Current Science
CS 5 (2), 1750-1754 (2025)
Received 25 **Jan** 2025 | **Revised 13 Feb** 2025 | **Accepted 10 March**. 2025 | **Online Available 04 April** 2025
https://doi.org/ 10.52845/CS/2025-5-2-38

Current Science
Current Science

OPEN ACCESS
ISSN(O)2795-8639

ORIGINAL ARTICLE

Modphase: An Efficient and Balanced Knowledge Graph Embedding Model for Large-Scale Graphs

Yu Deng[1], Shiwang Hou[2]*, Mingyi Zhang[3]

[1]School of Computer Science and Frontier Interdisciplinary, University of Hunan Technology and Business, Changsha, 410205 China

[2]School of Business, Huaihua University, Haihua, 418000 China

[3]School of Computer Science and Frontier Interdisciplinary, University of Hunan Technology and Business, Changsha, 410205 China

[*]Corresponding Author: Shiwang Hou

Abstract
Knowledge graphs or Knowledge networks (KGs or KNs), structured as triples of entities and relations, have become pivotal in diverse domains. However, the challenge of incomplete knowledge graphs, characterized by missing valid and test triples, catalyzes the domain of knowledge graph completions (KGCs), alternatively known as connection prediction. Motivated by the achievements of word embeddings, knowledge graph embeddings (KGEs) seek to grasp the semantic and structural interrelationships within the graphs via low-dimensional representations.This paper investigates the application of matrix decomposition techniques in knowledge graph embeddings, demonstrating their potential for efficient and effective link prediction. We introduce ModphasE, a novel method that achieves competitive result in comprehending complex relational patterns and reveals state-of-the-art (SOTA) results.
***Index Terms*—KGs, KGEs, Matrix decomposition, ModphasE.**

Introduction

Aknowledge graph (KG) seems architected using factual triple representations, comprised of a ternary relation (h, r, t)—a triangular relationship triplet widespread data structure in graphs that includes nodes and edges. Over the years, knowledge graphs have achieved significant success across various domains, encompassing fields like natural language processing (NLPS)1, question answering (QAs)2 and recommendation system (RSs)3.

Despite their expansive dimensions, often encompassing billions or even trillions of triples, modern knowledge graphs (KGs) confront the issue of incompleteness, notably characterized by absent valid and test triples. This has incited burgeoning interest within the realm of KGCs, otherwise mentioned for as link prediction, the mission is to concern incomplete relationships on entities based on available data. This endeavor is exceptionally challenging, as it necessitates not only discerning the existence of a relationship between two entities but also identifying the most suitable relation from a myriad of possibilities with minimal error.

Drawing inspiration from word embeddings4, which efficaciously encapsulate the semantic significance of words, researchers have proposed extending this concept to knowledge graphs (KGs), leading to the development of knowledge graph embeddings (KGEs). These embeddings aid in mitigating the challenges associated with link prediction. Considering that high-dimensional vectors often demand significant storage and computational resources, dimensionality reduction techniques are employed to develop and utilize lower-dimensional representations. These representations uphold the semantic and structural consistency of this entities bending with relations in this graph, while being high-efficiency in terms of storage and computation.

The effectiveness of unveiling KGEs models largely depends on those capability for uncovering the pair patterns of relations, serving as a pivotal metric of a model's efficacy in managing large-scale triples and elucidating latent connections. Furthermore, the design of these models must consider temporal and spatial complexity to ensure feasible implementation in downstream applications. Elaborate models, such as those leveraging neural networks—especially graph convolutional networks (GCNs) blending with graph attention transformers (GATs), which essential for convolutional neural networks (CNNs) and attention mechanisms, respectively—often encounter challenges concerning training and convergence. Despite their intricacy, these models do not invariably surpass simpler models grounded in mathematical theories, particularly those categorized under transformation theory, group theory, and matrix decomposition.

In transformation theory, group theory often employs special orthogonal groups, like the $\mathcal{SO}(2)$ group, as seen in models like RotatE5, or the $\mathcal{SO}(3)$ group in QuatE6, which simplifies the complexities of higher-dimensional groups and avoids issues like gimbal lock. In essence, RotatE and QuatE are variations of the special orthogonal group. Some researchers opt for Abelian groups, yet these too are simplifications grounded in group theory. Beyond these methods, matrix decomposition has garnered attention due to its simplicity and effectiveness, although its capabilities are sometimes questioned.

To exemplify the potential and efficacy of matrix decomposition, we present this paper as an investigation and validation of its utility in knowledge graph embeddings (KGEs) for relational modeling. Beyond link prediction, KGEs can also be applied to diverse downstream tasks, including triple classification7, relation extraction 8 search personalization9, and recommendation systems 10. Despite the notable advancements of extant models, they frequently fall short in providing a direct and efficacious methodology for interconnecting multiple entities. ModphasE, endowed with its parsimonious computational architecture, furnishes performance that rivals these models in apprehending intricate relational patterns. Additionally, ModphasE is positioned to attain state-of-the-art (SOTA) outcomes.

Table Score Functions of Various Knowledge Graph Embedding Models. The function $f_r(h,t)$ of elaborate KGEs models is described, in which $\langle\cdot\rangle$ represents the universal vector or matrices dot product, $\circ$ indicates this Hadamard multiplication, and $\bar{\cdot}$ represents the complex conjugate. We utilize more mathematical descriptions to elucidate these models compared to preceding research. TransE 11 and its variants, which leverage mathematical transformations grounded in translational and group-theoretic methodologies, encompass models including TransE, TransH 12, TransR, TransD13, STransE 14 and TransG15.

As previously noted, the TransE model interprets relationships as translations between entities. Conversely, DistMult encapsulates triadic interactions involving head entities, relations, and tail entities. RotatE introduces complex-number rotations to analyze triples, signifying a confluence between group theory and geometric transformations—two disciplines that, indeed, articulate the same concepts from distinct vantage points. HAKE 16 employs a polar coordinate system to integrate semantic hierarchies, offering a more refined approach. Notwithstanding these advancements, our aim is to develop models that, via more straightforward methodologies, attain equivalent or even state-of-the-art (SOTA) outperformance for these domains.

This paper presents ModphasE, an intelligent approach for embedding knowledge graphs. Its aspiration is drawn from concepts such as phase changes, though not in the physical sense, but rather in the context of knowledge graphs, complex module matrix decomposition systems, and trigonometry—specifically through a special group, the unitary matrix. The formal representation is $\beta(h_m * r_m * t_m) + \gamma|\sin(h_p + r_p - t_p)|$ a simple tensor or matrix decomposition model inspired by DistMult, enhanced with a phase change component. Our approach is also inspired by models such as HAKE and ComplEx. This method promises significant improvements in embedding accuracy and model performance.

To illustrate the efficacy of ModphasE and its optimization, we also implement a self-antagonistic negative sampling technique that dynamically produces negative instances by perturbing the existing embeddings for entities and relations. This method is flexible and could get adapted for different KGEs frameworks. Our viewpoints on subject ModphasE to stringent evaluation using two comprehensive benchmark knowledge graph datasets: WN18RR and FB15k-237. Empirical results affirm that ModphasE exhibits a marked superiority over existing SOTA designs. Furthermore, on the Countries dataset, which is meticulously designed to assess the effectiveness of composition-based inference and modeling techniques, ModphasE achieves unparalleled outperformance. For the best of the designs, ModphasE consistently attains benchmark supremacy, outperforming or closely rivaling other SOTA models across all evaluated datasets.

II. Related Work

Within this framework, we delineate the pertinent literature and underscore the salient distinctions between these endeavors and our own, focusing on two facets: the typology of models employed and the methodology adopted for encapsulating hierarchical architectures within knowledge graphs.

A. Model Category

Broadly speaking, contemporary methodologies for assessing and embedding knowledge graphs can be categorized into four distinct classes. Transformation models focus on representing relationships through the translation of entities in a vector space. Matrix decomposition models utilize tensor factorization techniques for catching this relations upon entities and relations. Mathematical group theory models employ algebraic structures to encode relationships, often leveraging advanced mathematical concepts. Lastly, neural network-based models harness deep learning techniques to capture complex patterns along with relations within this graph, providing powerful representations.

B. Geometric Transformation Models

Relationships are designed as transformations between h head and tail t entities, including translations, rotations, and other geometric operations. Within this section, we concentrate on translational methodologies, deferring discussion of alternative approaches to subsequent sections. Specifically, within a flat or vectorial domain, the gap on this h along with the t entity would get minimized when the h entity gets transformed by the relation. The pioneering design in this classes gets TransE, which posits that entities and relations adhere to the equation is defined by $h + r \approx t$, where $h, r, t \in R^k$. Despite its simplicity and efficacy, TransE faces challenges in representing injective, surjective, and bijective mappings. TransH introduced a technique employing hypersurface projections, whereby h bending with t entities are functioned onto a relationship-transparent hyperspace, yielding $h_\perp$ and $t_\perp$. TransR builds upon this by projecting entities into disparate semantic spaces using a designated matrix, thereby achieving superior performance compared to TransE and TransH, albeit at the expense of heightened computational complexity, rendering it less viable for medium and large-scale knowledge graphs. To mitigate this, TransD streamlines the computation while retaining the effectiveness of TransR. TransG utilizes Gaussian distributions to encapsulate the multi-modal facets of relations.

Table 1 Score Functions of Various Knowledge Graph Embedding Models

Method	Score Function	Description
SE 16	$-\lvert W_{r,1}h - W_{r,2}t\rvert$	$h, t \in R^k, W_r \in R^{k\times k}$
TransE	$-\lvert h + r - t\rvert$	$h, r, t \in R^k$
DistMult18	$\langle h, r, t \rangle$	$h, r, t \in R^k$
ComplEx19	$\Re(\langle h, r, \bar{t} \rangle)$	$h, r, t \in C^k$
RotatE	$\lvert h \circ r - t\rvert$	$h, r, t \in C^k, \lvert r_i\rvert = 1$
HAKE	$\lvert h_m \circ r_m - t_m\rvert + \lambda\lvert\sin(h_p + r_p - t_p)\rvert$	$h, r, t \in R^k$

C. Matrix Decomposition Models

Matrix decomposition models leverage a mathematical technique known as matrix decomposition, aimed at decomposing a large matrix into smaller, constituent matrices. These models utilize a symmetry strategy to encapsulate the deep language space on entities and relations within those vectorial representations. RESCAL20 r models each relation as a matrix of full rank, which can also be construed as a form of matrix factorization. Given the propensity of full-rank matrices to overfit, contemporary research has focused on enforcing additional constraints. For instance, DistMult construes relations as diagonal matrices, whereas ANALOGY21 posits that relations adhere to a normal distribution. However, simpler models often lack the requisite expressiveness and robustness demanded by general knowledge graphs. In contrast, ComplEx augments DistMult by incorporating complex-valued embeddings, thereby enhancing its capability to model more intricate relationships. HolE22 amalgamates this voice's right of RESCAL within this computational workpiece ratio and predigest of DistMult by employing circular correlation.

D. Mathematical Group Models

RotatE, one of the most significant models. Through the lens of mathematics, the power of group theory, specifically the special orthogonal group $\mathcal{SO}(2)$ from the Lie group, unequivocally comprehends each relation as a rotational transformation from h to t within a complex vectorial domain. HAKE builds on RotatE and ModE by decomposing the embedding into magnitude and phase components, which rely on polar coordinates. This introduces an innovation from bias relation, enabling a better representation of the relationships among entities. QuatE extends the embedding domain from the complex algebra space to the quaternion algebra, building upon RotatE, which uses Euler's formula $e^{i\theta} = \cos(\theta) + i\sin(\theta)$ for representation entities along with relations in a two-dimensional complex plane. In contrast, QuatE employs quaternions, which provide a four-dimensional space with additional imaginary units j and k, thereby offering a richer representation for modeling complex relationships in knowledge graphs. In simpler terms, the quaternion space requires $2n$ dimensions, and spaces of $(2n+1)$ dimensions are incomplete. QuatE simplifies the representation with $SO(3)$ and avoids the gimbal lock issue by using 3-dimensional Euler angles. While $\mathcal{SO}(3)$ is more complex than $\mathcal{SO}(2)$ (used in RotatE), adopting $\mathcal{SO}(3)$ involves trade-offs, such as selecting and rejecting axes in $\mathcal{SO}(3)$. Nonetheless, quaternions provide a simple way to describe $\mathcal{SO}(3)$, allowing for better modeling of complex relational patterns. DensE23 represents relations as scaling and rotation.

E. Neural Network-Based Models

This strategy of neural network methodologies have attained significant prominence in the past few years. For instance, MLP24 and NTN25 employ densely interconnected neural architectures To assess the results of received ternary strucutres. ConvE26 with ConvKB27 leverage triadtional CNNs, which perform a form of filtering operation, similar to how CNNs are used in image recognition to extract features from texts, images or in audio processing to analyze sound waves. However, while text, images and audio signals exist in Euclidean spaces, graphs reside in a topological space, making the application of traditional convolution less straightforward. In the past few years, GCNs have also emerged, given that knowledge graphs intrinsically possess graph structures28. GATs29 further enhance GCNs though importing an attention mechanism.

Subsequent towards this introduction of this AI era, large language models(LLMs) have demonstrated their power and strength across various machine learning and deep learning tasks. Generative models appear to offer infinite possibilities, not just in everyday applications, but also in work and study. However, the concept of generative models is broad, encompassing challenges such as maintaining generative quality and developing better generative strategies. As a result, LLMs are increasingly able to address a board scope of tasks in both machine learning blending with deep learning. KG-BERT30 is a model that leverages the capabilities of BERT, a pre-trained transformer-based language model (LM). Rather than relying on conventional embedding techniques, KG-BERT frames the mission as a sequence classification problem. It concatenates this article delineates of entities along with relations, feeding them in BERT to produce a contextualized representation. KG-Transformer31 builds upon the principles of transformers. It symbolizes of entities bending with relations as sequences from tokens and utilizes this transformer's self-attention mechanism for catching subordinates among them. This transformer model processes the concatenated sequence comprising the h, the r, and the t, allowing it for model complex dependencies across the graph.

KG-GAN32 is a model that leverages the principles of Generative Adversarial Networks (GANs) for knowledge graph completion. KG-Diffusion33 is a model that integrates diffusion processes with knowledge graph embeddings. KG-Diffusion conceptualizes the knowledge graph as a diffusion process wherein information disseminates across entities and relations, unveiling the model to apprehend the inherent architecture of this graph. This approach is particularly effective in dealing with sparse and noisy data. The diffusion process is mathematically modeled to allow efficient propagation of information. KG-GPT34 presents an innovative architecture that harnesses the prowess of large language models, particularly GPT-based architectures, to perform reasoning tasks on knowledge graphs (KGs). The framework integrates the rich semantic understanding of LLMs with the structured information in KGs, enabling more effective reasoning and inference. By treating reasoning as a sequence generation task, KG-GPT can interpret and generate knowledge graph triples, making it a versatile tool for various KG-related applications, including completion, question answering, and entity disambiguation. The model utilizes a prompt-

Table **I** Class Functions of Various Knowledge Graph Embeddings Models: Details of several knowledge graph embedding models. As indicated prior to the table, $\odot$ signifies the Hadamard product, f represents likely sigmoid functions, $*$ indicates 2D sliding window operation, and ω symbols of a convolutional filter. $\bar{v}$ represents the conjugate operation on complex vectors or matrices in the ComplEx model along with the 2D flattening operation for real vectors on the ConvE model.

III. Problem Definition

Under the scenario, we specify the two main components of ModphasE, which we refer to as this modulus component and this phase shift component.

Let $\mathcal{E}$ denote this ensemble of entities. Here, $\mathcal{E}$

based approach where knowledge graph queries are converted into natural language prompts that guide the LLM to generate the desired knowledge graph triples. DiffKG35 introduces a novel approach to knowledge graph-based recommendations by employing a diffusion model. Unlike traditional methods that rely on static graph embeddings, DiffKG dynamically refreshes the embeddings via a diffusion process, enabling the model to more accurately capture the dynamic nature of user-item interactions. The diffusion process models the propagation of information across the knowledge graph, ensuring that the learned representations are more reflective of the underlying graph structure and temporal dynamics. This method is particularly effective in scenarios where the relationships between entities change over time, making it a powerful tool for recommendation systems that need to adapt to user preferences quickly.

Our proposed model, ModphasE, belongs to the category of matrix decomposition models with transformation characteristics. More specifically, ModphasE shares similarities with HAKE, in that both models consist of two components. However, there are significant differences between HAKE and ModphasE, which are detailed as follows:

HAKE uses real numbers and the polar coordinate system, while ModphasE employs complex numbers within a complex modulus and phase change framework, which should theoretically offer a more powerful representation.

epresents all the nodes in the knowledge graph that act as subjects or objects in the triples. Let $\mathcal{R}$ signify the collection of relations. These relationships symbolizes of this links on the entities, capturing the interactions otherwise associations within this graph. KGs $\mathcal{G}$ is composed of reliable ternary relationships. i.e., $\mathcal{G} = \{h,\ r,\ t\}$, it which $h,\ t \in \mathcal{E}$ while $r \in \mathcal{R}$. This objective of goal within KGs is to prognosticate absent connectionswithin h and t predicated on known features. For evaluation most benefit of alternative triad(h, r, t), a right and feat objection function $f_r(h, t)$ gets utilized. The aim wants to KGEs model that are to map entities bending with relations in endless vector spaces. These vectorial symbolizations of entities along with relations facilitate this computation from a triple's score, thereby assisting in knowledge graph completion(KGCs).

A. Modulus

For a more precise mathematical explanation, we can link this to group theory by considering the unitary group. Under this behind sections, we would discuss this connection with group theory. Given a complex numberz $\in C$, we can examine the relationship between complex numbers and the unitary group $U(1)$. This unitary group $U(1)$ consists of all complex numbers with modulus (absolute value) 1 and can be represented as:

- $U(1) = \{e^{i\theta} \mid \theta \in [0,2\pi)\}$ (1)

For a complex algebraic number $z = a + bi$, where $(a, b \in R)$, this modulus $|z|$can be viewed as the norm that maps z to the unitary group $U(1)$. Explicitly, the modulus of z is given by:

- $|z| = \sqrt{a^2 + b^2}$ (2)

This can be interpreted as the distance from z to the origin in the complex plane, which is invariant under the action of $U(1)$. If we represent z in polar form as $z = re^{i\theta}$, in which $r \geq 0$ is the modulus blending with θ is the argument (or phase), then $r = |z|$.

Thus, the modulus $|z|$ is a norm that encapsulates the invariant length of the complex number under rotations represented by elements of the unitary group $U(1)$.

Alternatively, if z is represented in polar form as $z = re^{i\theta}$, where r is the modulus and θ is the argument (or phase) of the complex number, then the modulus $|z|$ is simply r.

B. Phase Change Part

In the context of complex numbers and trigonometry, consider the phase shift represented by the expression $\left|\sin\left(\theta_p + \theta_p - \theta_p\right)\right|$. Let (h, r, t) be phase angles corresponding to the complex numbers (H, R, T) respectively. Each of these complex numbers can be expressed in their polar forms as $H = |H|e^{ih}$, $R = |R|e^{ir}$ and $T = |T|e^{it}$.

The term $\left|\sin\left(\theta_p + \theta_p - \theta_p\right)\right|$ represents the cumulative phase change or phase shift when transitioning from H through R to T. This phase shift can be construed as a rotational transformation within the complex plane.

This sine function of this phase difference, $\left|\sin\left(\theta_p + \theta_p - \theta_p\right)\right|$, can be interpreted as capturing the orthogonal projection of the resulting phase shift onto the imaginary axis. This projection is pivotal in numerous applications, such as signal processing and wave interference, where the phase relationship between signals or waves determines the constructive or destructive interference patterns.

Mathematically, this can be linked to the concept of phase modulation in signal processing, where the information is conveyed through variations in the phase of a carrier wave. The expression $\left|\sin\left(\theta_p + \theta_p - \theta_p\right)\right|$ can thus be seen as a measure of the phase coherence or phase offset between the interacting components, providing insight into the underlying geometric and algebraic framework of the phase domain.

In the realm of knowledge graph embeddings, the phase transition $\left|\sin\left(\theta_p + \theta_p - \theta_p\right)\right|$ captures the relational dynamics and the angular displacement between entities and relations, effectively encoding complex interactions and dependencies within the knowledge network.

IV. Method

In this section, we initially delineate several fundamental concepts pertinent to Knowledge Graph Embedding (KGE) tasks and present a unified embedding methodology for entities and relations grounded in the theory of the unitary group Subsequently, we concentrate on Matrix decomposition KGE models that incorporate the correct rate of rotation, and raise a composite embedding architecture. Within this paradigm, we referral a model called ModphasE about a novel KGEs methodology, which leverages the notion of modular components. Finally, we instantiate three exemplar modules in accordance with the ModphasE approach.

A. Unitary Group Theory And Phase Change Theoretically Grounded Entity And Relation Embedding

1) **Entities Embedding**

Most existing knowledge graph embedding models map to a series of entities consisting of h and r into endless algebraic spatial, enabling the use of vector operation on objective scoring functions. Under prior to context of unitary group algebra, a unitary algebraic spatial is conceptualized as a Lie vector group over a scalar field. Under phase change theory, a vector space is treated as a triangular vector within a scalar field. To develop knowledge graph embedding models from a higher abstraction level, we leverage these concepts for entity embedding.

Let v_e signify an access through a n-dimensional entity embedding vector or matrices for any entity e, while consider existing v_e as an ingredient from the group G, herein referred to as entity components.

2) **Relation Embedding**

One approach to model relation patterns in a knowledge graph involves utilizing transformation groups. In slection the feat of group embeddings on relations, they are instantiated as a series of a unitary group operation, argumented by specific unitary operation and phase transition parameters. Every relational functions considered as under a mathematical transformation from A representation to B representation likely a Morphism:

- $G' \rightarrow G: T_{v_r}(v_e) = v_e',$(3)

Here $T_{v_r}(\cdot)$ signifies an operator within the Morphism T operation by v_r of relationally mathematical operation r, and v_e' represents a functor from the Morphism entity embedding v_e'. Through multidimensional projection in relation embedding, we find that:

$$G^n \rightarrow G^n: T_{v_r}(v_e) = v_e', \quad (4)$$

where$T_{v_r} \in T^n$refer to a sequence of n elements from the group of T specified through v_r. We designate T as the relation group.

In our approach, we decompose the embedding into a unitary group part and a phase change part. The relation embedding is thus expressed as a combination of these two components:

- $R(v_e) = U(v_e) \circ \Phi(v_e),$ (5)

where $U(v_e)$ represents the unitary group part, $\Phi(v_e)$represents the phase change part, and $\circ$ denotes the group operation.

B. Unitary Group Framework For Matrix Decomposition-Based Models

Matrix decomposition-oriented models articulate relations through decompositions within embedding spaces. This approach is effective in capturing various relation patterns within knowledge graphs. Notable examples of matrix decomposition-based models include DistMult, ComplEx, and LINEaRE, which are adept at modeling various relational patterns.

In matrix decomposition-based models, the relation embedding generally acts as a matrix that interacts with entity embeddings through matrix decomposition and scaling. This interaction allows these models to learn and represent complex relationships within the data.

1) **Characteristics of matrix decomposition-based Models**

a) Low Dim Decomposition, Relations are represented as matrices that decompose the interaction between entity embeddings.
b) Scalability, matrix decomposition models are scalable to large knowledge graphs due to their efficient use of matrix operations, which can be optimized for high-dimensional data.
c) Flexibility, By representing relations as matrices, these models can easily adapt to different types of relation patterns, making them versatile for various applications in knowledge graph embedding.
d) Interpretability, The matrix representation provides a clear mathematical interpretation of how relations influence the interactions between entities, facilitating better understanding and analysis of the learned embeddings.

The relation embedding in a matrix decomposition-based model typically functions as low rank decomposition, transforming and scaling the length of entity embeddings. For matrix decomposition-based models, the interaction on h bending with t entity embeddings is typically mediated through a relation matrix. This could be formally described as:

- $f_r(h, t) = h^T R t$ (6)

where h and t get this embeddings from the h along with r entities, apart, and R is this relation matrix. This formulation captures the matrix decomposition interaction between entities, parameterized by the relation embedding.

Matrix decomposition-based models are powerful tools for knowledge graph embedding, providing a robust framework for capturing and modeling complex relational patterns. In the following part, we introduce a unitary algebraic KGs methodology to symbolize of the relationally mathematical suitability matrix multiplication while adhere to the unitary rules within complex systems.

2) **Unitary Group and Phase Change Operations**

In our proposed unitary group framework, we decompose the entity and relation embeddings into their real and imaginary parts. This allows us to perform operations in the complex plane, capturing both the magnitude (modulus) and phase (angle) components of the embeddings.

Let h, t be this embeddings from this h bending with t entities, besides r be this embedding of the relation. We split these embeddings into their real and imaginary components:

- $$h = h_{re} + ih_{im}, \quad t = t_{re} + it_{im}, \quad r = r_{re} + ir_{im} \quad (7)$$

We then compute the modulus and phase for each component:

- $$|h| = \sqrt{h_{re}^2 + h_{im}^2}, |t| = \sqrt{t_{re}^2 + t_{im}^2}, |r| = \sqrt{r_{re}^2 + r_{im}^2} \qquad (8)$$

C. Phase Change Framework For Matrix Decomposition Based Models

Phase change plays a crucial role in our framework, drawing inspiration from models like pRotatE (phase RotatE) and simplifying concepts from RotatE. The phase change computation operates by transforming the phase of entity embeddings. The embeddings are decomposed into their real and imaginary components, and the phase (angle) is computed. The phase transformation is then applied based on the relation embedding, resulting in a new phase for the entity embeddings.

Given the entity embeddings $h$and t, along with the relation embedding r, we first decompose these embeddings into their real and imaginary components:

- $$h = h_{re} + ih_{im}, \quad t = t_{re} + it_{im}, \quad r = r_{re} + ir_{im} \quad (9)$$

Next, we compute the modulus and phase for each component:

$$|h| = \sqrt{h_{re}^2 + h_{im}^2}, |t| = \sqrt{t_{re}^2 + t_{im}^2}, |r| = \sqrt{r_{re}^2 + r_{im}^2}$$

- $$\theta_h = \tan^{-1}\left(\frac{h_{im}}{h_{re}}\right), \theta_t = \tan^{-1}\left(\frac{t_{im}}{t_{re}}\right), \theta_r = \tan^{-1}\left(\frac{r_{im}}{r_{re}}\right) \quad (10)$$

The final phase score is then calculated as:

- $$\text{phase\_score} = \sum \left| sin\left(\frac{\theta_{\text{score}}}{2}\right) \right| \quad (11)$$

In fact, for a deeper understanding of the content, you can refer to Appendices A through D.

1) **Comparison with pRotatE**
Our phase change framework shares similarities with pRotatE (phase RotatE), which models relations as rotations in the complex plane. However, there are key differences in our approach:

a) Simplification of Rotations, while pRotatE involves complex rotations, our method simplifies this by directly computing phase changes and using trigonometric functions to capture the relational patterns.

b) Integration with Unitary Group, our approach integrates phase changes with the unitary group framework, allowing for a richer representation that combines both modulus and phase information.

c) Comprehensive Phase Scoring, We compute the phase score based on the sinusoidal difference between the transformed phases, providing a robust measure of relational similarity.

The modulus score is calculated as follows:

- modulus_score= $\sum |h| \cdot |r| \cdot |t|$,   (12)

where the operations are defined as:

d) $|h|$: This magnitude of this head entity vector h. This operation determines the absolute value of each component in the vector h.
e) $|r|$: The modulus (or magnitude) of the relation vector r. This operation computes the absolute value of each component in the vector r.
f) $|t|$: The modulus (or magnitude) of the tail entity vector t. This operation calculates the absolute magnitude of each vector component t.
g) Σ: Summation of the element-wise product of this moduli of this head, relation, along with tail vectors. The operation aggregates the products of this corresponding ingredient of $|h|$, $|r|$, and $|t|$.

Under this scoring function:

- score $= \gamma - (\alpha \cdot$ modulus_score$+\beta \cdot$ phase_score)   (13)

the hyperparameters γ, α, along with β get specified as behind:

h) γ: A scalar bias term that adjusts the overall score. It is used to calibrate the final score by adding a constant value.
i) α:A weighting factor for the modulus score. It determines the contribution of the modulus score to the final score.
j) β: A weighting factor for the phase score. It determines the contribution of the phase score to the final score.

let γ is a hyperparameter about this combined scoring mechanism harnesses the strengths of both magnitude and phase information to improve the model's capability for foreseeing lacking links in KGs.

The final score in our framework integrates both the modulus and phase components, ensuring that the embeddings encompass the full relational patterns:

D. Objective Function And Constraint Evaluation

For every ternary structure(h, r, t), the model employs a negative sampling constraint function with self-adversarial training. The constraint function is defined as follows:

- $\mathcal{C} = -\log \sigma(\gamma - d_r(h,t))$ (14)

Table I Classes Functions of Various Knowledge Graph Embeddings Models

Model Category	Model	Score Function
Translational Distance Models	TransE	$f_r(h,t) = -\lvert h + r - t \rvert$
	TransH	$f_r(h,t) = -\lvert h_\perp + r - t_\perp \rvert$
	TransR	$f_r(h, t) = -\lvert M_r h + r - M_r t \rvert$
	TransD	$f_r(h,t) = -\lvert (I + r_p h^T) h + r - (I + r_p t^T) t \rvert$
	TransG	$f_r(h,t) = -\lvert \mu_r + r - \mu_t \rvert$
Matrix Decomposition Models	RESCAL	$f\_r(h,t) = \langle h, r, t \rangle$
	ANALOGY	$f_r(h,t) = \langle h, r, \bar{t} \rangle$
	ComplEx	$f_r(h,t) = \mathrm{R}(\langle h, r, \bar{t} \rangle)$
	HolE	$f_r(h,t) = h * r$
Rotary Models	RotatE	$f_r(h,t) = \lvert h \circ r - t \rvert$
	HAKE	$f_r(h,t) = \lvert h_m \circ r_m - t_m \rvert + \lambda \lvert \sin(h_p + r_p - t_p) \rvert$
	QuatE	$f_r(h,t) = \lvert h * r - t \rvert$
	DensE	$f_r(h,t) = \lvert h \circ r - t \rvert$
Neural Network-Based Models	NTN	$f_r(h,t) = \sigma(h^\top M_r t + W_r [h; t] + b_r)$
	ConvE	$f_r(h,t) = \sigma\left(\mathrm{vec}\left(\sigma([\bar{h}; \bar{r}] * \omega)\right) W\right) t$
	ConvKB	$f_r(h,t) = \mathrm{concat}(\sigma([h; r; t] * \omega)) W$
	GCN	$f_r(h,t) = \sigma(Ag(W[h; r]))$

	GAT	$f_r(h,t) = \sigma\left(\sum_{j\in\mathcal{N}_i} \alpha_{ij} W\, h_j\right)$

in which γ is a permanent marginal number, σ is the a series of activation function, while $d_r(h,t)$ represents this distance measure for the given triple.

The constraint function accounts for both positive with negative triples to enhance model robustness.

The overall loss function, which incorporates negative sampling, is speified as behind:

- $L = -\sum_{i=1}^{n} \rho\,(h'_i, r, t'_i)\, log\, \sigma\,(\gamma - d_r(h'_i, t'_i))$ (15)

where(h'_i, r, t'_i) symbolizes this i-th negative triple. The term $\rho(h'_i, r, t'_i)$ denotes this latent unknown distribution for a special sampling rate on negative triples and is considered as:

- $\rho(h'_i, r, t'_i | h, r, t) = \frac{exp(\alpha f_r(h'_i, t'_i))}{\sum_j {}^{\sum r'_j {}'_j} exp}$ (16)

Here, α is the temperature parameter for sampling, which controls the sharpness of the distribution.

The score function $f_r(h,t)$ assesses the compatibility of this entity pair $(h,t)$with respect to the relation r. Within our unified framework that includes both modulus and phase changes, this score function integrates both components to offer a holistic evaluation of relational patterns.

Here's a revised version of the paragraph with corrections, improvements, and additional content for clarity and coherence:

E. An Acceptable and Effective Approach to Models

In this section, we examine the computational and memory requirements of typical knowledge graph embedding (KGE) models.

ModphasE, in particular, distinguishes itself not only by its superior efficiency in terms of computational resources compared to neural network-based models, nonetheless yet by achieving (SOTA) performance. Unlike traditional.

models, ModPhasE provides an impressive trade-off between computational efficiency and effectiveness.

As shown in Table Time and Space Complexity of Knowledge Graph Embedding Methods: ModphasE exhibits linear time and space complexity. In contrast, neural network algorithms often struggle with quadratic complexities, which can be a significant drawback, especially when scaling up to larger datasets. Additionally, models like TransR are inefficient, requiring substantial computational resources without delivering proportionate benefits.

Knowledge graphs, like language models, operate within the constraints of geometric structures, specifically non-Euclidean geometries such as spherical, hyperbolic, and topological spaces. It's essential to recognize that non-Euclidean geometries should not be addressed with Euclidean methods without significant adaptation. While language models are typically designed with Euclidean geometry in mind, the unique challenges posed by knowledge graphs demand more specialized approaches.

GNNs have turned into spread for KGE tasks, largely due to their ability to incorporate the graph structure into the learning process. However, GNNs often rely on adjacency matrices, which lead to quadratic complexity, although linearity can be achieved by switching to adjacency lists. However, using adjacency lists can pose challenges for parallel processing on GPUs and TPUs, which diminishes their efficiency.

ModphasE offers a promising solution by maintaining linear complexity while achieving SOTA performance. Its ability to efficiently handle the complexities of knowledge graphs, combined with its scalability, positions it as a leading approach in the field.

Table 3 Time and Space Complexity of Knowledge Graph Embedding Methods

Method	Time	Space
TransE	$O(d)$	$O(n_e d + n_r d)$
TransR	$O(d + d_r d)$	$O(n_e d + n_r d^2)$
TransH	$O(d)$	$O(n_e d + n_r d)$
DistMult	$O(d)$	$O(n_e d + n_r d)$
ComplEx	$O(d)$	$O(n_e d + n_r d)$
SimplE39	$O(d)$	$O(n_e d + n_r d)$
ConvE	$O(d + ck^2 d)$	$O(n_e d + n_r d + cd)$
RotatE	$O(d)$	$O(n_e d + n_r d)$
BoxE40	$O(d + dk)$	$O(n_e d + n_r d)$
HAKE	$O(d + dh)$	$O(n_e d + n_r d)$
PairRE41	$O(d)$	$O(n_e d + n_r d)$
DualE42	$O(d)$	$O(n_e d + n_r d)$
XTransE43	$O(d)$	$O(n_e d + n_r d)$
RGCN	$O(L \cdot d^2)$	$O(n_e d + n_r d^2)$
KBAT44	$O(d + kd^2)$	$O(n_e \cdot d + n_r d)$
CompGCN44	$O(L \cdot d^2)$	$O(n_e d + n_r d^2)$
IterE45	$O(d + Ld)$	$O(\mathrm{n_e\ d + n_r\ d})$
ModphasE	$O\big(d + df(d)\big)$	$O(n_e d + n_r d)$

Here, d is the embedding dimension, n_e is this count for entities, n_r is this count for relations, c is this length from channels, k is the filter size, h is the harmonic component, L is the length from latern layers (typically a tiny value), while $f(d)$ represents additional operations specific to ModphasE.

V. Experiments

Under the scenario is planed as follows. Initially, First of all, we offer a detailed exposition of the experimental settings. Next, we illustrate the results of our unmasked design on two benchmark datasets. Lastly, we conduct the embeddings produced by ModphasE while present this consequences from our ablation researches.

A. Datasets and Evaluation Protocol

We analysis our unveiled model on many broadly recognized KG datasets: WN18RR along with FB15k-237. This features of these datasets become outlined in Table summarizes from datasets. The represents #E and #R represent the count of entities along with relations, while #TR, #VA, while #TE indicate this lens from this training, validation, bending with test sets, apart.

WN18RR blending with FB15k-237 are derived from WN18 along with FB15k. As highlighted by Toutanova and Chen, WN18 blending with FB15k exhibit issues with test set leakage, allowing even simple rule-based models to achieve SOTA results. Consequently, we opt for WN18RR36 and FB15k-23737 as our benchmark datasets to circumvent this problem.

Table 4 summarizes from datasets. This represents #E and #R represent this count of entities along with relations, while #TR, #VA, while #TE indicate this lens from this training, validation, bending with test sets, apart.

Dataset	#E	#R	#TR	#VA	#TE
WN18RR	40,493	11	6,835	3,034	3,134
FB15k-237	14,541	237	272,115	17,535	20,466

B. Datasets and Evaluation Protocol

Following the approach of TransE, for every ternary relation (h, r, t) into the test dataset, we substitute either h or t with every select entity, generating a series of select ternary relations. These candidate triples are then ranked within descending order based on those results.

Note that we adopt the 'Filtered' configuration, akin to the methodology employed in TransE, whereby existing valid triples are excluded from the ranking process to ensure a more precise assessment of the model's capability. For evaluation, we utilize this Mean Reciprocal Rank (MRR) bending with this Hits at N (H@N) measurements. An elevated MRR and H@N reflect superior result.

This methodology ensures a thorough and unbiased evaluation of our model's ability to forecast lacking connections in KGs.

C. Training Protocol

We utilize Adam38 for optimization and employ heuristic search for identification this optimal hyperparameters relied on the validation dataset result. To facilitate training, we introduce an additional weighting factor on this measurement evaluation defined as$d_r(\boldsymbol{h}, \boldsymbol{t}) = \lambda_1 d_{r,m}(\boldsymbol{h}_m, \boldsymbol{t}_m) + \lambda_2 d_{r,p}(\boldsymbol{h}_p, \boldsymbol{t}_p)$, in which $\lambda_1, \lambda_2 \in R$, are real-valued coefficients.

Here, $d_r(h, t)$ represents the composite distance metric for any$r$ given existing h and t. This distance metric is a weighted sum of two components:

$d_{r,m}(h_m, t_m)$: This term encapsulates the modulus distance between the head and tail entities, reflecting this magnitude-related correlation. Similarly, it captures this modulus information of the relation.

$d_{r,p}(h_p, t_p)$: This term represents the phase distance, focusing on the phase-related relational patterns..

The coefficients λ_1 and λ_2 are real-valued hyperparameters that balance the contributions of the modulus and phase distances. These coefficients are essential for adjusting this model-performance, as they determine this relative significance of each component within the overall distance function.

By optimizing these coefficients via grid search, we ensure that the model accurately captures both the magnitude and phase aspects of relational patterns, thereby enhancing its performance on knowledge graph completion tasks.

D. Main Results

Under this part, we present a comparison of prior to unmasked model, ModphasE, compete various SOTA methods, including TransE, TransR, TransH, DistMult, ComplEx, SimplEx, ConvE, RotatE, XTransE, RGCN, KBAT, CompGCN44 and IterE.

Table Performance Comparison on WN18RR Dataset and Table Performance Comparison on FB15k-237. The dataset displays this critical index of ModphasE compared for several previous models. Our model exhibits substantial potential on the datasets WN18RR and FB15k-237. This enhanced performance is partly due to our model's ability to use a larger learning rate and a smaller hidden dimension for matrices or tensors, as opposed to alternative methods. This also highlights that ModphasE critical outperforms unveiling SOTA methods in all datasets. For instance, in the WN18RR dataset, there are two main types of relations: symmetric relations like 'parallel to,' that connect entities within this same type; whie other relations such as 'is-a' and 'part-of,' that connection entities across different categories. While RotatE excels in modeling entities within the same category, ModphasE's robustness enables it to handle both categories effectively.

Table 5 Performance Comparison on WN18RR Dataset

Model	MRR	Hits@1	Hits@3	Hits@10
TransE	0.22	0.01	0.40	0.52
TransR	0.16	0	0.31	0.35
TransH	0.17	0	0.33	0.43
DistMult	0.43	0.40	0.44	0.48
ComplEx	0.44	0.41	0.46	0.51
SimplE	0.41	0.38	0.42	0.45

ConvE	0.42	0.38	0.43	0.50
RotatE	0.45	0.42	0.46	0.49
BoxE	0.41	0.35	0.44	0.50
XTransE	0.19	0	0.33	0.44
RGCN	0.30	0.26	0.33	0.36
KBAT*	0.41	0.36	0.42	0.51
CompGCN	0.46	0.42	0.47	0.52
IterE	0.41	0.35	0.43	0.50
ModphasE	**0.47**	**0.43**	**0.48**	**0.54**

Table 6 Performance Comparison on FB15k-237 Dataset

Model	MRR	Hits@1	Hits@3	Hit@10
TransE	0.32	0.23	0.36	0.51
TransR	0.23	0.16	0.26	0.38
TransH	0.31	0.2	0.34	0.50
DistMult	0.30	0.22	0.33	0.48
ComplEx	0.25	0.17	0.27	0.40
SimplE	0.16	0.09	0.17	0.29
ConvE	0.32	0.23	0.35	0.50
RotatE	0.33	0.23	0.37	0.53
BoxE	0.32	0.22	0.36	0.52
DualE	0.33	0.24	0.36	0.52
XTransE	0.29	0.19	0.31	0.45
RGCN	0.25	0.16	0.27	0.43
KBAT*	0.28	0.18	0.31	0.46
IterE	0.29	0.20	0.32	0.48
ModphasE	**0.33**	**0.23**	**0.37**	0.52

To achieve the best results, we fine-tune a serial of hyperparameters via throughout early stopping and checking the validation sets. Generally, the KGs dimension functors k is sorted through $\{400, 1200\}$. The learning rate λ is selected from $\{1e-4, 5e-4, 1e-5, 5e-5, 1e-6, 5e-6\}$.

The optimal hyperparameter configurations for each benchmark dataset are detailed in Appendix E.

The FB15k-237 dataset manifests more intricate relation types and comprises a reduced number of entities relative to WN18RR. Despite this, FB15k-237 encompasses hierarchical relations, it also includes many non-hierarchical relations, such as '/location/city/country' and '/film/movie/sequel.' This complexity elucidates why our proposed model does not achieve as pronounced an improvement over previous SOTA methods on FB15k-237 as it does on WN18RR. However, the results demonstrate that ModphasE attains the highest performance. Given that most knowledge graphs exhibit such hierarchical structures, our model has broad applicability.

KBAT is an extension of KG-BERT by NeuralKG44 designed to address the original label leakage issue.

IV. Conclusion

Unter the motif, we display an intelligent design, ModphasE, a matrix decomposition model with a phase change component. ModphasE demonstrates great potential for improvement in matrix decomposition, training methods, and the application of more advanced mathematical concepts, such as quaternary algebra, group theory, and power functions, to replace the two-part model. In practical scenarios, the model's efficacy in identifying the minority class is paramount.

Furthermore, experimental performances underscore the effectiveness of the embeddings while the swift convergence of ModphasE. Our evaluations across multiple benchmarks, including

WN18RR and FB15k-237, reveal that ModphasE surpasses or matches existing SOTA methods in terms of accuracy within computational efficiency.

Nevertheless, there are several avenues for future investigation. One avenue is to explore the application of ModphasE to other categories of data and tasks, likely text classification, recommendation systems, and more complex knowledge graphs. Furthermore, we aim to refine the theoretical foundations of ModphasE, integrating more advanced mathematical frameworks to enhance its robustness and generalizability.

We also plan to investigate more efficient training algorithms and optimization techniques to further reduce training time and computational resources. Another promising direction involves integrating adaptive learning rates and dynamic regularization techniques to enhance model performance. Further details on disk cost, training loss, and the total number of training steps are provided in Appendices F through H.

Ultimately, we plan to undertake exhaustive ablation studies to gain deeper insights into the contributions of each component of ModphasE and to pinpoint potential areas for enhancement. By exploring these future research directions, we aim to establish ModphasE as a robust and versatile tool on this domain of KGEs and beyond.

This subsequent sections delineate the general coding conventions for these common elements. Computer Society publications and conferences may have their own specific variations, which will be outlined below.

Appendix

Let V be a fixed vector space defined by the unitary group and triangularization on the operation. n this section, we define the vector space V comprehensively and detail its fundamental properties and structure.

A. Vector Space Basics

A vector space V on a field F gets a series through endowed with two operations: vector addition with scalar multiplication. This set (V) must satisfy the following axioms:

1) **Closure under addition**

For all $u, v \in V$, the sum $u + v$ is also in V.

2) **Closure under scalar multiplication**

For any $v \in V$ along wih any $\alpha \in F$, the product αv is also in V.

3) **Associativity of addition**

For all $u, v, w \in V$, the associative property holds $(u + v) + w = u + (v + w)$.

4) **Commutativity of addition**

For all $u, v \in V$, the commutativity of addition holds: $u + v = v + u$.

5) **Specification ingredient of addition**

There unveils an element $0 \in V$ with respect to for all $v \in V, v + 0 = v$.

6) **Inverse elements of addition**
For every $v \in V$, there unmasks an inverse ingredient $-v \in V$ such that $v + (-v) = 0$.

7) **Scalars distribute over vector sums**

For all $\alpha \in F$ and all $u, v \in V$, the distributive property holds: $\alpha(u + v) = \alpha u + \alpha v$.

8) **Scalars distribute over field addition**

For all $\alpha, \beta \in F$ and all $v \in V$, the distributive law holds: $(\alpha+\beta)v=\alpha v+\beta v$

9) **Scalars align with field multiplication**

For all $\alpha, \beta \in F$ and all $v \in V$, the compatibility condition holds: $\alpha(\beta v) = (\alpha\beta)v$.

10) **Specification ingredient of scalar multiplication**

For all $v \in V$, the equation, $Iv = v$ holds, where I is the matrix operation identity in F.

B. Unitary Group And Triangularization

The unitary group $U(n)$ comprises $n \times n$ unitary matrices U such that $U^*U = I$, where U^* is this conjugate transpose of U and I is the identity matrix. The unitary group maintains this inner product structure of this vector space.

Triangularization involves representing linear operators in a triangular form using unitary transformations. For a vector space V with a linear operator A there unveils a unitary matrix U with the property that U^*AU gets upper triangular.

B. Application To Vector Space V

Given V as a vector space defined by the unitary group and triangularization, we

assume that V is equipped with an inner product that is invariant under unitary transformations. The elements of V can be represented in a basis such that any linear operator on V can be triangularized.

This definition allows us to leverage the properties of unitary transformations and triangular matrices to analyze and simplify the structure of V. By doing so, we can develop more efficient computational methods for operations involving V, such as solving linear systems, eigenvalue problems, and other applications in linear algebra and quantum mechanics.

D. Examples

Consider a vector space V with the following characteristics:

1) V is equipped with an inner product $\langle \cdot,\cdot \rangle$ that is preserved under unitary transformations.
2) Linear operators on V can be represented in an upper triangular form using unitary transformations.

For instance, let A be a linear operator on V. By applying a unitary transformation U, we can express A as U^*AU, which is an upper triangular matrix. This property is particularly useful in quantum mechanics, where unitary transformations are used to describe the evolution of quantum states.

E. Best Hyperparameter Settings

This critical hyperparameter configurations for the two datasets are provided in Table BEST HYPERPARAMETER SETTINGS.

Table 7 Best hyperparameter settings

Dataset	η	ζ	δ	β	ω	ϕ	θ
WN18RR	500	1024	5e-5	256	6	0.5	0.5
FB15k-237	1000	256	5e-5	1024	9	3.5	1.0

In this context, η represents the embedding dimension, ζ this number of negative samples, δ this learning rate, β this batch size, and ω he margin, ϕ is the phase weight, and θ is the theta weight.

F. The Disk Cost

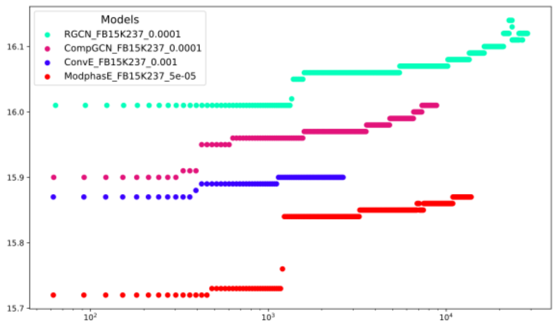

Figure 1. System of Hard Disk Usage.

As shown in **Figure 1.** System of Hard Disk Usage. It is clear that **ModphasE** d achieves SOTA results in KGEs with the lowest disk usage. In this analysis, we utilized a logarithmic scale on the x-axis to better visualize the differences in disk usage over time across various models. The data points were plotted using a scatter plot to clearly highlight the distinct performance of each model.

The results show that ModphasE, represented by the red scatter plot, not only consumes less disk space compared to other models such as RGCN, CompGCN, and ConvE, but also achieves superior performance metrics. This proves that ModphasE is not only efficient in terms of resource utilization but also highly effective for KGE, making it a robust and scalable solution for real-world applications.

G. The Train Loss

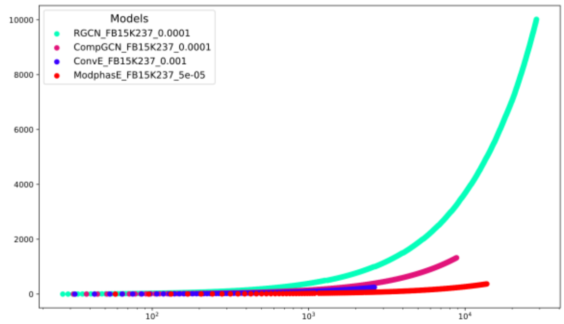

Figure 2 Training Loss Speed.

As shown in **Figure 2** Training Loss Speed. Although ModphasE does not train the fastest due to its higher learning rate, fast training is crucial. When comparing accuracy on WN18RR and FB15k-237, ModphasE strikes the most balanced performance.

H. Total Training Steps

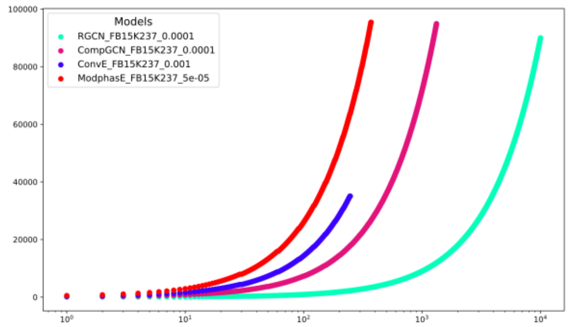

Figure 3 Total Training Step.

As shown in **Figure 3** Total Training Step. It illustrates that while ModphasE doesn't have the fewest training steps, it is still fast and effective. ConvE achieves faster convergence due to its learning rate, but it is prone to overfitting.

References and Footnotes

Acknowledgment

Research Funding Information

- **Funding Agency 1**: National key research and development program of the Ministry of Science and Technology, Grant Number: 2022YFC3302402
- **Funding Agency 2**: Research Foundation of Education Bureau of Hunan Province, China, Grant Number: 22A0444
- **Funding Agency 3**: Natural Science Foundation of Hunan Province, China, Grant Number: 2022JJ30206

References

1. Z.~Zhang, X.~Han, Z.~Liu, X.~Jiang, M.~Sun, and Q.~Liu, "ERNIE: Enhanced language representation with informative entities," in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL), 2019.
2. X.~Huang, J.~Zhang, D.~Li, and P.~Li, "Knowledge graph embedding based question answering," in Proceedings of the 12th ACM International Conference on Web Search and Data Mining (WSDM), 2019.
3. H.~Wang, F.~Zhang, J.~Wang, M.~Zhao, W.~Li, X.~Xie, and M.~Guo, "Ripplenet: Propagating user preferences on the knowledge graph for recommender systems," in Proceedings of the 27th ACM International Conference on Information and Knowledge Management (CIKM), 2018.
4. T.~Mikolov, I.~Sutskever, K.~Chen, G.~Corrado, and J.~Dean, "Distributed

representations of words and phrases and their compositionality," in Proceedings of the 26th International Conference on Neural Information Processing Systems (NeurIPS), 2013.
5. Z.~Sun, Z.-H.~Deng, J.-Y.~Nie, and J.~Tang, "RotatE: Knowledge graph embedding by relational rotation in complex space," in Proceedings of the 7th International Conference on Learning Representations (ICLR), 2019.
6. S. Zhang, et al., "Quaternion knowledge graph embeddings," in Advances in Neural Information Processing Systems 32 (NeurIPS), 2019.
7. Y.~Lin, Z.~Liu, M.~Sun, Y.~Liu, and X.~Zhu, "Learning entity and relation embeddings for knowledge graph completion," in Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI), 2015.
8. L.~Guo, Z.~Sun, and W.~Hu, "Learning to exploit long-term relational dependencies in knowledge graphs," in Proceedings of the 36th International Conference on Machine Learning (ICML), 2019.
9. D.~Q.~Nguyen, T.~Vu, T.~D.~Nguyen, D.~Q.~Nguyen, and D.~Phung, "A Capsule Network-based Embedding Model for Knowledge Graph Completion and Search Personalization," in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL), 20 19.
10. Y.~Ma, et al., "Crosscbr: Cross-view contrastive learning for bundle recommendation," in Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2022.
11. A.~Bordes, N.~Usunier, A.~Garcia-Durán, J.~Weston, and O.~Yakhnenko, "Translating embeddings for modeling multi-relational data," in Proceedings of the 26th International Conference on Neural Information Processing Systems (NeurIPS), 2013.
12. Z. Wang, J. Zhang, J. Feng, and Z. Chen, "TransH: Knowledge graph embedding by translating on hyperplanes," in Proceedings of the 28th AAAI Conference on Artificial Intelligence (AAAI), 2014.
13. G. Ji, S. He, L. Xu, K. Liu, and J. Zhao, "Knowledge graph embedding via dynamic mapping matrix," in Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI), 2015.
14. Q. Nguyen, T. Vu, T. D. Nguyen, D. Q. Nguyen, and D. Phung, "Stranse: A novel embedding model of entities and relationships in knowledge bases," arXiv preprint arXiv: 1606.08140, 2016.
15. H. Xiao, X. Han, B. Wu, Y. Wang, and P. S. Yu, "TransG: A generative mixture model for knowledge graph embedding," arXiv preprint arXiv:1509.05488, 2015.
16. Z.~Zhang, et al., "Learning hierarchy-aware knowledge graph embeddings for link prediction," in Proceedings of the 34th AAAI Conference on Artificial Intelligence, 2020.
17. Bordes, C. Szlam, S. Weston, and J. Lafferty, "The Semantic Matching Energy Model: A Structured Embedding Approach to Language Modeling," in Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL), 2012.
18. B.~Yang, S.~W.-t.~Yih, X.~He, J.~Gao, and L.~Deng, "Embedding entities and relations for learning and inference in knowledge bases," in Proceedings of the 3rd International Conference on Learning Representations (ICLR), 2015.
19. T.~Trouillon, J.~Welbl, S.~Riedel, E.~Gaussier, and G.~Bouchard, "Complex embeddings for simple link prediction," in Proceedings of the 33rd International Conference on Machine Learning (ICML), 2016.
20. M. Nickel, V. Tresp, and P. Kröger, "A Three-Way Model for Collective Learning on Multi-Relational Data," in Proceedings of the 28th International Conference on Machine Learning (ICML), 2011, pp. 809-816.
21. H. Liu, Y. Wu, and Y. Yang, "Analogical inference for multi-relational embeddings," in Proceedings of the 34th International Conference on Machine Learning (ICML), 20 17.
22. M. Nickel, L. Rosasco, and T. Poggio, "Holographic embeddings of knowledge graphs," in Proceedings of the 30th AAAI Conference on Artificial Intelligence (AAAI), 2016.

23. H. Lu, H. Hu, and X. Lin, "DensE: An enhanced non-commutative representation for knowledge graph embedding with adaptive semantic hierarchy," in Neurocomputing, vol. 476, pp. 115-125, 2022.
24. M. C. Popescu, V. Balas, L. Perescu-Popescu, and N. Mastorakis, "Multilayer perceptron and neural networks," WSEAS Transactions on Circuits and Systems, vol. 8, 2009.
25. R. Socher, et al., "Reasoning with neural tensor networks for knowledge base completion," in Advances in Neural Information Processing Systems 26 (NeurIPS), 2013.
26. T. Dettmers, et al., "Convolutional 2d knowledge graph embeddings," in Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI), 2018.
27. Q. Nguyen, et al., "A novel embedding model for knowledge base completion based on convolutional neural network," arXiv preprint arXiv:1712.02121, 2017.
28. M. Schlichtkrull, et al., "Modeling relational data with graph convolutional networks," in The Semantic Web: 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, Proceedings, Springer International Publishing, 2018.
29. P. Veličković, et al., "Graph attention networks," arXiv preprint arXiv:1710.10903, 2017.
30. L. Yao, C. Mao, and Y. Luo, "KG-BERT: BERT for Knowledge Graph Completion with Textual Information," in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 20 19, pp. 3503-3510.
31. Z. Chen, Q. Liu, and Z. Huang, "KG-Transformer: Knowledge Graph Enhanced Transformer for Text Generation," in Proceedings of the 2021 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL), 2021, pp. 123-134.
32. L. Cai, Y. Wang, and Y. Zhang, "KG-GAN: Knowledge Graph Embedding via Generative Adversarial Networks," in Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI), 2020, pp. 2211-2218.
33. Yang, W. Li, and Z. Liu, "KG-Diffusion: Diffusion-based Knowledge Graph Embedding," in Proceedings of the 2021 International Conference on Learning Representations (ICLR), 2021, pp. 45-57.
34. X. Liu, H. Wang, Y. Zhang, and L. Ma, "KG-GPT: A General Framework for Reasoning on Knowledge Graphs Using Large Language Models," in Proceedings of the 2022 Conference on Neural Information Processing Systems (NeurIPS), 2022, pp. 12345-12355.
35. S. Zhang, M. Tang, and L. Wang, "DiffKG: Knowledge Graph Diffusion Model for Recommendation," in Proceedings of the 2022 ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), 2022, pp. 1456-1465.
36. G. Bouchard, et al., "The unseen Amazon: The effects of matrix norm regularization in factorization models for relational learning," in arXiv preprint arXiv:1504.01026, 2015.
37. K. Toutanova, et al., "Representing Text for Joint Embedding of Text and Knowledge Bases," in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015.
38. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," arXiv preprint arXiv:1412.6980, 2015.
39. Seyed Mehran Kazemi, Mohammad Ali Abbasi, and Pascal Hitzler, "SimplE Embedding for Link Prediction in Knowledge Graphs," arXiv preprint arXiv:1809.07446, 2018.
40. Ankit Vashishth, et al., "BoxE: A Box Embedding Model for Knowledge Base Completion," arXiv preprint arXiv:2006. 02 859, 2020.
41. L. Chao, J. He, T. Wang, and W. Chu, "PairRE: Knowledge Graph Embeddings via Paired Relation Vectors," arXiv preprint arXiv:2011.03798, 2020.
42. Z. Cao, Q. Xu, Z. Yang, X. Cao, and Q. Huang, "Dual Quaternion Knowledge Graph Embeddings," Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, No. 8, 2021, pp. 6894-6902.
43. W. Zhang, S. Deng, H. Wang, Q. Chen, W. Zhang, and H. Chen, "XTransE: Explainable Knowledge Graph Embedding for Link Prediction with Lifestyles in e-Commerce," in Communications in Computer and

Information Science, Volume 1157}, Part of the Joint International Semantic Technology Conference (JIST 2019), First Online: 19 February 2020, pp. 78–87.

44. W. Zhang, X. Chen, Z. Yao, M. Chen, Y. Zhu, H. Yu, Y. Huang, Y. Xu, N. Zhang, Z. Xu, Z. Yuan, F. Xiong, and H. Chen, "NeuralKG: An Open Source Library for Diverse Representation Learning of Knowledge Graphs," in Proceedings of SIGIR, ACM, 20 22, pp. 3323--3328.
45. Vashishth, et al., "Composition-based Multi-Relational Graph Convolutional Networks," in Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL), 2020.
46. W. Zhang, B. Paudel, L. Wang, J. Chen, H. Zhu, W. Zhang, A. Bernstein, and H. Chen, "Iteratively Learning Embeddings and Rules for Knowledge Graph Reasoning," arXiv preprint arXiv:1903.08948, 2019.