Temporal Knowledge Graph Completion with Approximated Gaussian Process Embedding

Linhai Zhang[1] **Deyu Zhou**[1*]
[1] School of Computer Science and Engineering, Key Laboratory of Computer Network and Information Integration, Ministry of Education, Southeast University, China
{lzhang472, d.zhou}@seu.edu.cn

Abstract

Knowledge Graphs (KGs) stores world knowledge that benefits various reasoning-based applications. Due to their incompleteness, a fundamental task for KGs, which is known as Knowledge Graph Completion (KGC), is to perform link prediction and infer new facts based on the known facts. Recently, link prediction on the temporal KGs becomes an active research topic. Numerous Temporal Knowledge Graph Completion (TKGC) methods have been proposed by mapping the entities and relations in TKG to the high-dimensional representations. However, most existing TKGC methods are mainly based on deterministic vector embeddings, which are not flexible and expressive enough. In this paper, we propose a novel TKGC method, TKGC-AGP, by mapping the entities and relations in TKG to the approximations of multivariate Gaussian processes (MGPs). Equipped with the flexibility and capacity of MGP, the global trends as well as the local fluctuations in the TKGs can be simultaneously modeled. Moreover, the temporal uncertainties can be also captured with the kernel function and the covariance matrix of MGP. Moreover, a first-order Markov assumption-based training algorithm is proposed to effective optimize the proposed method. Experimental results show the effectiveness of the proposed approach on two real-world benchmark datasets compared with some state-of-the-art TKGC methods.

1 Introduction

Knowledge Graphs (KGs) provide an efficient way to store world knowledge. Various KGs such as DBpedia (Auer et al., 2007), NELL (Carlson et al., 2010), YAGO (Suchanek et al., 2007) and Freebase (Bollacker et al., 2008) have been constructed and benefited downstream applications such as information retrieval, question answering, etc (Hao et al., 2017; Zhang et al., 2021). Generally, a fact in KG can be represented as a triple (s, p, o), where s (subject) denotes head entity node, o (object) denotes tail entity node and p (predicate) denotes relation edge between them. However, in the real world, some facts are time-aware. For example, the fact *(Joe Biden, presidentOf, the United States)* is not valid until January 21, 2021, the United States presidential inauguration of Joe Biden. Therefore, some KGs store time-aware facts or events as the quadruple (s, p, o, t), where t is the timestamp. Such KGs are referred as Temporal Knowledge Graphs (TKGs), which mainly include YAGO3 (Mahdisoltani et al., 2014), GDELT (Leetaru and Schrodt, 2013), Wikidata (Erxleben et al., 2014) and ICEWS (Lautenschlager et al., 2015).

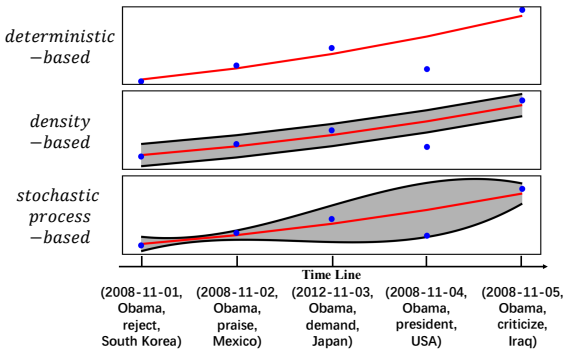

Figure 1: Illustration of three types of embedding-based TKGC methods which share the same embedding position function. Blue points denote the ideal position for the embeddings of entity $Obama$, red lines denote the embedding evolution function estimated by three types of methods, and shade areas denote the uncertainty estimated by density-based and stochastic process-based methods.

Temporal Knowledge Graph Completion (TKGC), aiming at inferring the missing edges based on known facts, is a fundamental task for the incomplete real-world TKGs. A large class methods perform TKGC by mapping nodes and edges in TKGs into high-dimensional latent feature spaces while preserving the semantic and structural

*Corresponding author.

Proceedings of the 29th International Conference on Computational Linguistics, pages 4697–4706
October 12–17, 2022.

information as much as possible. In recent years, extensive research have been conducted on embedding-based TKGC with notable process, which is also known as temporal knowledge graph embeddings (TKGEs). Early works extend the translation-based Knowledge Graph Embedding (KGE) approaches by mapping each timestamp into a specific time embedding (Leblay and Chekol, 2018; Dasgupta et al., 2018). Obviously, such approaches cannot deal with the unseen timestamps. To tackle this problem, some researchers model the entity and relation embeddings in TKGE as the continuous functions of time (García-Durán et al., 2018; Goel et al., 2020), which can be categorised as deterministic-based approaches. Recently, some researchers model TKGE as multivariate Gaussian density embeddings (Xu et al., 2019), which can be considered as density-based methods.

Although the notable progresses have been made, most of existing TKGE methods still suffer from the following two disadvantages. (1) Most of them are not flexible enough, the entity and relation embeddings are usually learned as the deterministic function of time, which is good at capturing the global trend while failing to model the surging local fluctuation. For example, as shown in Figure 1, the semantic meaning of $Obama$ should have a violent fluctuation on $2008-11-04$ as he won the presidential election, which is hard to model solely based on the deterministic function of time. (2) Most of them are not expressive enough, the temporal uncertainties of entity and relation embeddings are often ignored or under-fitted. As shown in Figure 1, existing deterministic-based methods often ignore the uncertainties while density-based methods learn embeddings with stationary uncertainties.

The above disadvantages could be naturally tackled by the stochastic process-based method. On the one hand, a stochastic process-based method is flexible to deal with the local fluctuations by modeling the correlations of embeddings at neighboured timestamps. On the other hand, a stochastic process-based method is expressive to model the dynamic changes of temporal uncertainties with the covariance matrix and kernel function.

Therefore, in this paper, we propose a novel method to learn flexible and expressive temporal knowledge graph embeddings based on approximated multivariate Gaussian processes (TKGC-AGP). In specific, each entity and relation in TKGs are mapped into a specific multivariate Gaussian process. The evolution dynamics of the entities and relations are modeled using the mean function of multivariate Gaussian process. The temporal correlation for each entity/relation are captured by the kernel function. The temporal uncertainty of the entities and relations is modeled by the entity/relation-specific covariance matrix. Further more, a first-order Markov assumption based algorithm is proposed to approximate the likelihood of multivariate Gaussian process. To investigate the effectiveness of the proposed approach, extensive experiments have been conducted on two large-scale TKG datasets. Experimental results show the effectiveness of the proposed approach compared to various competitive baselines.

In general, our contributions are listed as follows.

- A novel temporal knowledge graph embedding approach based on multivariate Gaussian process, TKGC-AGP, is proposed. Both the correlations of entities and relations over time and the temporal uncertainties of the entities and relations are modeled. To our best knowledge, we are the first one to utilize multivariate Gaussian process in TKGC.
- A novel first-order Markov assumption based algorithm is proposed to approximate the likelihood of multivariate Gaussian process.
- Experimental results show that TKGC-AGP outperforms several competitive baselines on two TKG datasets.

2 Related Work

Our work is mainly related to two lines of research, described as follows.

2.1 Temporal Knowledge Graph Completion

Temporal knowledge graph completion has been an attractive research topic in recent years. Works have been done with notable progress. Leblay and Chekol (2018) proposed TTransE, which extended the translation-based knowledge graph embedding methods to temporal knowledge graph by mapping the time information into low-dimensional vector space. Similar to TTransE, Dasgupta et al. (2018) proposed HyTE by incorporating the time information by assigning each timestamp with a temporal hyperplane. García-Durán et al. (2018)

proposed TA-TransE and TA-DistMult to learn the time-aware relation embedding by concatenating relation with time information as the input of a recurrent neural network. Xu et al. (2019) proposed ATiSE to represent the entity and relation in TKG as additive time series with Gaussian white noise to capture the temporal uncertainty. Goel et al. (2020) introduced the diachronic embedding method to model the evolution of entities along with time. Lacroix et al. (2020) presented an extension of ComplEx by introducing new regularization schemes to control the evaluation rate of embeddings. Xu et al. (2020) defined the temporal evolution of entities as the rotation in the complex vector space to deal with the symmetric and asymmetric relation simultaneously.

However, all aforementioned methods focus on modeling the evolution of entities and relations over time, ignoring the local correlations within them. To our best knowledge, we are the first one to consider model the correlation of entities and relations over long and short term in TKGs.

2.2 Probabilistic Representation Learning

Probabilistic embeddings have been extensively explored in many natural language processing tasks. Vilnis and McCallum (2015) introduced Gaussian embedding into the word representation learning task to tackle polysemy with the variance of Gaussian distributions. Brazinskas et al. (2018) further explored to learn the context-specific Gaussian word embeddings with a Bayesian learning framework. Athiwaratkun and Wilson (2018) proposed to learn the entailment relationships in the visual-semantic hierarchy with the Gaussian density order embeddings. Beyond probabilistic distribution, stochastic processes have also been considered. For example, Bamler and Mandt (2017) proposed a recursive stochastic process to model the dynamic changes of word semantics over time. To model uncertainty in KG, He et al. (2015) first employed Gaussian distribution to represent entities and relations in KG. Xiao et al. (2016) proposed a generative KGE method with a Bayesian non-parametric framework to generate Gaussian embedding and address the polysemy of relations. Other distributions such as Beta distribution have also been explored. Ren and Leskovec (2020) presented to model the first-order logic queries with Beta distributions by translating the logic operators with operations on Beta distribution.

However, all the aforementioned methods focus on specific-designed probabilistic distributions or stochastic processes. To our best knowledge, we are the first one to explicitly learn representations based on nontrivial multivariate Gaussian processes.

3 Method

In this section, we will discuss the details about the proposed method. We will start from the background knowledge of multivariate Gaussian process. Then we will talk about how to construct the entity and relation embeddings based on multivariate Gaussian process. Finally, the training and inference process of the proposed approach is explained.

3.1 Multivariate Gaussian Process

In this subsection, we will introduce the definition of multivariate Gaussian process and some common properties. We will start from matrix Gaussian distribution, which is the base for defining a multivariate Gaussian process.

Definition 1 (Matrix Gaussian Distribution)**.** A random matrix $\mathbf{X} \in \mathbb{R}^{n\times p}$ is matrix Gaussian distribution with location parameter $\mathbf{M} \in \mathbb{R}^{n\times p}$ and scale parameters $\mathbf{U} \in \mathbb{R}^{n\times n}$ and $\mathbf{V} \in \mathbb{R}^{p\times p}$ if and only if

$$vec(\mathbf{X}) \sim \mathcal{N}_{np}(vec(\mathbf{M}), \mathbf{V} \otimes \mathbf{U}) \tag{1}$$

where $\mathcal{N}_{np}(\mu, \Sigma)$ denotes multivariate Gaussian distribution on $\mathbb{R}^{n\times p}$ with mean vector μ and covariance matrix Σ, $vec(\mathbf{X})$ denotes the vectorization of $\mathbf{X}$ and $\otimes$ denotes the Kronecker product. In this case, we denote

$$\mathbf{X} \sim \mathcal{MN}_{n\times p}(\mathbf{M}, \mathbf{U}, \mathbf{V}) \tag{2}$$

With definition of matrix Gaussian distribution, we can define the multivariate Gaussian process.

Definition 2 (Multivariate Gaussian Process (MGP))**.** $\mathbf{f}$ is a multivariate Gaussian process on $\mathbb{R}^p$ with vector-valued mean function $\boldsymbol{\mu} = \{\mu_j\}_{i=1}^d : \mathbb{R}^p \Rightarrow \mathbb{R}^d$, kernel $k : \mathbb{R}^p \times \mathbb{R}^p \Rightarrow \mathbb{R}$ and positive semi-definite covariance matrix $\Omega \in \mathbb{R}^{d\times d}$ if and only if any finite collection of variables have a joint matrix Gaussian distribution,

$$[f(x_1), ..., f(x_n)] \sim \mathcal{MN}_{d\times n}(\mathbf{M}, \Omega, \Sigma) \tag{3}$$

where $\mathbf{M} \in \mathbb{R}^{d\times n}$ with $M_{ij} = \mu_j(x_i)$ and $\Sigma \in \mathbb{R}^{n\times n}$ with $\Sigma_{ij} = k(x_i, x_j)$. In this case, we denote

$$\mathbf{f} \sim \mathcal{MGP}(\boldsymbol{\mu}, k, \Omega) \tag{4}$$

3.2 Temporal Knowledge Graph Embedding based on Multivariate Gaussian Process

In this subsection, we describe how to construct the entity and relation embeddings based on MGP. Without loss of generality, we can denote a temporal knowledge graph as $\mathcal{G} = \{\mathcal{E}, \mathcal{R}, \mathcal{T}\}$, where $\mathcal{E}$, $\mathcal{R}$ and $\mathcal{T}$ are the set of entities, relations and timestamps respectively. Given a quadruplet (e_s, r_p, e_o, t) from $\mathcal{E} \times \mathcal{R} \times \mathcal{E} \times \mathcal{T}$, the goal is to learn temporal representations for $\{e_{i,t}|e_i \in \mathcal{E}\}$ and $\{r_{j,t}|r_j \in \mathcal{R}\}$ and a score function $f : \mathcal{E} \times \mathcal{R} \times \mathcal{E} \times \mathcal{T} \Rightarrow \mathbb{R}$ that is maximized for quadruplets in valid dataset D^+ and minimized for quadruplets in corrupted dataset D^-.

To model the temporal correlations and uncertainty simultaneous, entity or relation will be represented as a d-dim MGP on $t \in \mathbb{R}$, where d is the dimension of embeddings and t is time variable:

$$\begin{aligned} e_i(t) &\sim \mathcal{MGP}(\boldsymbol{\mu}_{e_i}(t), k_{e_i}(t,t'), \Omega_{e_i}) \\ r_j(t) &\sim \mathcal{MGP}(\boldsymbol{\mu}_{r_j}(t), k_{r_j}(t,t'), \Omega_{r_j}) \end{aligned} \tag{5}$$

From Definition 2, we know that an MGP can be fully specified by its mean function, kernel function and covariance matrix. We can define a MGP-based entity/relation embeddings by specifying those three components.

For the mean function, which controls the location and evolution trend of the embedding, we define it as a second order function of time variable t to make it more flexible:

$$\begin{aligned} \boldsymbol{\mu}_{e_i}(t) &= b_{e_i} + \alpha_{e_i}\phi_{e_i}t + \beta_{e_i}\psi_{e_i}t^2 \\ \boldsymbol{\mu}_{r_j}(t) &= b_{r_j} + \alpha_{r_j}\phi_{r_j}t + \beta_{e_i}\psi_{r_j}t^2 \end{aligned} \tag{6}$$

where $b_{e_i}, b_{r_j} \in \mathbb{R}^d$ are time-irrelevant bias vectors, $\alpha_{e_i}, \alpha_{r_j} \in \mathbb{R}$ are scalar first-order evaluation rates, $\phi_{e_i}, \phi_{r_j} \in \mathbb{R}^d$ are first-order evaluation direction vectors, $\beta_{e_i}, \beta_{r_j} \in \mathbb{R}$ are scalar second-order evaluation rates and $\psi_{e_i}, \psi_{r_j} \in \mathbb{R}^d$ are second-order evaluation direction vectors.

For the kernel function, it controls the correlation of the embeddings between different timestamps. The common choices of kernel functions for GP are various, such as white noise kernel, exponential quadratic kernel, rational quadratic kernel, etc. Here we assume that the correlations in TKG are mainly smooth and short-term, so we choose exponential quadratic kernel as the kernel function of TKGC-AGP. Formally we define:

$$\begin{aligned} k_{e_i}(t,t') &= \sigma_{e_i}^2 \exp(-\frac{\|t-t'\|^2}{2l_{e_i}^2}) \\ k_{r_j}(t,t') &= \sigma_{r_j}^2 \exp(-\frac{\|t-t'\|^2}{2l_{r_j}^2}) \end{aligned} \tag{7}$$

For the covariance matrixes that reflect randomness of entity (relation) in the real world, we set them as time-irrelevant diagonal matrixes for simplification:

$$\begin{aligned} \Omega_{e_i} &= diag(\omega_{e_i}) \\ \Omega_{r_j} &= diag(\omega_{r_j}) \end{aligned} \tag{8}$$

where $\omega_{e_i}, \omega_{e_i} \in \mathbb{R}^d$ denote the diagonal vectors of covariance matrixes, $diag(x)$ means making matrix with x as diagonal.

Given a quadruple q=(e_s, r_p, e_o, t), a translation-based score function is employed to measure the validity:

$$f(q) = f(e_s, r_p, e_o, t) = D_B(e_{s,t} - e_{o,t}, r_{p,t}) \tag{9}$$

where $D_B(d, d') \in \mathbb{R}$ is the Bhattacharyya distance between distribution d and d', $e_{s,t}, e_{o,t}, r_{p,t} \in \mathbb{R}^d$ are multivariate Gaussian distribution embeddings specific for timestamp t generated by the corresponding MGP:

$$\begin{aligned} e_{i,t} &= \mathcal{N}_d(\boldsymbol{\mu}_{e_i}(t), \Omega_{e_i}) \\ r_{j,t} &= \mathcal{N}_d(\boldsymbol{\mu}_{r_j}(t), \Omega_{r_j}) \end{aligned} \tag{10}$$

3.3 Approximation of MGPs and Training

In this subsection, we will describe the learning process of TKGC-AGP. To improve the robustness of training, it is common to train the model based on the valid dataset D^+ as well as a corrupted dataset D^- (Leblay and Chekol, 2018; Dasgupta et al., 2018). Following (Xu et al., 2019), a valid quadruple (s, p, o, t) is randomly corrupted by replacing the subject or object with a sampled entity from $\mathcal{E}$ to construct the corrupted dataset D^-. A common learning objective of probabilistic method is to maximize the joint likelihood of the data and the parameters, which is

$$\begin{aligned} &p(D^+, D^-, E, R) \\ &= p(D^+, D^-|E, R) \cdot p(E) \cdot p(R) \\ &= \prod_{t\in T} \prod_{q\in D_t^+} \prod_{q'\in D_t^-} \sigma(\gamma - f(q))\sigma(-\gamma + f(q')) \\ &\quad \prod_{e_i\in E} p(e_i) \prod_{r_i\in R} p(r_j) \end{aligned} \tag{11}$$

where E, R are the set of entity and relation embeddings respectively, $\sigma(\cdot)$ is Sigmoid function and γ is margin parameter. However, it is intractable to calculate the joint distribution $p(e_i)$ or $p(r_j)$ because they have a joint matrix Gaussian distribution over all possible timestamps t. That is,

$$\begin{aligned} p(e_i) &= p(e_{i,1}, e_{i,2}, ..., e_{i,t}) \\ p(r_j) &= p(r_{j,1}, r_{j,2}, ..., r_{j,t}) \end{aligned} \tag{12}$$

To approximate these joint distributions, we take a first-order Markov assumption that the current state of one entity or relation embedding is only depended on the last state of it. That is,

$$p(e_{i,t}|e_{i,t-1}, e_{i,t-2}, ..., e_{i,1}) = p(e_{i,t}|e_{i,t-1}) \tag{13}$$

To approximate the joint distributions of the entity or relation at adjacent timestamps $p(e_{i,t}, e_{i,t+1})$ or $p(r_{j,t}, r_{j,t+1})$, we further approximate the effect of this likelihood with a l_2 norm between kernel function $k_{e_i}(t, t+1)$ or $k_{r_j}(t, t+1)$ and the distances between their embeddings at the adjacent timestamps $D_B(e_{i,t}, e_{i,t+1})$ or $D_B(r_{j,t}, r_{j,t+1})$.

Then the learning objective can be transformed in minimizing the approximated negative log likelihood, which could be decomposed into four parts,

$$\begin{aligned} l_1 &= \sum_{t\in T} \sum_{q\in D_t^+} -\log\sigma(\gamma - f(q)) \\ l_2 &= \sum_{t\in T} \sum_{q'\in D_t^-} -\log\sigma(-\gamma + f(q')) \\ l_3 &= \sum_{t\in T} \sum_{e_i\in E} \|D_B(e_{i,t}, e_{i,t+1}) - k_{e_i}(t, t+1)\|^2 \\ l_4 &= \sum_{t\in T} \sum_{r_j\in R} \left\|D_B(r_{j,t}, r_{j,t+1}) - k_{r_j}(t, t+1)\right\|^2 \end{aligned} \tag{14}$$

where l_1, l_2 are the losses for valid dataset D^+ and corrupted dataset D^-, l_3, l_4 are the approximations to the log likelihood for entity and relation embeddings. We train the model by adding those losses together.

$$l = l_1 + \lambda_2 \cdot l_2 + \lambda_3 \cdot l_3 + \lambda_4 \cdot l_4 \tag{15}$$

where $\lambda_2, \lambda_3, \lambda_4$ are weights for losses.

To align the embedding across time, inspired by (Kumar et al., 2019), we further propose a time-batch training strategy, which takes the data between timestamp t and timestamp $t + L$ as a batch, where L is the length of time window. The detailed learning algorithm of TKGC-AGP is shown in Algorithm 1.

Algorithm 1 Training of TKGC-AGP

Input: the entity set $\mathcal{E}$, the relation set $\mathcal{R}$, the arranged timestamp set $\mathcal{T}$, the valid dataset D^+, the negative sample rate η, the number of epoch n, the margin γ, the embedding dimension d, the length of time window L.

Output: the parameters for TKGC-AGP $P = \{b, \alpha, \phi, \beta, \psi, \sigma, l, \omega\}$.

```
randomly initialize P
for i = 1, ..., n do
    for t' ∈ T do
        D_t^+ ← {q = (e_s, r_p, e_o, t) | t ∈ [t', t' + L], q ∈ D^+}
        for (e_s, r_p, e_o, t) ∈ D_t^+ do
            D_t^- = +{(e_s^k, r_p, e_o^k, t)}_{k=1,...,η}
        end for
        Update P = {b, α, φ, β, ψ, σ, l, ω} w.r.t. Loss l
    end for
end for
```

3.4 Complexity Analysis

Though the structure of TKGC-AGP is complex, as shown in Table 1, the space complexity and time complexity of TKGC-AGP remains the same as that of most of static and temporal KGE methods. For the space complexity, the parameter space of TKGC-AGP comprises $P = \{b, \alpha, \phi, \beta, \psi, \sigma, l, \omega\}$. So the total number of parameters of TKGC-AGP is $8 \times (|\mathcal{E}| + |\mathcal{R}|) \times d$. Since the length of time window is constant, the time complexity is also constant with embedding dimension d.

Method	Space complexity	Time complexity
TransE	$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d)$	$\mathcal{O}(d)$
ComplEx	$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d)$	$\mathcal{O}(d)$
TTransE	$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d + \|\mathcal{T}\|d)$	$\mathcal{O}(d)$
DE-SimplE	$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d)$	$\mathcal{O}(d)$
ATiSE	$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d)$	$\mathcal{O}(d)$
TKGC-AGP	$\mathcal{O}(\|\mathcal{E}\|d + \|\mathcal{R}\|d)$	$\mathcal{O}(d)$

Table 1: Complexity analysis of some existing methods.

4 Experiments

In this section, we perform extensive experiments on link prediction to investigate the effectiveness of TKGC-AGP on two real-world TKG datasets compared to some state-of-the-art KGE methods and TKGC methods.

4.1 Dataset

Two standard benchmark datasets, ICEWS-14 and ICEWS05-15, for TKGC are employed for experiments, which are two subsets of the Integrated Crisis Early Warning System (ICEWS) dataset (Lautenschlager et al., 2015). ICEWS-14 includes events happened in 2014 while ICEWS05-15 includes events happened between 2005 to 2015. The fact stored in ICEWS follows the form (s, p, o, t) with specific time point, such as *(Barack Obama, investigate, Iraq, 2008-07-21)*. Following (Xu et al., 2019), we employ the filtered version of ICEWS-14 and ICEWS05-15. The detailed statistics of two datasets are listed in Table 2.

Dataset	ICEWS-14	ICEWS05-15
# Entities	6,869	10,094
# Relations	230	251
# Timestamps	365	4,017
# Training	72,826	368,962
# Validation	8,941	46,275
# Test	8,963	46,092

Table 2: Dataset statistics.

4.2 Baselines

We compare TKGC-AGP with the following baselines:

- **TransE** (Bordes et al., 2013): static method that considers relation as a translation between entities in the embedding space.
- **DistMult** (Yang et al., 2015): static method that deals with the problem of symmetric relation with a bilinear objective function.
- **CompleEx** (Trouillon et al., 2016): static method that maps entities and relations into complex space with tensor factorization technique.
- **RotatE** (Sun et al., 2019): static method that regards relation as the rotation in the complex space.
- **TTransE** (Leblay and Chekol, 2018): temporal method that extend TransE to TKG by mapping each timestamp as specific embedding.
- **HyTE** (Dasgupta et al., 2018): temporal method that extend TransH (Wang et al., 2014) to TKG by learning time-specific hpyerplanes.
- **TA-TransE** (García-Durán et al., 2018): temporal method that employs a LSTM to encode the time information into relation representations.
- **DE-SimplE** (Goel et al., 2020): temporal method that represents entities with diachronic embeddings.
- **ATiSE** (Xu et al., 2019): temporal method that maps entities and relations as additive time series with Gaussian white noise.

4.3 Evaluation Metrics

In the link prediction experiment, following the previous literature (Goel et al., 2020), for each valid quadruple (e_s, r_p, e_o, t) in validation and test set, we generate query by masking the subject entity or object entity of it. Then we rank all the possible entities by filling the missing entity with candidate entity. Followed by previous work(Xu et al., 2019), we employ two kinds of metrics to evaluate the performance of all the methods, the **Mean Reciprocal Rank** (MRR), which is the average of reciprocal of the rank of golden entity and **Hit@K**, which is the frequency that the rank of golden entity is no greater than K.

4.4 Implementation Details

TKGC-AGP is implemented with PyTorch (Paszke et al., 2019). Part of results are taken from (Goel et al., 2020; Xu et al., 2019). The embeddings are trained with ADAM optimizer (Kingma and Ba, 2015) with learning rate = 0.001, maximum epoch = 1000, negative sample rate = 5, dimension of embedding = 100, length of time window = 3, margin = 1. All vector parameters are normalized to have unit l-2 norm.

4.5 Link Prediction Results

The link prediction results on the two dataset are shown in Table 3. It can be observed: 1) Some static methods outperform the temporal methods. For example, the performances of ComplEx and DistMulti are generally better than those of TTransE and HyTE. Capturing the basic structure of knowledge graph is still important for TKGC.

	ICEWS14				ICEWS05-15			
Metrics	MRR	Hit@1	Hit@3	Hit@10	MRR	Hit@1	Hit@3	Hit@10
TransE	0.280	0.094	-	0.637	0.294	0.090	-	0.663
DistMult	0.439	0.323	-	0.672	0.456	0.337	-	0.691
ComplEx	0.467	0.347	0.527	0.716	0.481	0.362	0.535	0.729
TTransE	0.255	0.074	-	0.601	0.271	0.084	-	0.616
HyTE	0.297	0.108	0.416	0.655	0.316	0.116	0.445	0.681
TA-TransE	0.275	0.095	-	0.625	0.299	0.096	-	0.668
TA-DistMult	0.477	0.363	-	0.686	0.474	0.346	-	0.728
DE-TransE	0.326	0.124	0.467	0.686	0.314	0.108	0.453	0.685
DE-SimplE	0.526	0.418	0.592	0.725	0.513	0.392	0.578	0.748
ATiSE	0.550	0.436	0.629	**0.750**	0.519	0.378	0.606	0.794
TKGC-AGP	**0.561**	**0.458**	**0.631**	0.738	**0.532**	**0.398**	**0.621**	**0.797**

Table 3: Link prediction results on ICEWS14 and ICEWS05-15 datasets. Bold values indicate the best-performing models under corresponding settings.

2) With temporal information, some KGE methods perform better on TKG. For example, DE-SimplE and TA-DistMult generally outperform other methods except ATiSE and TKGC-AGP. 3) The time series or stochastic process based methods achieve the best performance. TKGC-AGP outperforms ATiSE on all meatrics except Hit@10 for ICEWS14. The improvement is mainly attributed to the correlation modeled by the kernel function of MGP, although ATiSE employs a Gaussian white noise component to model temporal uncertainty.

4.6 Ablation Study

In this section, we analyze how the hyperparameters and the key components of TKGC-AGP affect the final performance. We focus on embedding dimension, length of time window and kernel function.

4.6.1 Effect of Embedding Dimension

The embedding dimension is one of the important hyperparameters for the representation learning methods. On the one hand, too small embedding dimension prevents methods to encode sufficient information. On the other hand, too large embedding dimension leads to the time and computation overhead, which is critical for TKGC with over ten thousands parameters. In this part, we evaluate the performance of TKGC-AGP under different embedding dimension $(50, 100, 200, 300, 400, 500)$ on ICEWS14 dataset. The results are shown in Figure 2.

It can observed that in general the performance of TKGC-AGP on ICEWS14 dataset is increasing at first and then decreasing with peak at 100 dim.

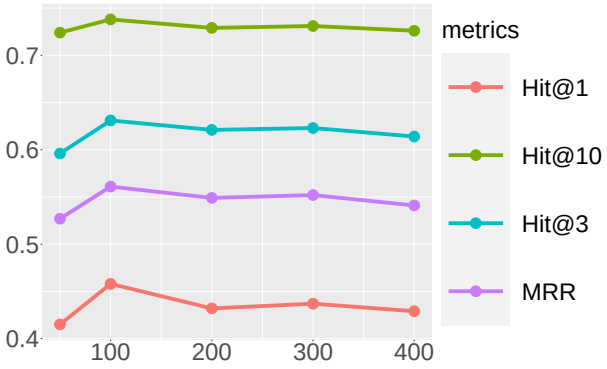

Figure 2: The performances of TKGC-AGP on ICEWS14 dataset under different embedding dimension setting.

The reason may be that when the dimension is too small, the embeddings cannot encode sufficient information while when the dimension is too large, the embeddings become too sparse to learn from the dataset. It should be pointed out that the optimal dimension of TKGC-AGP is generally smaller than that is other KGE or TKGC methods. It could be attributed to the complex structure of TKGC-AGP where there is no need for large dimension to encode sufficient information.

4.6.2 Effect of Length of Time Window

As described in Algorithm 1, the length of time window L is important for training TKGC-AGP. On the one hand, too small L will result in underfitted correlation. On the other hand, too large L will lead to more computational overhead and have the risk of over-fitting. To investigate the effect of the length of time window L on the performance of TKGC-AGP, experiments were conduct

Metrics	MRR	Hit@1	Hit@3	Hit@10
$L = 1$	0.551	0.439	0.627	0.733
$L = 2$	0.555	0.443	0.629	0.734
$L = 3$	**0.561**	**0.458**	**0.631**	**0.738**
$L = 4$	0.531	0.420	0.599	0.724
$L = 5$	0.526	0.415	0.595	0.727
$L = 10$	0.458	0.357	0.529	0.676

Table 4: The performances of TKGC-AGP with different lengths of time window on ICEWS14 dataset. Bold values indicate best-performing models under corresponding settings.

on ICEWS14 dataset with the length of time window L among $(1, 2, 3, 4, 5, 10)$. The results are shown in Table 4.

From the results, we can observe that in general the performance of TKGC-AGP on ICEWS14 dataset is firstly increasing as the length of time window increases and then quickly saturates at $L = 3$. It might be explained by the processing of TKGC-AGP from under-fitting to over-fitting. Especially when $L = 10$, the performance of TKGC-AGP is affected seriously because of over-fitting.

4.6.3 Effect of Kernel Function

As described in Section 3.1, the kernel function is an important component of MGP that controls the correlation across the index set. To investigate the effect of kernel function on the performance of TKGC-AGP, we perform experiments on ICEWS14 dataset with the following kernel functions besides exponential quadratic kernel. The results are shown in Table 5.

- **White noise kernel**, which means any two points from MGP are uncorrelated.

$$k(t, t') = \sigma^2 \mathbf{I}_n \tag{16}$$

- **Exponential quadratic kernel**, a smooth correlation decreasing with the distance between two points.

$$k(t, t') = \sigma^2 exp(-\frac{\|t - t'\|^2}{2l^2}) \tag{17}$$

- **Rational quadratic kernel**, which is similar to the exponential quadratic, when $\alpha \to \infty$, the rational quadratic kernel converges into the exponential quadratic kernel.

$$k(t, t') = \sigma^2 (1 + \frac{\|t - t'\|^2}{2\alpha l^2})^{-\alpha} \tag{18}$$

Metrics	MRR	Hit@1	Hit@3	Hit@10
TKGC-AGP-E	**0.561**	**0.458**	**0.631**	**0.738**
TKGC-AGP-W	0.548	0.435	0.626	0.732
TKGC-AGP-R	0.559	0.460	0.627	0.735
TKGC-AGP-P	0.479	0.370	0.532	0.685

Table 5: The performances of TKGC-AGP on ICEWS14 dataset with different kernel functions. Bold values indicate best-performing models under corresponding settings. -E, -W, -R, -P denotes TKGC-AGP with exponential quadratic kernel, white noise kernel, rational quadratic kernel and periodic kernel respectively.

- **Periodic kernel**, which allows to model periodic functions, where p denotes the period.

$$k(t, t') = \sigma^2 exp(-\frac{2}{l^2} sin^2(\pi \frac{\|t - t'\|}{p})) \tag{19}$$

It can be observed that TKGC-AGP with exponential quadratic kernel achieves the best performance. It should be pointed out that ATiSE (Xu et al., 2019) can be considered as a MGP-based method with white noise kernel. With white noise kernel, TKGC-AGP is similar to ATiSE. Therefore their performances are also very similar. Since rational quadratic kernel is very similar to exponential quadratic kernel, the performances of TKGC-AGP-E and TKGC-AGP-R are also very similar. For periodic kernel, it has the worst performance, we attribute this to little periodic pattern in the experiment dataset.

5 Conclusion

In this paper, we proposed TKGC-AGP, a novel temporal knowledge graph completion method based on approximated Gaussian process embeddings. With the flexibility and capacity, we can naturally model the global trends of entity and relation embeddings as well as the surging local fluctuations. Moreover, the temporal uncertainties can be also naturally modeled with the kernel function and corvariance matrix of MGP. To training TKGC-AGP, we employ the fisrt-order Markov assumption to approximate the joint distribution of MGP as well as a time-batch-based training strategy to align the embeddings across the time. The experimental results demonstrate that the proposed method outperform various static and temporal KGE baselines. Further work could be done by taking a Bayesian perspective to learn the proposed method.

6 Acknowledgement

The authors would like to thank the anonymous reviewers for the insightful comments. This work was funded by the National Natural Science Foundation of China (62176053).

References

Ben Athiwaratkun and Andrew Gordon Wilson. 2018. Hierarchical density order embeddings. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In *The semantic web*, pages 722–735. Springer.

Robert Bamler and Stephan Mandt. 2017. Dynamic word embeddings. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 380–389. PMLR.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1247–1250.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. *Advances in neural information processing systems*, 26.

Arthur Brazinskas, Serhii Havrylov, and Ivan Titov. 2018. Embedding words as distributions with a bayesian skip-gram model. In *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 1775–1789. Association for Computational Linguistics.

Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R Hruschka, and Tom M Mitchell. 2010. Toward an architecture for never-ending language learning. In *Twenty-Fourth AAAI conference on artificial intelligence*.

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

Fredo Erxleben, Michael Günther, Markus Krötzsch, Julian Mendez, and Denny Vrandečić. 2014. Introducing wikidata to the linked data web. In *International semantic web conference*, pages 50–65. Springer.

Alberto García-Durán, Sebastijan Dumančić, and Mathias Niepert. 2018. Learning sequence encoders for temporal knowledge graph completion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4816–4821, Brussels, Belgium. Association for Computational Linguistics.

Alberto García-Durán, Sebastijan Dumancic, and Mathias Niepert. 2018. Learning sequence encoders for temporal knowledge graph completion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4816–4821. Association for Computational Linguistics.

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

Yanchao Hao, Yuanzhe Zhang, Kang Liu, Shizhu He, Zhanyi Liu, Hua Wu, and Jun Zhao. 2017. An end-to-end model for question answering over knowledge base with cross-attention combining global knowledge. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 221–231, Vancouver, Canada. Association for Computational Linguistics.

Shizhu He, Kang Liu, Guoliang Ji, and Jun Zhao. 2015. Learning to represent knowledge graphs with gaussian embedding. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015*, pages 623–632. ACM.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Srijan Kumar, Xikun Zhang, and Jure Leskovec. 2019. Predicting dynamic embedding trajectory in temporal interaction networks. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 1269–1278.

Timothée Lacroix, Guillaume Obozinski, and Nicolas Usunier. 2020. Tensor decompositions for temporal knowledge base completion. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Jennifer Lautenschlager, Steve Shellman, and Michael Ward. 2015. ICEWS Event Aggregations.

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

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

Farzaneh Mahdisoltani, Joanna Biega, and Fabian Suchanek. 2014. Yago3: A knowledge base from multilingual wikipedias. In *7th biennial conference on innovative data systems research*. CIDR Conference.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Hongyu Ren and Jure Leskovec. 2020. Beta embeddings for multi-hop logical reasoning in knowledge graphs. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Fabian M Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: a core of semantic knowledge. In *Proceedings of the 16th international conference on World Wide Web*, pages 697–706.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 2071–2080. JMLR.org.

Luke Vilnis and Andrew McCallum. 2015. Word representations via gaussian embedding. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada*, pages 1112–1119. AAAI Press.

Han Xiao, Minlie Huang, and Xiaoyan Zhu. 2016. TransG : A generative model for knowledge graph embedding. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2316–2325, Berlin, Germany. Association for Computational Linguistics.

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

Chengjin Xu, Mojtaba Nayyeri, Fouad Alkhoury, Hamed Shariat Yazdi, and Jens Lehmann. 2020. Tero: A time-aware knowledge graph embedding via temporal rotation. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 1583–1593. International Committee on Computational Linguistics.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Linhai Zhang, Chao Lin, Deyu Zhou, Yulan He, and Meng Zhang. 2021. A bayesian end-to-end model with estimated uncertainties for simple question answering over knowledge bases. *Computer Speech & Language*, 66:101167.