	**82.1**	**94.6**
(*, *, *, 2014-11-19)	P1	64.7	57.9	77.6	**85.9**	**82.6**	**92.1**
(Citizen (Nigeria), Make an appeal or request, *, 2014-12-29)	P2	80.5	70.0	90.0	**88.3**	**80.0**	**100.0**
(*, Make statement, Iraq, 2014-08-11)	P2	73.8	58.3	91.7	**95.8**	**91.7**	**100.0**
(*, Exp. intent to engage in diplomatic cooperation, *, *-08-11)	P3	26.6	16.7	58.3	**75.7**	**66.7**	**83.3**
(*, Arrest, detain, or charge with legal action, *, *-12-10)	P3	87.5	75.0	**100.0**	**95.8**	**92.9**	**100.0**

Table 4: Evaluation results on different quadruples in test set. Where TP represents time perspective, P1 represents property 1 Simultaneousness, P2 represents property 2 Aggregation, P3 represents property 3 Associativity, * in *-08-11 and *-12-10 represents the years 2005 to 2015, other * represents the rest elements of quadruples in the test set that satisfy the current constraints. The first four group quadruples are from the ICEWS14 dataset, and the others are from the ICEWS05-15 dataset.

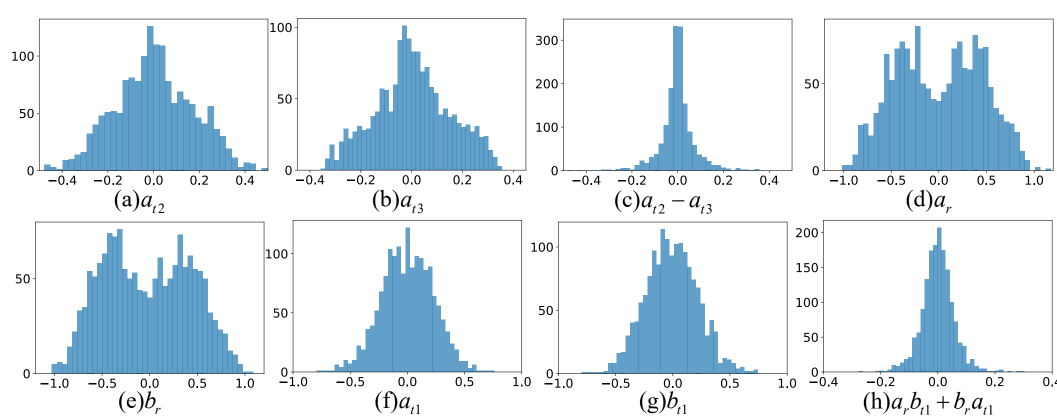

Figure 2: Histogram visualization of TPComplEx symmetry relation and timestamp. The relation r is "Consult", and the timestamp t is "2014-10-23". From Appendix A.2 we can get $a_{t2} = a_{t3}$ and $a_r b_{t1} + b_r a_{t1} = 0$.

5.3 ANALYSIS AND CASE STUDY

Quadruple Test of Different Properties from Time Perspective. According to the definition of the three properties of time perspective in Section 3, we select different groups of quadruples for testing. Table 4 shows the main results. Among them, the timestamp in TP1 (property 1 Simultaneousness) is selected with inversion relations exiting (such as "Make a visit" and "Host a visit"). In TP2 and TP3 (property 2 Aggregation and property 3 Associativity), due to the strong constraints in quadruples, we also take the validation set for testing. It can be seen from Table 4 that the test results of TPComplEx in these groups of quadruples are all better than TComplEx.

Visualize Some Typical Relation Patterns. To further verify the learned relation patterns, we visualize some examples. For symmetry pattern, TPComplEx requires relation and timestamp embeddings to satisfy $a_{t2} = a_{t3}, b_{t2} = b_{t3}$ and $a_r b_{t1} + b_r a_{t1} = 0$ (refer to Appendix A.2 for details). We take the facts (France, Consult, Canada, 2014-10-23) and (Canada, Consult, France, 2014-10-23) as examples, as shown in Figure 2. In addition, Figure 4 shows a visualization of the inversion relations, see Appendix A.4 for more details.

	ICEWS05-15				GDELT			
	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
TPComplEx (×0.1)	74.5	68.0	78.5	86.6	28.3	19.9	30.4	44.6
TPComplEx (×0.2)	76.8	70.7	80.6	87.9	31.2	22.7	33.4	47.5
TPComplEx (×0.5)	82.6	77.3	86.3	92.3	35.5	27.2	37.9	51.6
TPComplEx	84.4	79.2	88.0	93.2	40.1	32.1	42.6	55.6
TPComplEx (×2)	84.5	79.3	**88.2**	93.3	43.1	35.5	45.8	57.9
TPComplEx (×5)	84.5	79.3	88.1	93.3	50.7	**43.9**	53.7	63.2
TPComplEx (×10)	**84.6**	**79.5**	**88.2**	**93.4**	**51.1**	43.5	**54.6**	**65.4**

Table 5: Evaluation results on ICEWS05-15 and GDELT with different embedding ranks. Where × means n times the embedding rank than the original, *e.g.* 0.1 × of embedding rank 886 in IEWSC05-15 is 88.

Effect of Embedding Rank. We mainly test the effect of different embedding ranks on ICEWS05-15 and GDELT datasets. From Table 5, when the embedding rank is reduced, the test results decrease accordingly. However, even if the embedding rank is reduced to one-tenth of the original model, the obtained results are still very competitive (refer to Table 2 and Table 3). Furthermore, increasing the embedding rank has a larger impact on GDELT dataset, which is consistent with our analysis in Section 5.2. Compared to the original model, the ten times embedding rank improves result by +11.0% on MRR.

6 CONCLUSION

In this work, we analyze the temporal knowledge graph from a new perspective - **Time Perspective**, and obtain three important properties, namely Simultaneousness, Aggregation, and Associativity. Then, we analyze the existing temporal tensor decomposition model TComplEx, and find that TComplEx degenerates when the inverse relations occur. The degenerated model cannot fully represent the other facts on the current timestamp, that is, TComplEx cannot satisfy the property simultaneousness. In addition, there is lack of suitable solutions for the properties aggregation and associativity. Based on this, we propose three modules for three properties, and simplify them to add the relative temporal bias for the corresponding entities. Experimental results show that TPComplEx outperforms current state-of-the-art methods by a large margin on four standard datasets. Further experimental analysis verifies that TPComplEx can better handle the three properties of time perspective and model various relation patterns. For further work, one direction is to mine the properties of temporal knowledge graph from different perspectives, and apply them to more temporal knowledge graph completion models, such as ChronoR Sadeghian et al. (2021), RotateQVS Chen et al. (2022), *etc*. Another direction is to eliminate the gap between low-dimensional and high-dimensional models, and increase the expressivity of models at a lower number of embedding parameters per entity.

REFERENCES

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Proc. of Annual Conference on Neural Information Processing Systems*, 2013.

Kai Chen, Ye Wang, Yitong Li, and Aiping Li. Rotateqvs: Representing temporal information as rotations in quaternion vector space for temporal knowledge graph completion. In *Proc. of Annual Meeting of the Association for Computational Linguistics*, pp. 5843–5857, 2022.

Shib Sankar Dasgupta, Swayambhu Nath Ray, and Partha Talukdar. Hyte: Hyperplane-based temporally aware knowledge graph embedding. In *Proc. of Conference on Empirical Methods in Natural Language Processing*, pp. 2001–2011, 2018.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. Convolutional 2d knowledge graph embeddings. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 32, pp. 1811–1818, 2018.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7):2121–2159, 2011.

Alberto Garcia-Duran, Sebastijan Dumančić, and Mathias Niepert. Learning sequence encoders for temporal knowledge graph completion. In *Proc. of Conference on Empirical Methods in Natural Language Processing*, pp. 4816–4821, 2018.

Rishab Goel, Seyed Mehran Kazemi, Marcus Brubaker, and Pascal Poupart. Diachronic embedding for temporal knowledge graph completion. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 34, pp. 3988–3995, 2020.

Johannes Hoffart, Fabian M Suchanek, Klaus Berberich, and Gerhard Weikum. Yago2: A spatially and temporally enhanced knowledge base from wikipedia. *Artificial intelligence*, 194:28–61, 2013.

Seyed Mehran Kazemi and David Poole. Simple embedding for link prediction in knowledge graphs. *Proc. of Annual Conference on Neural Information Processing Systems*, 31, 2018a.

Seyed Mehran Kazemi and David Poole. Simple embedding for link prediction in knowledge graphs. In *Proc. of Annual Conference on Neural Information Processing Systems*, pp. 4289–4300, 2018b.

Timothée Lacroix, Nicolas Usunier, and Guillaume Obozinski. Canonical tensor decomposition for knowledge base completion. In *Proc. of International Conference on Machine Learning*, pp. 2863–2872, 2018.

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

Jennifer Lautenschlager, Steve Shellman, and Michael Ward. Icews event aggregations. *Harvard Dataverse*, 3(595):28, 2015.

Julien Leblay and Melisachew Wudage Chekol. Deriving validity time in knowledge graph. In *Companion Proc. of the The Web Conference*, pp. 1771–1776, 2018.

Kalev Leetaru and Philip A Schrodt. Gdelt: Global data on events, location, and tone, 1979–2012. In *ISA annual convention*, volume 2, pp. 1–49, 2013.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. Learning entity and relation embeddings for knowledge graph completion. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 29, pp. 2181–2187, 2015.

Haithem Mezni, Djamal Benslimane, and Ladjel Bellatreche. Context-aware service recommendation based on knowledge graph embedding. *IEEE Transactions on Knowledge and Data Engineering*, 2021.

Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. A three-way model for collective learning on multi-relational data. In *Proc. of International Conference on Machine Learning*, volume 11, pp. 809–816, 2011.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Proc. of Annual Conference on Neural Information Processing Systems*, 32, 2019.

Ali Sadeghian, Mohammadreza Armandpour, Anthony Colas, and Daisy Zhe Wang. Chronor: rotation based temporal knowledge graph embedding. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 35, pp. 6471–6479, 2021.

Apoorv Saxena, Aditay Tripathi, and Partha Talukdar. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *Proc. of Annual Meeting of the Association for Computational Linguistics*, pp. 4498–4507, 2020.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. Rotate: Knowledge graph embedding by relational rotation in complex space. In *Proc. of International Conference on Learning Representations*, 2019.

Rakshit Trivedi, Hanjun Dai, Yichen Wang, and Le Song. Know-evolve: Deep temporal reasoning for dynamic knowledge graphs. In *Proc. of International Conference on Machine Learning*, pp. 3462–3471, 2017.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. Complex embeddings for simple link prediction. In *Proc. of International Conference on Machine Learning*, volume 48, pp. 2071–2080, 2016.

Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, Nilesh Agrawal, and Partha Talukdar. Interacte: Improving convolution-based knowledge graph embeddings by increasing feature interactions. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 34, pp. 3009–3016, 2020.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. Knowledge graph embedding by translating on hyperplanes. In *Proc. of AAAI Conference on Artificial Intelligence*, volume 28, pp. 1112–1119, 2014.

Chengjin Xu, Mojtaba Nayyeri, Fouad Alkhoury, Hamed Shariat Yazdi, and Jens Lehmann. Temporal knowledge graph embedding model based on additive time series decomposition. *arXiv preprint arXiv:1911.07893*, 2019.

Chengjin Xu, Mojtaba Nayyeri, Fouad Alkhoury, Hamed Shariat Yazdi, and Jens Lehmann. Tero: A time-aware knowledge graph embedding via temporal rotation. In *Proc. of International Conference on Computational Linguistics*, pp. 1583–1593, 2020.

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

Shuai Zhang, Yi Tay, Lina Yao, and Qi Liu. Quaternion knowledge graph embeddings. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Proc. of Annual Conference on Neural Information Processing Systems*, volume 32, pp. 2735–2745, 2019.

Marinka Zitnik, Monica Agrawal, and Jure Leskovec. Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics*, 34:i457–i466, 2018.

A Appendix

A.1 TComplEx with Inverse Relation Pattern

For TComplEx, we can partially expand its scoring function

$$\begin{aligned}\phi(s,r,o,t) &= \mathrm{Re}\left(\left\langle C_s, C_r, \overline{C}_o, C_t\right\rangle\right)\\ &= \mathrm{Re}\left(\left\langle \langle C_s, \overline{C}_o\rangle, \langle C_r, C_t\rangle\right\rangle\right)\\ &= \mathrm{Re}\left(\langle (a_s a_o + b_s b_o + (-a_s b_o + b_s a_o)\mathbf{i}), (a_r a_t - b_r b_t + (a_r b_t + b_r a_t)\mathbf{i})\rangle\right)\\ &= (a_s a_o + b_s b_o)(a_r a_t - b_r b_t) - (-a_s b_o + b_s a_o)(a_r b_t + b_r a_t).\end{aligned} \tag{9}$$

For the inversion relations (s, r_1, o, t) and (o, r_2, s, t), we need to prove that $\forall s, o, t$, the following equations hold

$$\mathrm{Re}\left(\left\langle C_s, C_{r1}, \overline{C}_o, C_t\right\rangle\right) = \mathrm{Re}\left(\left\langle C_o, C_{r2}, \overline{C}_s, C_t\right\rangle\right). \tag{10}$$

Firstly, we expand the left term:

$$\begin{aligned}&\mathrm{Re}\left(\left\langle C_s, C_{r1}, \overline{C}_o, C_t\right\rangle\right)\\ &= (a_s a_o + b_s b_o)(a_{r1} a_t - b_{r1} b_t) - (-a_s b_o + b_s a_o)(a_{r1} b_t + b_{r1} a_t).\end{aligned} \tag{11}$$

We then expand the right term:

$$\begin{aligned}&\mathrm{Re}\left(\left\langle C_o, C_{r2}, \overline{C}_s, C_t\right\rangle\right)\\ &= (a_o a_s + b_o b_s)(a_{r2} a_t - b_{r2} b_t) - (-a_o b_s + b_o a_s)(a_{r2} b_t + b_{r2} a_t)\\ &= (a_s a_o + b_s b_o)(a_{r2} a_t - b_{r2} b_t) - (-a_o b_s + b_o a_s)(-a_{r2} b_t - b_{r2} a_t).\end{aligned} \tag{12}$$

Comparing Equation (11) and Equation (12), we can get

$$\begin{cases} a_t(a_{r1} - a_{r2}) - b_t(b_{r1} - b_{r2}) = 0\\ b_t(a_{r1} + a_{r2}) + a_t(b_{r1} + b_{r2}) = 0. \end{cases} \tag{13}$$

The satisfaction of Equation (13) requires a constraint on timestamp t, that is,

$$\begin{cases} a_t = 0\\ a_{r2} = -a_{r1}\\ b_{r2} = b_{r1} \end{cases} \quad \text{or} \quad \begin{cases} b_t = 0\\ a_{r2} = a_{r1}\\ b_{r2} = -b_{r1}. \end{cases} \tag{14}$$

We visualize the TComplEx embedding vector of inverse relation and the timestamp ((Romania, Host a visit, Evangelos Venizelos, 2014-02-20) and (Evangelos Venizelos, Make a visit, Romania, 2014-02-20)) in Figure 3, which is as expected from Equation (14). Therefore, when a reverse relation occurs in some timestamp, the embedding vector of timestamp will degenerate into a pure real or pure imaginary part, which affects TComplEx modeling other facts under that timestamp. That is, TComplEx cannot satisfy the requirement of temporal perspective property 1 Simultaneousness. Table 6 shows that the experimental performance decreases when the temporal complex embedding of TComplEx has only real or imaginary parts.

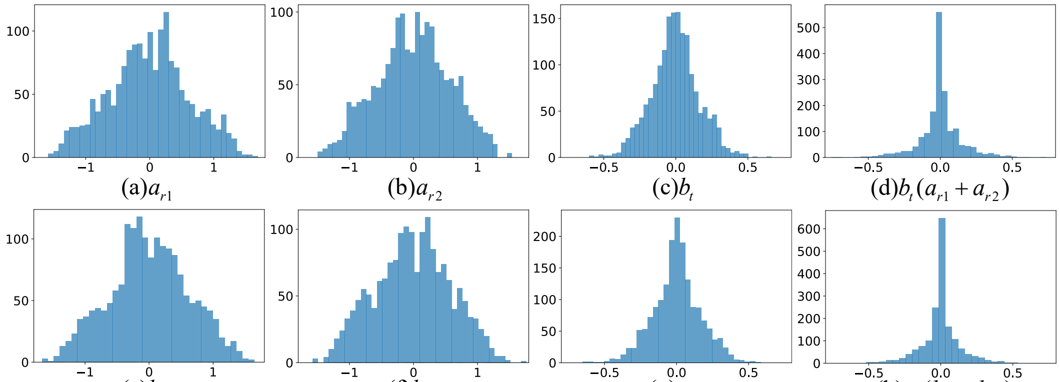

Figure 3: Histogram visualization of TComplEx inverse relation and timestamp. The relation r_1 is "Host a visit", the relation r_2 is "Make a visit", and the timestamp t is "2014-02-20". From Equation (14) we can get $b_t(a_{r1}+a_{r2})=0$ and $a_t(b_{r1}+b_{r2})=0$.

	ICEWS05-15			
	MRR	H@1	H@3	H@10
TComplEx_real	62.3	55.1	68.2	77.0
TComplEx_imag	62.1	54.9	67.6	76.8
TComplEx	**66**	**59**	**71**	**80**

Table 6: Evaluation results on ICEWS05-15 for TComplEx when the temporal complex embedding C_t has only real or imaginary parts.

A.2 Proof of Lemma 1

For TPComplEx, we can partially expand its scoring function

$$\begin{aligned}
\phi(s,r,o,t) &= \mathrm{Re}\left(\left\langle C_s+C_{t2}, C_r, \overline{C_o+C_{t3}}, C_{t1}\right\rangle\right)\\
&= \mathrm{Re}\left(\left\langle\left\langle C_s+C_{t2}, \overline{C_o+C_{t3}}\right\rangle, \langle C_r, C_{t1}\rangle\right\rangle\right)\\
&= \mathrm{Re}(\langle(a_sa_o+a_sa_{t3}+a_{t2}a_o+a_{t2}a_{t3}+b_sb_o+b_sb_{t3}+b_{t2}b_o+b_{t2}b_{t3}\\
&+(-a_sb_o-a_sb_{t3}-a_{t2}b_o-a_{t2}b_{t3}+b_sa_o+b_sa_{t3}+b_{t2}a_o+b_{t2}a_{t3})\mathbf{i}),\\
&(a_ra_{t1}-b_rb_{t1}+(a_rb_{t1}+b_ra_{t1})\mathbf{i})\rangle)\\
&= (a_sa_o+a_sa_{t3}+a_{t2}a_o+a_{t2}a_{t3}+b_sb_o+b_sb_{t3}+b_{t2}b_o+b_{t2}b_{t3})(a_ra_{t1}-b_rb_{t1})\\
&-(-a_sb_o-a_sb_{t3}-a_{t2}b_o-a_{t2}b_{t3}+b_sa_o+b_sa_{t3}+b_{t2}a_o+b_{t2}a_{t3})(a_rb_{t1}+b_ra_{t1}).
\end{aligned} \tag{15}$$

Proof of symmetry pattern. For the symmetry relations (s,r,o,t) and (o,r,s,t), we need to prove that $\forall s,o,t$, the following equations hold

$$\mathrm{Re}\left(\left\langle C_s+C_{t2}, C_r, \overline{C_o+C_{t3}}, C_{t1}\right\rangle\right)=\mathrm{Re}\left(\left\langle C_o+C_{t2}, C_r, \overline{C_s+C_{t3}}, C_{t1}\right\rangle\right). \tag{16}$$

Firstly, we expand the left term:

$$\begin{aligned}
&\mathrm{Re}\left(\left\langle C_s+C_{t2}, C_r, \overline{C_o+C_{t3}}, C_{t1}\right\rangle\right)\\
&=(a_sa_o+a_sa_{t3}+a_{t2}a_o+a_{t2}a_{t3}+b_sb_o+b_sb_{t3}+b_{t2}b_o+b_{t2}b_{t3})(a_ra_{t1}-b_rb_{t1})\\
&-(-a_sb_o-a_sb_{t3}-a_{t2}b_o-a_{t2}b_{t3}+b_sa_o+b_sa_{t3}+b_{t2}a_o+b_{t2}a_{t3})(a_rb_{t1}+b_ra_{t1}).
\end{aligned} \tag{17}$$

We then expand the right term:

$$
\begin{aligned}
&\text{Re}\left(\left\langle C_o + C_{t2}, C_r, \overline{C_s + C_{t3}}, C_{t1}\right\rangle\right) \\
&= (a_o a_s + a_o a_{t3} + a_{t2} a_s + a_{t2} a_{t3} + b_o b_s + b_o b_{t3} + b_{t2} b_s + b_{t2} b_{t3})(a_r a_{t1} - b_r b_{t1}) \\
&- (-a_o b_s - a_o b_{t3} - a_{t2} b_s - a_{t2} b_{t3} + b_o a_s + b_o a_{t3} + b_{t2} a_s + b_{t2} a_{t3})(a_r b_{t1} + b_r a_{t1}).
\end{aligned} \tag{18}
$$

Comparing Equation (17) and Equation (18), we can get

$$
\begin{cases}
a_{t2} = a_{t3} \\
b_{t2} = b_{t3} \\
a_r b_{t1} + b_r a_{t1} = 0.
\end{cases} \tag{19}
$$

A.3 Proof of Lemma 2

Proof of antisymmetry pattern. In contrast to symmetric relations, antisymmetric relations require that Equation (17) and Equation (18) are not equal. Based on this, we can obtain

$$
a_{t2} \neq a_{t3} \text{ or } b_{t2} \neq b_{t3} \text{ or } a_r b_{t1} + b_r a_{t1} \neq 0. \tag{20}
$$

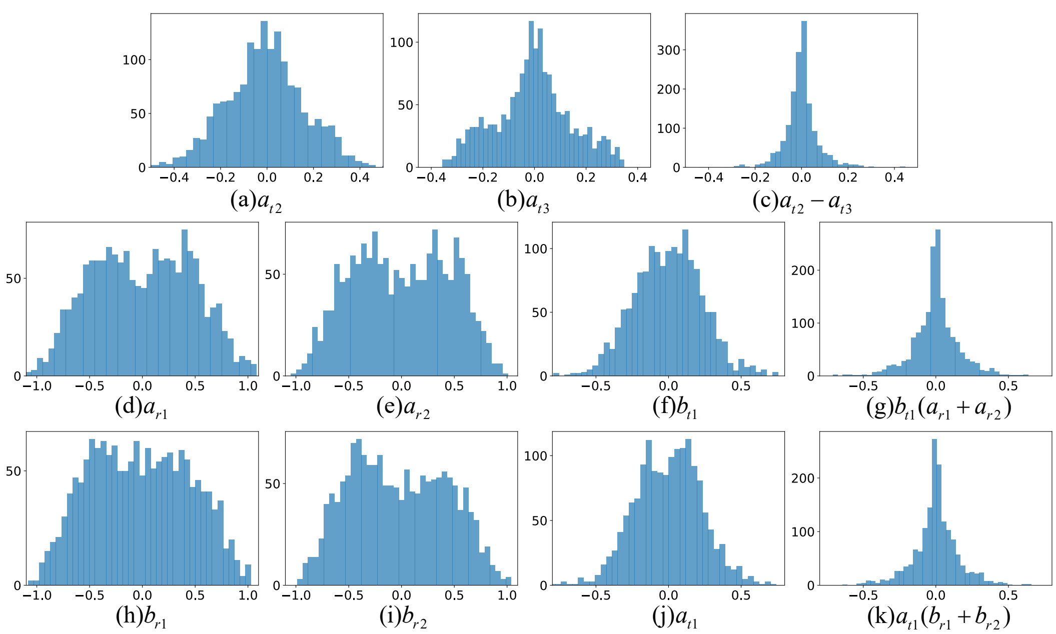

Figure 4: Histogram visualization of TPComplEx inverse relation and timestamp. The relation r_1 is "Host a visit", the relation r_2 is "Make a visit", and the timestamp t is "2014-04-28". From Equation (25) we can get $a_{t2} = a_{t3}$, $b_{t1}(a_{r1} + a_{r2}) = 0$ and $a_{t1}(b_{r1} + b_{r2}) = 0$.

A.4 Proof of Lemma 3

Proof of inverse pattern. For the inverse relations (s, r_1, o, t) and (o, r_2, s, t), we need to prove that $\forall s, o, t$, the following equations hold

$$
\text{Re}\left(\left\langle C_s + C_{t2}, C_{r1}, \overline{C_o + C_{t3}}, C_{t1}\right\rangle\right) = \text{Re}\left(\left\langle C_o + C_{t2}, C_{r2}, \overline{C_s + C_{t3}}, C_{t1}\right\rangle\right). \tag{21}
$$

Firstly, we expand the left term:

$$
\begin{aligned}
&\operatorname{Re}\left(\left\langle C_s + C_{t2}, C_{r1}, \overline{C_o + C_{t3}}, C_{t1}\right\rangle\right) \\
&= (a_s a_o + a_s a_{t3} + a_{t2} a_o + a_{t2} a_{t3} + b_s b_o + b_s b_{t3} + b_{t2} b_o + b_{t2} b_{t3})(a_{r1} a_{t1} - b_{r1} b_{t1}) \\
&- (-a_s b_o - a_s b_{t3} - a_{t2} b_o - a_{t2} b_{t3} + b_s a_o + b_s a_{t3} + b_{t2} a_o + b_{t2} a_{t3})(a_{r1} b_{t1} + b_{r1} a_{t1}).
\end{aligned} \tag{22}
$$

We then expand the right term:

$$
\begin{aligned}
&\operatorname{Re}\left(\left\langle C_o + C_{t2}, C_{r2}, \overline{C_s + C_{t3}}, C_{t1}\right\rangle\right) \\
&= (a_o a_s + a_o a_{t3} + a_{t2} a_s + a_{t2} a_{t3} + b_o b_s + b_o b_{t3} + b_{t2} b_s + b_{t2} b_{t3})(a_{r2} a_{t1} - b_{r2} b_{t1}) \\
&- (-a_o b_s - a_o b_{t3} - a_{t2} b_s - a_{t2} b_{t3} + b_o a_s + b_o a_{t3} + b_{t2} a_s + b_{t2} a_{t3})(a_{r2} b_{t1} + b_{r2} a_{t1}).
\end{aligned} \tag{23}
$$

Comparing Equation (22) and Equation (23), we can get

$$
\begin{cases}
a_{t2} = a_{t3} \\
b_{t2} = b_{t3} \\
a_t(a_{r1} - a_{r2}) - b_t(b_{r1} - b_{r2}) = 0 \\
b_t(a_{r1} + a_{r2}) + a_t(b_{r1} + b_{r2}) = 0.
\end{cases} \tag{24}
$$

Similarly, the satisfaction of Equation (24) requires a constraint on timestamp t, that is,

$$
\begin{cases}
a_{t2} = a_{t3} \\
b_{t2} = b_{t3} \\
a_t = 0 \\
a_{r2} = -a_{r1} \\
b_{r2} = b_{r1}
\end{cases}
\quad \text{or} \quad
\begin{cases}
a_{t2} = a_{t3} \\
b_{t2} = b_{t3} \\
b_t = 0 \\
a_{r2} = a_{r1} \\
b_{r2} = -b_{r1}.
\end{cases} \tag{25}
$$

Although C_{t1} degenerates to the real or imaginary part like TComplEx, TPComplEx still retains the timestamp embedding complex vector $C_{t2} = C_{t3}$, which is beneficial to the representation of other facts at the same timestamp. We visualize the TPComplEx embedding vector of inverse relation and the timestamp ((Barack Obama, Make a visit, Malaysia, 2014-04-28) and (Malaysia, Host a visit, Barack Obama, 2014-04-28)) in Figure 4, which is as expected from Equation (25). Compared with Table 6 and Table 7, TPComplEx shows a smaller decrease in performance when the temporal complex embedding C_{t1} has only real or imaginary parts.

	ICEWS05-15			
	MRR	H@1	H@3	H@10
TPComplEx_real	83.5	78.3	87.2	92.6
TPComplEx_imag	83.2	78.1	87.0	92.3
TPComplEx	**84.4**	**79.2**	**88.0**	**93.2**

Table 7: Evaluation results on ICEWS05-15 for TPComplEx when the temporal complex embedding C_{t1} has only real or imaginary parts.

Models	Parameters	ICEWS14	ICEWS05-15	YAGO15k	GDELT
ComplEx	$2r(\|E\| + 2\|P\|)$	1820	1860	1960	3820
TComplEx	$2r(\|E\| + \|T\| + 2\|P\|)$	1740	1360	1940	2270
TPComplEx	$2r(\|E\| + 3\|T\| + 2\|P\|)$	1594	886	1892	1256

Table 8: Embedding ranks for each model in different datasets.

A.5 Ablation Study

Table 9 shows the results of the ablation experiments. We mainly test the simplified model (refer to Equation (4)). Therefore, after removing the temporal bias of head or tail entity, the model becomes an asymmetric structure, which leads to a large decline in the experimental performance. And we can find that ICEWS14, ICEWS05-15 and GDELT have more head entity aggregation (retaining tail entity bias of C_{t3}, and $\mathrm{Re}\left(\left\langle C_s, C_r, C_{t1}, \overline{C}_{t3}\right\rangle\right)$ is used for head entity aggregation,). In contrast, YAGO15k has more facts about tail entity aggregation.

	ICEWS14				ICEWS05-15			
	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
TPComplEx_head	60.4	51.7	65.6	76.5	59.9	50.6	65.3	77.1
TPComplEx_tail	62.3	54.2	66.8	77.6	65.8	57.6	70.7	81.3
TPComplEx	**84.3**	**79.5**	**87.5**	**92.7**	**84.4**	**79.2**	**88.0**	**93.2**
	YAGO15k				GDELT			
	MRR	H@1	H@3	H@10	MRR	H@1	H@3	H@10
TPComplEx_head	44.3	37.2	47.7	58.0	20.4	14.4	21.3	31.5
TPComplEx_tail	36.3	28.0	38.6	54.5	35.8	27.4	38.4	52.1
TPComplEx	**57.3**	**51.5**	**60.0**	**68.4**	**40.1**	**32.1**	**42.6**	**55.6**

Table 9: Ablation study for TPComplEx on four datasets. Where TPComplEx_head means that only the temporal bias of the head entity is preserved, and TPComplEx_tail means that only the temporal bias of the tail entity is preserved.

A.6 Standard Deviations

Table 10 shows the standard deviations for the MRR computed over 5 runs of TPComplEx on all datasets.

	ICEWS14	ICEWS05-15	YAGO15k	GDELT
TPComplEx	0.0015	0.0027	0.0025	0.0039

Table 10: Standard deviations for the MRR on four datasets.