This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3501735

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

DualRep: Knowledge Graph Completion by Utilizing Dual Representation of Relational Paths and Tail Node Density Insights

HAJI GUL[1], FERAS AL-OBEIDAT[2], ADNAN AMIN[3], MUHAMMAD WASIM[4], FERNANDO MOREIRA*[5]

[1 3]School of Computer Science and Information Technology, Institute of Management Sciences Peshawar Pakistan
[2]Zayed University Abu Dhabi,UAE
[4]Department of Computer Science, City University of Science and Information Technology, Peshawar
[5]REMIT, IJP, Universidade Portucalense, Porto, Portugal & IEETA, Universidade de Aveiro,Portugal

Corresponding author:Fernando Moreira (fmoreira@upt.pt)*

This work was supported by FCT – Fundação para a Ciência e a Tecnologia, I.P. [Project UIDB/05105/2020]

ABSTRACT Knowledge graphs (KGs) possess a vital role in enhancing the semantic comprehension of extensive datasets across many fields. It facilitate activities like recommendation systems, semantic searching, and intelligent data mining. However, lacking information can sometimes limit the usefulness of knowledge graphs (KGs), as the lack of relationships between entities could severely limit their practical application. Most existing approaches for KG completion primarily concentrate on embedding-based methods or just use relational paths, neglecting the valuable structural information offered by node density. This research presents an approach that effectively combines relational paths and the density features of tail nodes to enhance the accuracy of predicting relationships that are missing in knowledge graphs. Our method combines the sequential relational context represented by paths with the structural prominence indicated by node density, allowing for a dual view on possible entity connections. We validate the effectiveness of our technique by conducting comprehensive tests on many benchmark datasets, revealing substantial enhancements compared to conventional approaches. The Dual-Rep model, which incorporates relational paths and node density features, has continuously shown improved performance across several metrics, such as Mean Reciprocal Rank (MRR), Hit at 1 (Hit@1), and Hit at 3 (Hit@3). The DualRep model achieved a mean reciprocal rank (MRR) of 90.80. Additionally, it achieved a hit rate of 87.39 at rank 1 (Hit@1) and a hit rate of 91.18.

INDEX TERMS Knowledge Graph Completion, Relational Paths, Node Density Analysis,Graph Structural Features, Entity Relationship Prediction,Graph Neural Networks, Machine Learning in Knowledge Graphs, Entity Embeddings,, Relational and Structural Dynamics

I. INTRODUCTION

Knowledge Graphs (KGs) are crucial in artificial intelligence and data analysis. It provides a structured representation of real-world entities and their interrelationships. Networks represent complex structures associated with data points, allowing intelligent machines to comprehend and discover novel information about the environment around us. Knowledge graphs (KGs) have a wide range of applications, including improving semantic search by delivering search results that are aware of the context [1], and enabling recommendation systems to provide tailored content based on user preferences and content linkages [2]. Furthermore, in the domain of natural language comprehension, knowledge graphs (KGs) have a vital function in enhancing language models with factual information, hence enhancing the comprehension of context and the production of more logical and contextually appropriate replies [3]. Although knowledge graphs are useful, a major challenge in using them is knowledge graph completion (KGC), which involves deducing and including missing relations (edges) between entities (nodes). KGs are inherently limited; therefore, they ignore all potential information about objects and their associations. The lack of completeness is a difficulty for applications that depend on KGs for precise information retrieval and decision-making [4], [5]. Accurately forecasting these absent relationships is vital for maintaining the significance and usefulness of KGs in real-world applications. However, accomplishing this job is challenging because of the extensive and varied range of knowledge domains, the possibility of ambiguity in entity connections, and the sheer magnitude of data that has to be processed.

Research in Knowledge Graph (KG) completion has progressed from initial statistical techniques to more contemporary embedding-based methodologies. In these approaches, entities and interactions are expressed as continuous vectors to capture their semantic significance [6]. Embedding-based methods have been more popular recently [6], where they express entities and connections as vectors in continuous spaces. Models like TransE have demonstrated notable advancements in scalability and performance [7], DistMult [8], and Graph Convolutional Networks (GCNs) [9]. Owing to their effectiveness in managing huge datasets, these techniques, which can capture the semantic links between entities, have been extensively used for KGC tasks.

Even though embedding-based techniques have succeeded, they are not without pitfalls. However, these approaches fail to capture the complex, multi-relational structures in KGs. Models like TransE are limited in capturing complicated patterns like symmetric or many-to-many relationships because they presume that relationships can be modeled as translations in a vector space [7]. Similarly, further developed approaches like QuatE and RotatE concentrate mostly on pairwise interactions while attempting to depict more complex connections [10], [11]. The structural characteristics of the graph, such as node centrality and the relevance of higher-order structures, which are essential for comprehending the overall connection and significance of entities within the KG, are not properly utilized by these methods. The structural information represented in the graph is an important factor that is often ignored in these methods. In particular, a node's degree and its number of relationships are crucial factors in establishing its significance and impact on the graph. High-degree nodes frequently serve as key elements that link various network parts and participate in multiple interactions. The existing embedding-based methods usually fail to take node degree into meaningful consideration. For instance, graph convolutional networks (GCNs) combine data from adjacent nodes but do not particularly use node degree as a feature to improve relational predictions [9].

Because of these drawbacks, structural attributes such as node degree have recently been included in KGC research to enhance model performance. To evaluate the significance of adjacent nodes during aggregation, for example, Graph Attention Networks (GATs) employ attention processes and implicitly take node degrees into account [12]. This provides an incomplete solution. However, because these techniques depend on indirect aggregation, the processes influence high-degree nodes on connection predictions; they are still unable to fully recognize node degree as a direct characteristic in relationship prediction.

In this study, we propose an approach that integrates relational pathways and node degree information, consequently addressing the shortcomings of existing KGC approaches. Our method combines the structural knowledge offered by tail node degrees with the semantic information relational paths to improve the ability to predict existing models. Relational paths that capture direct and indirect links are sequences of connections between entities. These pathways provide important information about the underlying connection patterns within KGs. Node degree, however, provides a measure of centrality, a significant element in the graph. These two key elements of the graph work together to improve our model's ability to distinguish between entities with similar relational paths but different levels of centrality, which results in more precise predictions.

- Increase the accuracy of relationship predictions in knowledge graphs by combining relational path data with structural node properties.
- Enhance decision-making in relation to prediction tasks by taking into account the complex relationship between relational dynamics and the structural significance of nodes.
- Enhance the model's capacity to differentiate between potential relationships in cases when entities have similar pathways but vary in their centrality within the network, by using the node's degree as a distinguishing characteristic.

The rest of the sections of the paper are arranged as follows: A comprehensive review of the literature is provided in Section II which also highlights the gaps that motivate our study by summarising important works that are relevant to our problem. A thorough description of our proposed methodology including the techniques, algorithms, and experimental setup employed is given in Section III. A detailed explanation of the datasets used in the experiments and the evaluation criteria are also covered in this part. The findings of our studies are covered in detail in Section IV, which also includes an ablation study and a case study on model explainability using DualRep. Additionally, we compare our approach with other current methodologies to establish the usefulness of our methodology. In Section V, the study is finally concluded with a summary of the key findings and offering recommendations for future research directions.

II. RELTED WORK

GC presents a substantial challenge in Knowledge Graph Analysis (KGA) by explicitly predicting absent information within the knowledge graph. The missing information scould be related to the head $(?, r, t)$, the tail $(h, r, ?)$, or the relationship $(h, ?, t)$.The current literature presents many KGC methodologies.

knowledge graphs are becoming indispensable artifacts of present-day data science and artificial intelligence. They play a vital role in diverse fields like semantic web services, recommendation systems, AI reasoning, etc. Essentially, a knowledge graph is a well-structured representation of real-world things, their properties, and their intricate connections. In other words, knowledge graphs go beyond mere data storage; rather, they represent a network of interconnected information that enhances the ability of machines to comprehend and reason [13], [14]. By augmenting the contextual understanding of AI apps, knowledge graphs provide a semantic foundation that AI algorithms can leverage to discover trends and links easily. This feature significantly boosts the decision-making process since algorithms can draw more informed conclusions as they understand the semantic connections and context of the content. On the other hand, knowledge graphs are critical in recommendation systems since they enhance the accuracy and context-harvesting of the suggestions. This greatly improves the user experience since the information and recommendations are more tailored to individual preferences. In addition, knowledge graphs offer a substantial future in AI-driven applications since they help expand capabilities in areas such as natural language interpretation, semantic search, and intricate data analysis. Knowledge graphs offer a consolidated perspective of items and associative connections that help in semantic searches, providing a large sum of information that one may want. Integration of knowledge graphs with AI makes various applications more useful and opens the door to new ways of handling and getting information from large datasets [15]–[17].

Knowledge graphs (KGs) often encounter the issue of insufficient information, resulting in gaps in the data that may significantly hinder the effectiveness and precision of data-driven operations. The incompleteness emerges due to the fact that knowledge graphs (KGs) often gather information from multiple sources, which may not include all potential information or connections. As a result, the representation of real-world entities and their interactions in KGs is sometimes sparse. For example, when a knowledge graph is built using text data, it may not capture relationships that are not clearly stated in the text. This might result in important gaps in the overall structure of the network. The presence of these gaps could affect the functionality of different applications of KGs, such as question-answering and recommendation systems. The success of the results in these applications is highly dependent on the extent to which the underlying knowledge graph is comprehensive. The research emphasizes the need to solve these gaps by creating strong techniques for knowledge graph completion. These techniques attempt to deduce and complete missing relationships or characteristics using the current structure and content of the graph [18]. It is essential to complete a knowledge graph in order to improve its usefulness, dependability, and suitability for different AI-powered applications. Knowledge graph completion (KGC) is the process of accurately predicting and adding missing triples (consisting of entities and their connections) to the graph. This enhances the graph's informative richness and makes it a more complete representation of knowledge. The importance of this work comes from the direct influence that a more comprehensive knowledge graph has on enhancing the efficiency and precision of subsequent tasks such as semantic search, customized recommendations, and intelligent decision-making systems. KGC facilitates the creation of a more associated and semantically robust dataset by filling in the missing relationships. This could significantly boost the reasoning skills of AI models that rely on these graphs for extracting information and making inferences. Moreover, the incorporation of sophisticated methods such as graph neural networks into knowledge graph completion (KGC) procedures has potential in capturing intricate patterns inside graphs, hence enhancing the precision and effectiveness of completion tasks [19].

A. KNOWLEDGE GRAPH COMPLETION TECHNIQUES

Knowledge Graph Completion (KGC) is a critical area of research aimed at inferring missing information in knowledge graphs. These graphs are structured representations of knowledge, consisting of entities (nodes) and relationships (edges). KGC techniques can be broadly categorized into embedding-based approaches and path-based methods, each with unique methodologies and applications.

1) Embedding-Based Approaches

Embedding-based approaches are designed to represent entities and relationships in a continuous vector space, facilitating efficient computation and inference. The primary goal of these methods is to learn low-dimensional representations (embeddings) for entities and relationships that capture their semantic meanings and interactions.

TransE (Translation-based Embedding) TransE posits that relationships can be modeled as translations in the vector space. For a given relationship r between a head entity h and a tail entity t, the model asserts that the vector representation of the head entity plus the vector representation of the relationship should approximate the vector representation of the tail entity. This relationship can be mathematically expressed as:

$$\mathbf{e}_h + \mathbf{r} \approx \mathbf{e}_t \tag{1}$$

where $\mathbf{e}_h$, $\mathbf{r}$, and $\mathbf{e}_t$ are the embeddings of the head entity, relationship, and tail entity, respectively. The model is trained to minimize the distance between the left-hand side and the right-hand side of the equation, typically using a margin-based loss function. Other embedding-based methods, such as TransH, TransR, and RotatE, build upon this foundation by introducing additional complexities to better capture the nuances of relationships in knowledge graphs [20].

RotatE is a knowledge graph embedding model that leverages complex-valued embeddings to represent entities and relationships. The key idea in RotatE is to model relationships as rotations in the complex plane. In RotatE, each entity and relationship is represented by a complex vector:

- **Entity Embeddings**: $\mathbf{e}_h$ (head entity) and $\mathbf{e}_t$ (tail entity).
- **Relation Embedding**: $\mathbf{r}$ (relationship).

The core assumption of RotatE is that the relation $\mathbf{r}$ rotates the head entity $\mathbf{e}_h$ to align with the tail entity $\mathbf{e}_t$ in the complex space. Mathematically, this is expressed as $\mathbf{e}_h \circ \mathbf{r} \approx \mathbf{e}_t$ where $\circ$ denotes the Hadamard product, which is the element-wise multiplication of complex vectors. RotatE captures various relational patterns, including symmetric and antisymmetric relations, by leveraging this rotation-based approach. Despite its strengths, RotatE can face challenges with highly complex relational patterns and may involve more computational complexity due to complex-valued calculations [21].

ComplEx is a knowledge graph embedding model that utilizes complex-valued embeddings to enhance the representation of entities and relationships in knowledge graphs. In ComplEx, both entities and relationships are represented using complex vectors. Specifically, each head entity and tail entity is associated with a complex vector, and relationships are also represented as complex vectors. The core idea behind ComplEx is to apply a scoring function that incorporates these complex-valued embeddings to assess the plausibility of a given relationship between entities. The scoring function for a head entity $\mathbf{h}$, relationship $\mathbf{r}$, and tail entity $\mathbf{t}$ is calculated as the real part of a bilinear form involving these complex embeddings. Mathematically, the scoring function is expressed as:

$$f(\mathbf{h}, \mathbf{r}, \mathbf{t}) = \mathrm{Re}\left(\langle \mathbf{e}_h, \mathbf{r}, \mathbf{e}_t \rangle\right) \tag{2}$$

Here, $\langle \mathbf{e}_h, \mathbf{r}, \mathbf{e}_t \rangle$ denotes the bilinear scoring function in the complex space, and $\mathrm{Re}(\cdot)$ represents the real part of the complex number resulting from this function. The scoring function is computed as:

$$\langle \mathbf{e}_h, \mathbf{r}, \mathbf{e}_t \rangle = \sum_{i=1}^{d} \left(e_{h,i} \cdot \overline{r_i} \cdot e_{t,i} \right) \tag{3}$$

where $e_{h,i}$ and $e_{t,i}$ are the i-th components of the head and tail entity embeddings, respectively, and r_i are the components of the relation embedding. The term $\overline{r_i}$ denotes the complex conjugate of r_i. ComplEx is particularly effective at capturing various types of relational patterns, including symmetric, antisymmetric, and asymmetric relations, due to its use of complex embeddings and the bilinear scoring function. This approach allows ComplEx to model more nuanced interactions between entities and relationships compared to models that use only real-valued embeddings. Despite these advantages, ComplEx may face challenges related to the computational complexity of handling complex-valued embeddings and the accuracy required in the calculations [22].

DistMult is a knowledge graph embedding model designed to represent entities and relationships in knowledge graphs using real-valued embeddings. The primary aim of DistMult is to capture the interactions between entities and relationships through a simplified scoring function. In DistMult, each entity and relationship is represented by a real-valued vector. Specifically, the model assigns a vector to each head entity, tail entity, and relationship. The core idea behind DistMult is to model the relationship between a head entity and a tail entity as a bilinear function, where the relationship is represented by a diagonal matrix [23]. The scoring function in DistMult is defined as follows:

$$f(\mathbf{h}, \mathbf{r}, \mathbf{t}) = \mathbf{e}_h^T \mathbf{R} \mathbf{e}_t \tag{4}$$

where $\mathbf{e}_h$ and $\mathbf{e}_t$ are the real-valued vectors for the head and tail entities, respectively, and $\mathbf{R}$ is a diagonal matrix representing the relationship. In practice, this scoring function is simplified to:

$$f(\mathbf{h}, \mathbf{r}, \mathbf{t}) = \mathbf{e}_h^T \mathrm{diag}(\mathbf{r}) \mathbf{e}_t \tag{5}$$

where $\mathrm{diag}(\mathbf{r})$ denotes a diagonal matrix with the components of the relationship vector $\mathbf{r}$ on the diagonal.DistMult is effective at capturing symmetric relationships due to its bilinear form with a diagonal matrix. However, it is less suited for modeling asymmetric relationships because the diagonal matrix imposes symmetry constraints on the relationship representation. Despite its simplicity and computational efficiency, DistMult may struggle to capture more complex relational patterns that require richer representations of relationships [24].

SimplE is a knowledge graph embedding model designed to capture asymmetric relationships more effectively. Unlike traditional methods, SimplE uses separate representations for the head and tail entities, as well as for the head and tail components of relationships. Each entity e is represented by two vectors: $\mathbf{e}_h$ for the head entity and $\mathbf{e}_t$ for the tail entity. Similarly, each relationship r is represented by two vectors: $\mathbf{r}_h$ and $\mathbf{r}_t$. The scoring function for SimplE is defined as:

$$f(\mathbf{h}, \mathbf{r}, \mathbf{t}) = (\mathbf{e}_h \cdot \mathbf{r}_h) \cdot (\mathbf{e}_t \cdot \mathbf{r}_t)$$

where $\mathbf{e}_h$ and $\mathbf{e}_t$ are the head and tail entity vectors, respectively, and $\mathbf{r}_h$ and $\mathbf{r}_t$ are the head and tail components of the relationship vector, respectively. This approach enables SimplE to better model asymmetric relationships by differentiating how the relationship affects the head and tail entities, leading to improved performance in tasks requiring such asymmetry [25].

QuatE extends the concept of complex-valued embeddings to quaternions, providing a richer representation of entities and relationships. Each entity and relationship is represented by a quaternion q, which can be expressed as:

$$q = w + xi + yj + zk$$

where w, x, y, and z are real-valued components, and i, j, and k are the fundamental quaternion units. QuatE models relationships as rotations in the quaternion space. The scoring function for QuatE is:

$$f(\mathbf{h}, \mathbf{r}, \mathbf{t}) = \mathrm{Re}(\mathbf{h} * \mathbf{r} * \mathbf{t}^{-1})$$

where $*$ denotes quaternion multiplication, and Re extracts the real part of the resulting quaternion. This formulation allows QuatE to capture complex relational patterns by leveraging the additional dimensions of quaternions, making it effective for modeling intricate interactions between entities in knowledge graphs [26].

However, these methods are unable to adequately represent the Complex, multi-relational structures found in Knowledge Graphs (KGs). Because TransE and similar models assume that relationships may be represented as simple translations in vector space, they are not well suited to express complex patterns, such as symmetric or many-to-many relationships. Similarly, more sophisticated models, such as QuatE and RotatE, are limited in their capacity to completely reflect more complicated relationships because they mainly concentrate on pairwise interactions [21]. Additionally, the significance of higher-order structures and node centrality, two structural aspects of the graph that are critical to comprehending the overall interactions and significance of entities within the KG, are not sufficiently taken advantage of by these methods.

2) Path-Based Methods

Path-based methods leverage the structural properties of knowledge graphs to infer missing information. These techniques focus on the paths connecting entities within the graph, utilizing the relationships along these paths to derive new knowledge. The fundamental idea is that certain paths can indicate potential relationships or missing entities [27].

One notable path-based method is *PathRanking*, which ranks paths based on their relevance to a given query. The model evaluates paths between entities and assigns scores based on various features, such as the types of relationships involved and the lengths of the paths [28]. The scoring function can be expressed as:

$$\text{Score}(h, t, p) = \sum_{i=1}^{n} \alpha_i \cdot f_i(h, t, p) \tag{6}$$

where α_i are weights assigned to different features $f_i(h, t, p)$ derived from the path p connecting the head entity h and tail entity t. This approach allows the model to capture complex relationships that may not be directly observable from the graph's structure alone.

Another significant method is *DeepPath*, which employs deep learning techniques to learn representations of paths. In this approach, the model takes a feature vector representing the path as input and processes it through a neural network. The output can be expressed as:

$$z = \sigma(W \cdot x + b) \tag{7}$$

where x is the input feature vector, W is the weight matrix, b is the bias term, and σ is an activation function. This method allows for the automatic learning of complex patterns in the data, enhancing the model's ability to infer missing relationships based on the paths present in the knowledge graph [29], [30].

3) Integration of Relational Paths and Structural Features

Recent advancements in knowledge graph (KG) completion have focused on integrating relational paths with other features to enhance the accuracy and comprehensiveness of predictions [31], [32]. One notable approach is the use of Relational Graph Convolutional Networks (R-GCNs) and Multi-Relational GCNs [33].

Relational Graph Convolutional Networks (R-GCNs) R-GCNs extend traditional Graph Convolutional Networks (GCNs) to handle multiple types of relations in knowledge graphs. The core idea is to apply different convolutional filters for each relation type, allowing the model to capture complex relational patterns. By leveraging relational paths, R-GCNs can aggregate information from various types of relationships, providing a richer representation of entity interactions. This approach effectively addresses the challenge of multi-relational data by learning context-specific features for each relation type, improving the accuracy of KG completion tasks [34].

Multi-Relational GCNs further build on the concept of relational graph convolutions by incorporating additional relational features and aggregating information from various paths. These models utilize multiple layers of graph convolutions to capture hierarchical and compositional relations between entities. The integration of relational paths and structural features allows Multi-Relational GCNs to learn more nuanced representations of entities and their interactions, leading to better performance in predicting missing links and completing knowledge graphs [35].

While embedding-based methods like TransE, TransH, and RotatE have proven effective in learning low-dimensional representations of entities and relationships, like by treating relationships as geometric transformations in a vector space they frequently oversimplify the complex structure of knowledge graphs. Because of this, these models have trouble accurately capturing complex patterns like symmetric connections or many-to-many linkages [7], [10]. Comparably, path-based techniques like Path Ranking, although good at exploiting the graph's structural features, frequently overlook node-specific characteristics like centrality and node degree, which can offer important contextual data about an entity's importance [27], [28].

Our proposed method addresses these limitations by integrating relational paths and node degree information. Although path-based techniques concentrate on extracting information from entity relationships, we improve on this strategy by adding a tail node degree, which expresses the importance of entities in the graph. With the help of this dual approach, our model can predict relationships by differentiating between entities that have similar relational paths but distinct structural information. Our method offers a more comprehensive understanding of entity interactions in knowledge graphs by combining structural aspects like tail node degree with the advantages of embedding and path-based methods.

III. METHODOLOGY

This study introduces the DualRep approach, which predicts the relationship $(h, ?, t)$ in a KG. Before presenting the model, a concise explanation of the problem setting is presented.

Formulation of the Problem: A knowledge graph (KG) is an arrangement of textual components known as triples, denoted as $G = (h, r, t)$, where E and R signify the sets of entities and relationships in the graph G, respectively. In this context, $h \in E \rightarrow$ denotes the head entity, $t \in E \rightarrow$ signifies the tail entity, and $r \in R \rightarrow$ indicates their relationship. Predicting the entity $(h \ or \ t)$ is more complex in Knowledge Graph Completion (KGC) than predicting relationships (links). Entity prediction entails the problem of predicting a missing head h or tail t (denoted by ?) in an incomplete triple, $(? \ , \ r \ , \ t)$ where the head h entity is to be predicted, and $(h \ , \ r \ , \ ?)$ where the tail t entity needs to be predicted. This study addressed the problem of relationship prediction $(h, r, ?)$ in the knowledge graph $\mathcal{G}$.

Traditional path-based techniques for knowledge graph completion have mostly concentrated on the sequential relationships among entities to determine missing relationships. While effective in capturing the relational context, these methods often overlook the structural significance of the entities themselves, particularly the tail nodes in a given relational path. To address this gap, we propose an enhanced methodology that integrates both relational paths and the structural feature of tail node degree. This approach aims to enrich the prediction of relationships by incorporating the connectivity patterns within the graph and the centrality of the tail nodes. By combining these two aspects, our methodology provides a more efficient and accurate framework for predicting missing relationships in knowledge graphs. The proposed approach, which integrates path and structural features, outperforms the state-of-the-art model.

Define Tail Node Degree: To enhance the prediction of relationships in a knowledge graph (KG), the degree of each tail node is first calculated. The degree of a node, denoted as $\deg(t)$, represents the total number of edges connected to the node t. This metric provides insight into the node's structural importance or centrality within the graph. Nodes with higher degrees are considered more central, while those with lower degrees are deemed more peripheral.

Path Embedding: Path embedding for relational paths connecting head nodes to tail nodes are generated next. Each path P from a head node h to a tail node t is represented by an embedding $s(P)$. This embedding captures the sequential relational context of the path, reflecting the nature of the connections between entities in the graph.

Feature Extraction: The tail node degree is converted into a feature that can be integrated with the path embedding. A feature vector $f_{\text{deg}}(t)$, derived from the degree $\deg(t)$, is defined. This feature vector $f_{\text{deg}}(t)$ may be normalized or embedded to match the dimensionality of the path embedding $s(P)$.

Concatenation Path and Degree Features: The path embedding and the tail node degree feature are combined to create a unified representation. A combination function, specifically concatenation, is used to merge $s(P)$ and $f_{\text{deg}}(t)$. The resulting composite vector $s_{P,t}$ enhances the path's representation, integrating both relational and structural information. Below are consistent step-by-step equations, and clear notation is given to ensure all elements of the methodology are well-defined.

$$\deg : E \rightarrow \mathbb{N}, \quad \deg(t) = \text{associated edge to } t. \tag{8}$$

Between head and tail, each path P reaching h to t is represented by the embedding:

$$s(P) \in \mathbb{R}^d, \tag{9}$$

where $s(P)$ captures the sequential relational context along the path. In the DualRep component, the degree feature of a tail node is converted into a vector representation:

$$f_{\text{deg}} : E \rightarrow \mathbb{R}^m, \quad f_{\text{deg}}(t) = \text{vector format of } \deg(t). \tag{10}$$

The path embedding and degree feature vector are merged using a concatenation operation as follows:

$$\text{Concat} : \mathbb{R}^d \times \mathbb{R}^m \rightarrow \mathbb{R}^{d+m}, \quad s_{P,t} = \text{Concat}(s(P), f_{\text{deg}}(t)). \tag{11}$$

An attention mechanism is applied to determine the significance of each path, considering both the relational path and the tail node degree. The attention weights $\alpha_{P,t}$ are computed for different paths from h to t. Each path P based on its importance has its attention weights $\alpha_{P,t}$ computed as follows:

$$\alpha_{P,t} = \frac{\exp(s_{P,t}^\top \cdot s(h,t))}{\sum_{(P',t') \in P_{h\rightarrow t}} \exp(s_{P',t'}^\top \cdot s(h,t))}, \tag{12}$$

where $s(h, t)$ represents the combined relational context between h and t, and $\alpha_{P,t}$ quantifies the importance of each path based on both the relational path and the tail node degree. The final term of relationship representation, $s^*_{h\rightarrow t}$, aggregates the weighted path representations:

$$s^*_{h\rightarrow t} = \sum_{(P,t) \in P_{h\rightarrow t}} \alpha_{P,t} \cdot s_{P,t}. \tag{13}$$

Here, the term Concat refers to a mathematical operation denoted by the symbol $\oplus$. The Combine operation encompasses functions such as concatenation, weighted sum, or any other function specifically intended to merge the path and degree characteristics into a single, unified representation. For this purpose, concatenation is used. The final representation $s^*_{h\rightarrow t}$ is computed as follows:

$$s^*_{h\rightarrow t} = \sum_{(P,t) \in P_{h\rightarrow t}} \alpha_{P,t} \cdot s_{P,t} \tag{14}$$

This final representation incorporates both path-based and degree-based information, providing a comprehensive view of the relationship between h and t. Relation Prediction: The final representation $s^*_{h\rightarrow t}$ is then utilized to predict missing

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3501735

relationships in the knowledge graph. This enhanced representation improves the accuracy of predictions by integrating detailed relational path information with the structural significance of the tail node. The DualRep method is detailed in Algorithm 1 and the methodology diagram 1.

Algorithm 1 Enhanced Relation Prediction in Knowledge Graphs

Input: Knowledge Graph KG with nodes (entities) and edges (relationships), head node h
Output: Predicted relationships for node h with potential tail nodes t
Initialize KG with nodes and edges
for each potential tail node t in KG **do**
    Calculate deg(t) – the degree of tail node t
    Calculate the immediate neighborhood size for t
end for
for each relational path P from h to t **do**
    Embed(P) – Compute the embedding of path P
    Feature(deg(t)) – Normalize or embed the degree of t
    $s_{P,t} = \text{Combine}(\text{Embed}(P), \text{Feature}(\text{deg}(t)))$
        ▷ Combine path and degree information, typically by concatenation
end for
Apply Attention Mechanism:
for each path representation $s_{P,t}$ **do**
    Compute attention weights $\alpha_{P,t}$ based on the relevance of path and node degree
    $\alpha_{P,t} = \frac{\exp(s_{P,t} \cdot s(h,t))}{\sum(\exp(s_{P',t'} \cdot s(h,t)) \text{ for all } (P',t') \text{ in paths from } h \text{ to } t)}$
end for
Aggregate Representations:
$s^*_{h \to t} = \sum(\alpha_{P,t} \cdot s_{P,t})$ for all (P,t) in paths from h to t
    ▷ This is the final representation of the relationship between h and potential t
Predict Relationships:
for each t **do**
    Use $s^*_{h \to t}$ to predict the probability or type of relationship (r) between h and t
end for

A. EXPERIMENTAL SETUP

In this section, the experimental procedure is explained step by step.

Datasets: The DualRep model is evaluated on various widely used KG datasets. Displaying parameters such as the number of entities, relationships, and data divided into training, validation, and test triplets, the table summarizes the different knowledge graph datasets used in machine learning for tasks such as relation prediction or entity resolution. FB15K and its version, FB15K-237, are both datasets generated from Freebase. These datasets include thousands of entities and connections but vary in complexity and the amount of data they contain. The WordNet-based lexical datasets WN18 and WN18RR provide lexical datasets with fewer links and a more straightforward structure. Varied scales and densities are shown by NELL995, which is part of the NELL system. These scales and densities are represented by the diverse metrics of volume and complexity that they include. These metrics are vital for benchmarking and building AI models. The datasets are freely available online [1]. The statistical detail of all the datasets is given in Table 1.

Hyperparameter: The studies used a batch size of 16 and a learning rate of 5e-5, using the Adam optimizer with cross-entropy as the loss function. The studies were conducted on an NVIDIA GeForce RTX 3090 GPU, which included 24 GB of RAM. The epoch number was set at 20 for DualRep and the other models.

Evalaution: The evaluation of KGEM performance primarily relies on three rank-based metrics: Hits@k, Mean Rank (MR), and Mean Reciprocal Rank (MRR) [36]. The use of these metrics arises from the need to create negative triples to train and evaluate a KGEM. Therefore, positive triples are compared against negative ones to assess the model's ability to forecast credible facts accurately. To be more precise, when provided with a ground-truth triple $(, r, t)$, we construct all potential triples $(, ?, t)$, $(?, r, t)$ and $(h, r, ?)$ by considering all the entities seen in the knowledge graph (KG). Subsequently, the KGEM evaluates these triples and compares their scores with those assigned to the ground-truth triple. The mathematical equation for the evaluation is given in Equations 15 to 17.

1) Hit@k

The Hits@K metric, as defined by Equation 5, measures the percentage of ground-truth triples that are included in the top K highest-scoring triples.

$$\text{Hits@}K = \frac{1}{|\mathcal{B}|} \sum_{q \in \mathcal{B}} 1[rank(q) \leq K] \quad (1) \tag{15}$$

2) The Mean Rank (MR)

The Mean Rank (MR) Equation 6 is calculated as the arithmetic mean of the ranks of the ground-truth triples.

$$\text{MR} = \frac{1}{|\mathcal{B}|} \sum_{q \in \mathcal{B}} \text{rank}(q) \quad (2) \tag{16}$$

3) Mean Reciprocal Rank (MRR)

The Mean Reciprocal Rank (MRR) Equation 7, is calculated as the average of the ranks' reciprocals and ground-truth triples.

$$\text{MRR} = \frac{1}{|\mathcal{B}|} \sum_{q \in \mathcal{B}} \frac{1}{\text{rank}(q)} \quad (3) \tag{17}$$

[1] https://github.com/LIANGKE23/Awesome-Knowledge-Graph-Reasoning

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3501735

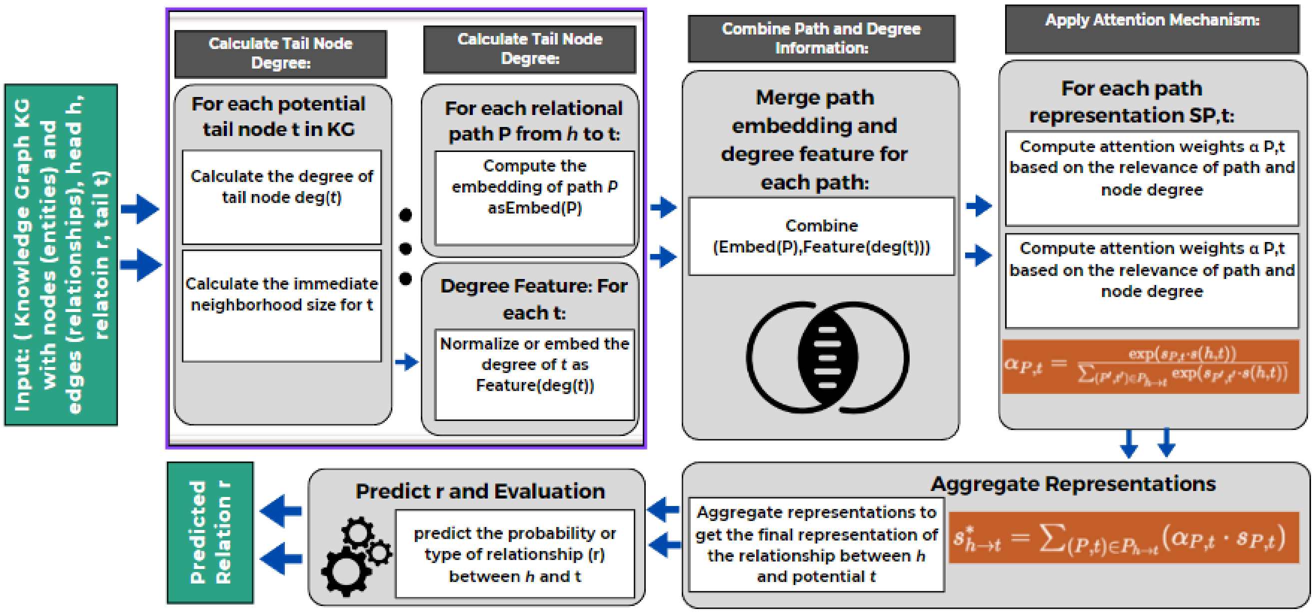

FIGURE 1: DualRep relation prediction methodology diagram

TABLE 1: Datasets Statistics

Data / Properties	Entities	Relationships	Train Triplets	Validation Triplets	Test Triplets	[V]	[M]
FB15K	14951	1345	483142	5000	59071	32441.7	64.55
FB15K-237	14540	237	272114	17535	20466	12336.3	37.42
WN18	40943	18	141441	5000	5000	234.4	6.87
WN18RR	40943	11	86835	3034	3134	64.26	4.21
NELL995	63917	198	137465	5000	5000	750.61	4.28

TABLE 2: Performance metrics of different models on various datasets.

Dataset	Metric	TransE	ComplEx	DistMult	RotatE	SimplE	QuatE	DualRep
FB15k	**MRR**	88.69	85.81	65.96	79.68	91.09	90.87	**92.32**
	Hit@1	86.56	80.82	40.88	75.62	82.36	87.99	**90.51**
	Hit@3	88.06	81.93	45.47	76.99	85.78	89.36	**93.64**
FB15k-237	**MRR**	85.39	80.21	83.33	76.37	80.68	90.24	**90.36**
	Hit@1	84.52	79.72	78.72	72.07	78.32	86.22	**87.51**
	Hit@3	86.16	81.32	79.41	74.88	80.39	80.52	**89.64**
WN18	**MRR**	90.45	89.56	74.02	86.78	**90.78**	88.41	88.86
	Hit@1	83.75	85.25	57.54	81.42	84.16	83.63	**85.46**
	Hit@3	86.71	87.63	61.56	84.74	87.24	85.65	**87.73**
WN18RR	**MRR**	76.84	84.86	83.97	77.67	69.87	80.75	**91.96**
	Hit@1	71.61	82.03	68.32	74.67	67.63	82.88	**90.65**
	Hit@3	73.41	83.98	71.64	77.99	70.77	84.37	**92.85**
NELL995	**MRR**	90.56	88.34	88.36	85.68	78.87	79.87	**91.12**
	Hit@1	87.67	85.68	81.79	81.75	75.87	76.63	**88.35**
	Hit@3	89.74	87.74	83.45	85.63	78.35	78.69	**92.65**

IV. RESULTS AND DISCUSSIONS

In this section, we have explained the outcomes of various approaches applied to multiple knowledge graph datasets. We have collected five distinct massive datasets, each of which is very important for the study and comparison of knowledge graphs.

The performance measures for the FB15k dataset show that DualRep consistently outperforms other models, achieving the greatest scores in terms of MRR (92.32), Hit@1 (90.51), and Hit@3 (93.64), for more detail see Table 2.

QuatE and SimplE exhibit impressive performance, particularly in terms of Mean Reciprocal Rank (MRR), achieving scores of 90.87 and 91.09 respectively. The models consistently demonstrate a pattern in which DistMult consistently performs worse than the others, especially in terms of Hit@1 (40.88) and Hit@3 (45.47). This indicates that DistMult is less effective in collecting the most accurate responds that are rated highest compared to the other models. TransE, ComplEx, and RotatE provide satisfactory to commendable performance, although none outperform DualRep in all measures. Within the FB15k-237 dataset, the DualRep and QuatE models demonstrate superior performance, achieving roughly comparable Mean Reciprocal Rank (MRR) scores of 90.36 and 90.24, respectively; for more detail, see Table 3. This indicates their usefulness within the context of this knowledge graph. TransE has a commendable level of performance, notably in the Hit@1 metric (84.52), which suggests its ability to properly forecast the best outcome. DistMult and RotatE demonstrate much worse performance, particularly in terms of MRR and Hit@1 measures, indicating potential difficulties in accurately predicting entities in this dataset. SimplE demonstrates average performance across all parameters without excelling in any one area. SimplE achieves outstanding performance on the WN18 dataset, with the greatest Mean Reciprocal Rank (MRR) of 90.78 and competitive Hit rates, for more detail see Table 4. This underscores its remarkable capabilities in the graph context. TransE has excellent performance, with ComplEx closely trailing behind. ComplEx achieves the highest Hit@1 score of 85.25, indicating its superior accuracy in making top-ranked predictions. DistMult exhibits significant performance deficiencies across all measures, especially in Hit@1 (57.54), suggesting a potential constraint in accurately prioritizing the right responses in this dataset. DualRep, while not leading in any certain category, regularly performs well in all areas. DualRep outperforms all other models in the WN18RR dataset, achieving the top scores in all measures. It significantly surpasses the competition in terms of Mean Reciprocal Rank (MRR) with a score of 91.96 and Hit rates, showcasing its strong performance in complicated relational reasoning; for more detail, see Table 5. QuatE demonstrates exceptional performance, notably in the accuracy of its top results, as seen by its Hit@1 score of 82.88 and Hit@3 score of 84.37. ComplEx and DistMult have comparable Mean Reciprocal Rank (MRR) outcomes, but, their Hit rates do not align with their MRR performance, indicating a discrepancy in ranking consistency. TransE and RotatE exhibit satisfactory performance, whereas SimplE lags behind, particularly in terms of Mean Reciprocal Rank (MRR) and Hit@1. DualRep outperforms other models in NELL995, achieving high scores in MRR (91.12), Hit@1 (88.35), and Hit@3 (92.65); for more detail, see Table 6. These results show its overall effectiveness in entity resolution across many measures. TransE demonstrates outstanding performance with a Mean Reciprocal Rank (MRR) of 90.56, along with high Hit@1 and Hit@3 scores, indicating its excellent retrieval

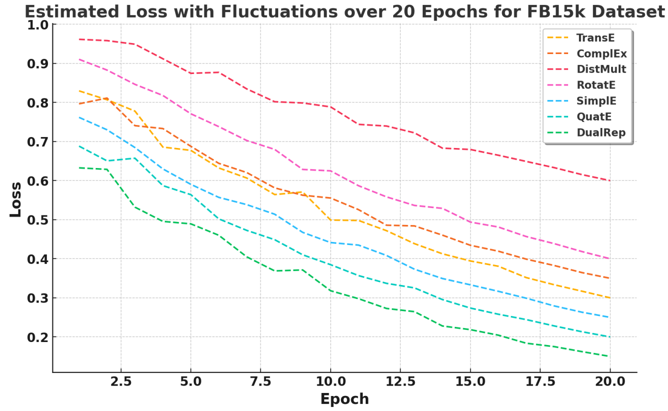

FIGURE 2: Loss over FB15k

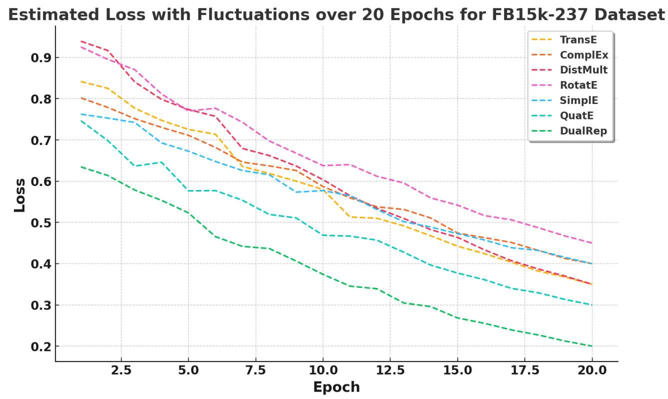

FIGURE 3: Loss over FB15k-237

capabilities. ComplEx and DistMult provide favorable Mean Reciprocal Rank (MRR) and Hit rates, indicating their suitability for this dataset. SimplE and QuatE, on the other hand, demonstrate worse performance in several measures, particularly in the accuracy of their highest-ranked predictions. This indicates the need for possible improvements in these models for this particular scenario. DualRep regularly demonstrates superior performance on many datasets, highlighting its resilience and capacity to handle a wide range of knowledge graphs. The success of models such as SimplE, TransE, and ComplEx varies based on each dataset's particular qualities and problems. DistMult often exhibits subpar performance, particularly in datasets with intricate relationship patterns, emphasizing the need for improvements or alterations in its methodology for such contexts. The loss of DualRep and other state-of-the-art over five datasets reported in Figures 2-6, shows the process of model validation over 20 epochs. DualRep consistently outperforms others, with the lowest final loss values, presenting its superior ability from the relational path and node degree features.

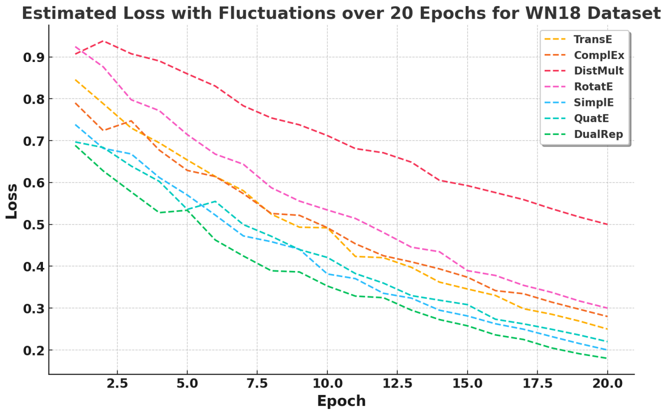

FIGURE 4: Loss over WN18

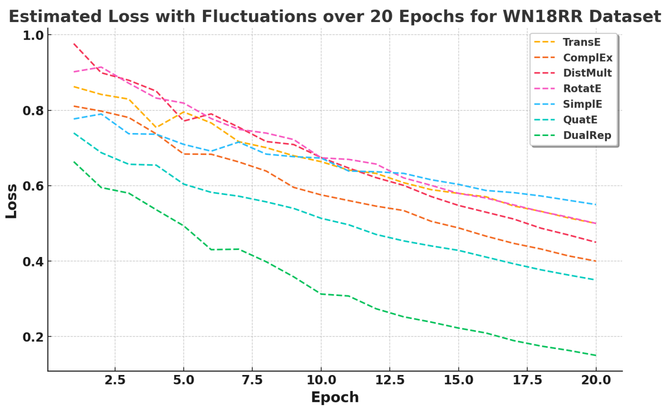

FIGURE 5: Loss over WN18RR

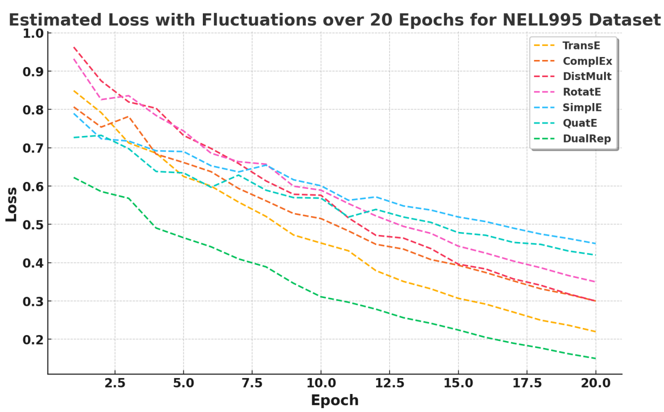

FIGURE 6: Loss over NELL995

A. MODEL EXPLAINABILITY USING DUALREP: CASE STUDY

We have utilized the three datasets to show the explainability of the proposed model, DualRep. DualRep concatenates KG's path and tail node degree to predict the missing relations. A case study assessed two relations from each KG dataset, incorporating the critical path and tail nodes' degree insights contributing to the model's prediction. The dualRep model was assessed across three knowledge graph datasets: WN18, WN18RR, and FB15k. It uses path and tail node degree information to predict the missing relationships in KG. In WN18 KG, hypernym, meronym, and antonym relationship are predicted sing path and node degree. Over WN18RR, synonym, hypernym, and hyponym relationships have been predicted using dual path and tail node density. In the FB15k dataset, the relation "actor in the movie" and "organization founded" has been predicted, given in Table 3. Overall, it demonstrates the ability of the DualRep model to predict the relationship using both path and density information of the tail node, which provides a transparent and interpretable framework for knowledge graph completion.

B. ABLATION STUDY

The DualRep model has been evaluated in an ablation study to determine the significance of its two main components: path and tail node degree features. The ablation study revealed the purpose and significance of each element in the DualRep model, contrasting outcomes across numerous datasets. Two variants of the DualRep model were created, eliminating the path and discarding the tail node degree. The evaluation revealed a significant decrease in performance when either component was omitted, highlighting the advantageous correlation between path and tail node degree attributes. The absence of a path resulted in reduced model performance across all datasets. Conversely, eliminating the tail node degree also impacts the effectiveness of DualRep. Conversely, these studies indicate that both elements of DualRep are crucial for optimal performance. The experimental findings are shown in Table 4.

V. CONCLUSION

Our DualRep model has been shown to be very effective in completing Knowledge Graph (KG) tasks, as evidenced by its excellent performance across several benchmark datasets. DualRep, which combines relational pathways with node density characteristics, has consistently shown superior performance in measures such as MRR, Hit@1, and Hit@3. DualRep demonstrated a mean reciprocal rank (MRR) of 90.80, with a hit rate at rank 1 (Hit@1) of 87.39 and a hit rate at rank 3 (Hit@3) of 91.18. These results highlight the strong capacity of DualRep to reliably anticipate connections and efficiently rank them. The results confirm our prediction that combining sequential relational context with structural prominence improves the accuracy and comprehension of entity connections in knowledge graphs. Our approach improves the semantic understanding of KGs by overcoming the

TABLE 3: Explainability of DualRep Model on Four Datasets

Dataset	Predicted Relation	Important Paths	Tail Node Degree Insight
WN18	hypernym	(is-a, similar to), (related to, type of)	High-degree hypernyms connect to many related terms
	meronym	(has part, consists of), (component of, comprises)	High-degree parts are essential components across many entities
	antonym	(related to, contrast with), (opposes, in contrast with)	High-degree terms have broad usage and many opposites
WN18RR	synonym	(related to, synonymous with), (equivalent to, alike in meaning)	High-degree synonyms suggest frequently used terms
	hypernym	(is-a, belongs to), (type of, generalization of)	Hypernyms with high degrees are widely connected in taxonomies
	hyponym	(more specific than, sub-class of), (narrower term, belongs to category)	Hyponyms with high degrees are likely specialized terms with many connections
FB15k	actor in movie	(acted in, directed by), (starred in, produced by)	Actors with high degrees are often central in the film industry
	person born in place	(born in, lives in), (hometown of, citizenship)	Cities with many residents born there suggest central locations
	organization founded	(founded by, headquartered in), (started in, leadership of)	Organizations with high degrees are influential hubs in the graph

TABLE 4: DualRep: Impact of Removing Path Embeddings and Tail Node Degree Features

Dataset	Metric	DualRep	Without Path	Without Degree
FB15k	MRR	92.27	91.10	90.00
	Hit@1	90.48	88.39	88.01
	Hit@3	93.56	92.30	91.00
FB15k-237	MRR	90.41	89.16	88.50
	Hit@1	87.419	85.90	86.00
	Hit@3	89.64	88.40	88.00
WN18	MRR	88.86	87.50	86.78
	Hit@1	85.46	84.30	84.01
	Hit@3	87.69	86.78	86.47
WN18RR	MRR	91.89	90.60	89.81
	Hit@1	90.77	89.31	88.48
	Hit@3	92.87	91.51	90.76
NELL995	MRR	91.12	90.20	89.50
	Hit@1	88.35	87.50	87.00
	Hit@3	92.65	91.50	90.80

limitations of conventional embedding-based methods. This enhancement allows for more precise and reliable predictions in applications such as recommendation systems, semantic search, and intelligent data mining. The constant performance of DualRep across several datasets demonstrates its capacity to adapt and effectively use multiple views in KG completion. This establishes a new standard in the area and provides significant enhancements compared to previous models.

REFERENCES

[1] A. Singhal et al., "Introducing the knowledge graph: things, not strings," Official google blog, vol. 5, no. 16, p. 3, 2012.
[2] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations," in Proceedings of the 10th ACM conference on recommender systems, pp. 191–198, 2016.
[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.
[4] M. Nickel, K. Murphy, V. Tresp, and E. Gabrilovich, "A review of relational machine learning for knowledge graphs," Proceedings of the IEEE, vol. 104, no. 1, pp. 11–33, 2015.
[5] H. Gul, A. Amin, A. Adnan, and K. Huang, "A systematic analysis of link prediction in complex network," IEEE Access, vol. 9, pp. 20531–20541, 2021.
[6] Q. Wang, Z. Mao, B. Wang, and L. Guo, "Knowledge graph embedding: A survey of approaches and applications," IEEE transactions on knowledge and data engineering, vol. 29, no. 12, pp. 2724–2743, 2017.
[7] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," Advances in neural information processing systems, vol. 26, 2013.
[8] B. Yang, W.-t. Yih, X. He, J. Gao, and L. Deng, "Embedding entities and relations for learning and inference in knowledge bases," arXiv preprint arXiv:1412.6575, 2014.
[9] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," arXiv preprint arXiv:1609.02907, 2016.
[10] Z. Sun, Z.-H. Deng, J.-Y. Nie, and J. Tang, "Rotate: Knowledge graph embedding by relational rotation in complex space," arXiv preprint arXiv:1902.10197, 2019.
[11] S. Zhang, Y. Tay, L. Yao, and Q. Liu, "Quaternion knowledge graph embeddings," Advances in neural information processing systems, vol. 32, 2019.
[12] A. G. Vrahatis, K. Lazaros, and S. Kotsiantis, "Graph attention networks: A comprehensive review of methods and applications," Future Internet, vol. 16, no. 9, p. 318, 2024.
[13] D. Li, H. Qu, and J. Wang, "A survey on knowledge graph-based recommender systems," in 2023 China Automation Congress (CAC), pp. 2925–2930, IEEE, 2023.
[14] J. Chicaiza and P. Valdiviezo-Diaz, "A comprehensive survey of knowledge graph-based recommender systems: Technologies, development, and contributions," Information, vol. 12, no. 6, p. 232, 2021.
[15] F. Lecue, "On the role of knowledge graphs in explainable ai," Semantic Web, vol. 11, no. 1, pp. 41–51, 2020.
[16] D. Dessì, F. Osborne, D. Reforgiato Recupero, D. Buscaldi, E. Motta, and H. Sack, "Ai-kg: an automatically generated knowledge graph of artificial intelligence," in The Semantic Web–ISWC 2020: 19th International Semantic Web Conference, Athens, Greece, November 2–6, 2020, Proceedings, Part II 19, pp. 127–143, Springer, 2020.
[17] E. Rajabi and K. Etminani, "Knowledge-graph-based explainable ai: A systematic review," Journal of Information Science, p. 01655515221112844, 2022.
[18] M. Zamini, H. Reza, and M. Rabiei, "A review of knowledge graph completion," Information, vol. 13, no. 8, p. 396, 2022.
[19] S. Hong, H. Qian, Y. Gao, and H. Lyu, "Knowledge graph completion based on graph attention networks and text information," in 2022 IEEE 5th International Conference on Computer and Communication Engineering Technology (CCET), pp. 182–186, IEEE, 2022.
[20] H. Zhao, "Documenting knowledge graph embedding and link prediction using knowledge graphs," Master's thesis, Hannover: Gottfried Wilhelm Leibniz Universität, 2024.
[21] X. Ge, Y. C. Wang, B. Wang, C.-C. J. Kuo, et al., "Knowledge graph embedding: An overview," APSIPA Transactions on Signal and Information Processing, vol. 13, no. 1, 2024.
[22] J. Cao, J. Fang, Z. Meng, and S. Liang, "Knowledge graph embedding: A survey from the perspective of representation spaces," ACM Computing Surveys, vol. 56, no. 6, pp. 1–42, 2024.
[23] X. Feng, H. Kamigaito, K. Hayashi, and T. Watanabe, "Model-based subsampling for knowledge graph completion," arXiv preprint arXiv:2309.09296, 2023.
[24] J. Shi, M. Li, and J. Zhao, "Knowledge graph link prediction model based on attention graph convolutional network," in International Conference on Advanced Data Mining and Applications, pp. 448–453, Springer, 2023.
[25] Y. Xiao, S. Zheng, J. Shi, X. Du, and J. Hong, "Knowledge graph-based manufacturing process planning: A state-of-the-art review," Journal of Manufacturing Systems, vol. 70, pp. 417–435, 2023.
[26] K. Yang, R. Yu, B. Guo, and S. Zhen, "Is multi-level data enhancement helpful for knowledge graph? a new perspective on multimodal fusion," Knowledge-Based Systems, p. 112285, 2024.
[27] Y. Lan, S. He, K. Liu, X. Zeng, S. Liu, and J. Zhao, "Path-based knowledge reasoning with textual semantic information for medical knowledge graph completion," BMC medical informatics and decision making, vol. 21, pp. 1–12, 2021.
[28] S. Seo, B. Oh, and K.-H. Lee, "Reliable knowledge graph path representation learning," IEEE Access, vol. 8, pp. 32816–32825, 2020.
[29] D. Shi, T. Wang, H. Xing, and H. Xu, "A learning path recommendation model based on a multidimensional knowledge graph framework for e-learning," Knowledge-Based Systems, vol. 195, p. 105618, 2020.

[30] S. Zhu, "Exploring knowledge graph-based neural-symbolic system from application perspective," arXiv preprint arXiv:2405.03524, 2024.
[31] H. Dong, P. Wang, M. Xiao, Z. Ning, P. Wang, and Y. Zhou, "Temporal inductive path neural network for temporal knowledge graph reasoning," Artificial Intelligence, vol. 329, p. 104085, 2024.
[32] Z. Bi, S. Cheng, J. Chen, X. Liang, F. Xiong, and N. Zhang, "Relphormer: Relational graph transformer for knowledge graph representations," Neurocomputing, vol. 566, p. 127044, 2024.
[33] J. Zhang, B. Hui, C. Mu, and L. Tian, "Learning multi-graph structure for temporal knowledge graph reasoning," Expert Systems with Applications, vol. 255, p. 124561, 2024.
[34] L. Yang, J. Chen, Z. Wang, and F. Shang, "Relation mapping based on higher-order graph convolutional network for entity alignment," Engineering Applications of Artificial Intelligence, vol. 133, p. 108009, 2024.
[35] Y. Fang, X. Li, R. Ye, X. Tan, P. Zhao, and M. Wang, "Relation-aware graph convolutional networks for multi-relational network alignment," ACM Transactions on Intelligent Systems and Technology, vol. 14, no. 2, pp. 1–23, 2023.
[36] C. T. Hoyt, M. Berrendorf, M. Galkin, V. Tresp, and B. M. Gyori, "A unified framework for rank-based evaluation metrics for link prediction in knowledge graphs," arXiv preprint arXiv:2203.07544, 2022.

HAJI GUL is currently pursuing a Ph.D. in Artificial Intelligence at the University of Brunei Darussalam, with a research focus on Knowledge Graphs. He received his B.Sc. degree in Computer Science, Mathematics, and Physics from the University of Peshawar, Pakistan, and a Bachelor of Education from Allama Iqbal Open University, Islamabad, Pakistan. He further obtained his M.S. and M.Sc. degrees in Computer Science from the Institute of Management Sciences, Hayatabad Peshawar, Pakistan. His M.S.C.S. thesis was titled "Link Prediction Using Node Information on Local Path," and his M.Sc. project involved developing an Internet download application. Additionally, he completed a Diploma in Information Technology from 2014 to 2015, which included courses in Information Technology and Environmental Awareness and Education from 2007 to 2008. Haji Gul previously served as a Lecturer at the Centre for Excellence in Information Technology, City University of Science and Information Technology, Peshawar, Pakistan. His expertise includes graph analysis, graph clustering, link prediction, data mining, and data analysis.

FERAS AL-OBEIDAT Feras Al-Obeidat received the master's and Ph.D. degrees in computer science from the University of New Brunswick, Canada. He is currently an Assistant Professor with the College of Technological Innovation, Zayed University. His primary fields of research are data mining and machine learning. After completing his Ph.D., he has contributed to industrial, university, and government teaching, and research with premier organizations, including IBM Canada.

ADNAN AMIN received the M.Sc. degree in computer science from the University of Peshawar, Pakistan, in 2008, and the M.S. degree (Hons.) in computer science from IMSciences, Peshawar, Pakistan, in 2015, where he is currently pursuing the Ph.D. degree in data mining/machine learning with the Centre of Excellence in Information Technology. Previously, he was working for Maxwell Stamp PLC, London, as an International Consultant for curricular and academic development activities for the School of ICT, National Institute of Management and Administration, University of Jyvaskyla, Finland, under the World Bank funded project (P102573). He is currently a Lecturer with the Centre of Excellence in Information Technology, Institute of Management Sciences, (IMSciences). He is also a leading Expert in data mining, machine learning, and data science. He is supervising many M.S. research students in the area of data mining and machine learning. He has authored more than 22 research articles, including more than 12 journal articles. His research interests include cross-disciplinary and mostly applied industry-oriented including Churn prediction, Prudent-based expert systems, customer analytics and target marketing, active and adaptive learning, real-time classification and segmentation, cyber security, rough set theory, ripple down learning, oversampling, cognitive skills, and big data analytics. He has been the Track Chair of WorldCist'21 workshop (LCBUDAMLT) and a Program Committee Member in numerous conferences, such as WorldCist-15, WorldCist-16, and SDIWC and also active reviewers for more than 23 reputed journals. He received the Gold Medal for his M.S. degree from IMSciences. He has conducted and led software development projects and collaborative scientific research projects with academia and industry.

MUHAMMAD WASIM is currently serving as Lecturer in the Department of Computer Science at City University. He obtained his Bachelor's and Master's degrees from City University of Science and Information Technology. With a focus on deep learning models, he actively explores advancements in the field. Email:Muhammadwasim443@gmail.com

FERNANDO MOREIRA Fernando Moreira received the degree in computer science in 1992, the M.Sc. and Ph.D. degrees in electronic engineering from the Faculty of Engineering, University of Porto, in 1997 and 2003, respectively, and the Habilitation degree in 2018. He has been a member of the Department of Science and Technology at Portucalense University, Portugal, since 1992. He was the Head of the Department of Science and Technology, from May 2018 to February 2022. He is currently a Full Professor at Portucalense University. He is also a Full Professor and a Visiting Professor at the University of Porto Business School. He teaches subjects related to undergraduate and postgraduate studies. He was the Computation Co-ordinator of the M.Sc. in computation during the last ten years. He supervises several Ph.D. and M.Sc. students. He organized several special issues from JCR journals. He co-authors over 250 scientific publications with peer reviews in national and international journals and conferences. His research interests include mobile computing, ICT in higher education, mobile learning, social business, and digital transformation. He serves as a member of the editorial advisory board for several journals and books. He has already regularly served as a member of program and scientific committees for national and international conferences. He is associated with NSTICC, ACM, and IEEE. He was awarded the Atlas Elsevier Award, in April 2019. He holds editorial experience and he is the co-editor of several books.