Diffusion-based Hierarchical Negative Sampling for Multimodal Knowledge Graph Completion

Guanglin Niu[1](✉) and Xiaowei Zhang[2]

[1] School of Artificial Intelligence, Beihang University, Beijing, China
beihangngl@buaa.edu.cn
[2] College of Computer Science and Technology, Qingdao University, Qingdao, China
xiaowei19870119@sina.com

Abstract. Multimodal Knowledge Graph Completion (MMKGC) aims to address the critical issue of missing knowledge in multimodal knowledge graphs (MMKGs) for their better applications. However, both the previous MMGKC and negative sampling (NS) approaches ignore the employment of multimodal information to generate diverse and high-quality negative triples from various semantic levels and hardness levels, thereby limiting the effectiveness of training MMKGC models. Thus, we propose a novel Diffusion-based Hierarchical Negative Sampling (DHNS) scheme tailored for MMKGC tasks, which tackles the challenge of generating high-quality negative triples by leveraging a Diffusion-based Hierarchical Embedding Generation (DiffHEG) that progressively conditions on entities and relations as well as multimodal semantics. Furthermore, we develop a Negative Triple-Adaptive Training (NTAT) strategy that dynamically adjusts training margins associated with the hardness level of the synthesized negative triples, facilitating a more robust and effective learning procedure to distinguish between positive and negative triples. Extensive experiments on three MMKGC benchmark datasets demonstrate that our framework outperforms several state-of-the-art MMKGC models and negative sampling techniques, illustrating the effectiveness of our DHNS for training MMKGC models. The source codes and datasets of this paper are available at https://github.com/ngl567/DHNS.

Keywords: Multimodal knowledge graph completion · Diffusion model · Hierarchical negative sampling

1 Introduction

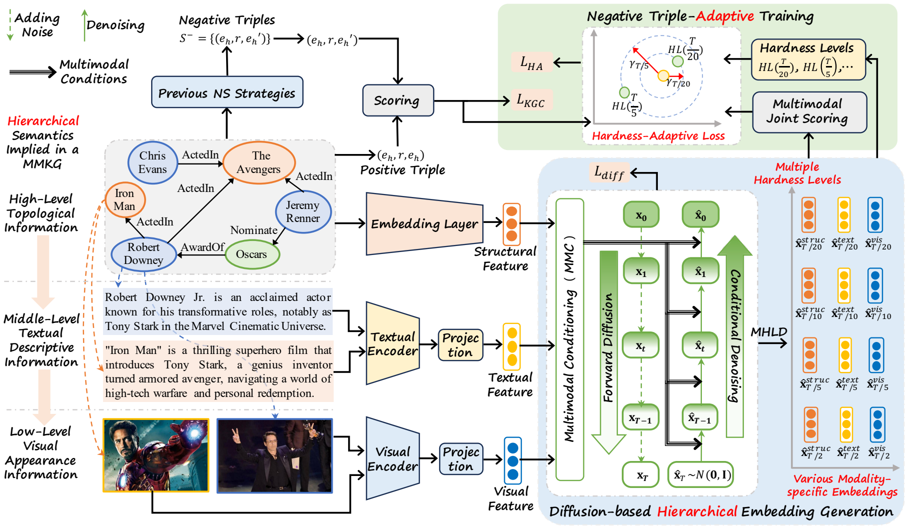
Multimodal knowledge graphs (MMKGs) have become a powerful paradigm for representing symbolic knowledge, integrating diverse modalities such as text, images, and audio [14]. These graphs are extensively applied across various domains such as multimodal question answering systems [12], where they enrich contextual relevance by representing multimodal information.

Knowledge graph completion (KGC) is an essential task in the context of MMKGs, as real-world MMKGs are frequently incomplete due to constraints in data collection and curation [41]. The objective of MMKGC is to infer missing knowledge and then enhance the MMKG's completeness and utility. In the training of MMKGC models, negative sampling (NS) is a critical component, given the scarcity of negative triples in any MMKG [34]. NS generates negative triples that contrast with the positive triples in MMKGs, allowing the model to learn semantic boundaries and associations among entities and relations.

However, existing negative sampling (NS) strategies, such as random sampling and adversarial-based methods, face three key challenges. First, **Current NS techniques [19] mainly rely on topological features while neglecting semantics from diverse modalities** especially in MMKGs, leading to simple or invalid negative triples. Second, although some approaches based on generative adversarial networks (GANs) [4] or self-adversarial strategies [25] can assess the quality of sampled negative triples, **their assessment depends on pre-sampled triples and the performance of knowledge graph completion (KGC) models, rather than directly generating high-quality negative triples**. Third, current KGC models employ a fixed margin for training [5], **making it challenging for a one-margin-fits-all training scheme to be effective across different hardness levels of negative triples**.

To address these challenges, we propose a novel **D**iffusion-based **H**ierarchical **N**egative **S**ampling (DHNS) paradigm for MMKGC, which is motivated by the recent success of diffusion models in various generative tasks. By leveraging the powerful denoising diffusion probabilistic model (DDPM) [9], we develop a **Diffusion-based Hierarchical Embedding Generation (DiffHEG)** module, which could directly generate diverse entity embeddings for composing negative triples, instead of the traditional NS paradigm of sampling entities. Specifically, the synthesized negative triples are obtained concerning hierarchical semantics from both multiple modality-specific embeddings and various hardness levels via conditional denoising at different time steps. Furthermore, these high-quality and diverse negative triples could be fed into any KGC model, enhancing its ability to distinguish between positive and negative triples. Specifically, we develop a **Negative Triple-Adaptive Training (NTAT)** mechanism with Hardness-Adaptive Loss (HAL) to enhance the KGC model's learning capability for different hardness levels of negative triples. An architecture of our DHNS framework is illustrated in Fig. 1. Our contributions can be summarized as:

- As we can be concerned, it is the first effort to leverage the diffusion model's capabilities within the context of MMKGC for negative sampling. Our NS module DiffHEG captures the diverse semantics of different modalities to generate hierarchical and high-quality negative triples while directly controlling the hardness levels with diffusion time steps.
- Based on the generated negative triples, we develop a hardness-adaptive training objective in which the pivotal parameter margin is adaptive to hardness levels of negative triples. It facilitates the comprehensive training of an MMKGC model for a diverse range of negative triples.
- Extensive experiments are conducted on three MMKGC benchmark datasets to compare the performance of our DHNS model against some state-of-the-


Adding Noise
Denoising
Multimodal Conditions
Negative Triples
$S^- = \{(e_h, r, e_h')\}$
(e_h, r, e_h')
Previous NS Strategies
Scoring
L_{HA}
L_{KGC}
Negative Triple-Adaptive Training
$HL(\frac{T}{20})$
$\gamma_{T/5}$
$\gamma_{T/20}$
$HL(\frac{T}{5})$
Hardness-Adaptive Loss
Hardness Levels
$HL(\frac{T}{20})$, $HL(\frac{T}{5})$,···
Multimodal Joint Scoring
Hierarchical Semantics Implied in a MMKG
High-Level Topological Information
Middle-Level Textual Descriptive Information
Low-Level Visual Appearance Information
Chris Evans
ActedIn
The Avengers
ActedIn
Iron Man
ActedIn
Jeremy Renner
ActedIn
Nominate
Robert Downey
AwardOf
Oscars
(e_h, r, e_h)
Positive Triple
Embedding Layer
Structural Feature
Textual Encoder
Projec tion
Textual Feature
Visual Encoder
Projec tion
Visual Feature
L_{diff}
Multimodal Conditioning (MMC)
Forward Diffusion
Conditional Denoising
$\mathbf{x}_0$
$\mathbf{x}_1$
$\mathbf{x}_t$
$\mathbf{x}_{T-1}$
$\mathbf{x}_T$
$\hat{\mathbf{x}}_0$
$\hat{\mathbf{x}}_1$
$\hat{\mathbf{x}}_t$
$\hat{\mathbf{x}}_{T-1}$
$\hat{\mathbf{x}}_T \sim N(\mathbf{0}, \mathbf{I})$
MHLD
Multiple Hardness Levels
$\hat{\mathbf{x}}^{struc}_{T/20}$
$\hat{\mathbf{x}}^{text}_{T/20}$
$\hat{\mathbf{x}}^{vis}_{T/20}$
$\hat{\mathbf{x}}^{struc}_{T/10}$
$\hat{\mathbf{x}}^{text}_{T/10}$
$\hat{\mathbf{x}}^{vis}_{T/10}$
$\hat{\mathbf{x}}^{struc}_{T/5}$
$\hat{\mathbf{x}}^{text}_{T/5}$
$\hat{\mathbf{x}}^{vis}_{T/5}$
$\hat{\mathbf{x}}^{struc}_{T/2}$
$\hat{\mathbf{x}}^{text}_{T/2}$
$\hat{\mathbf{x}}^{vis}_{T/2}$
Various Modality-specific Embeddings
Diffusion-based Hierarchical Embedding Generation

Robert Downey Jr. is an acclaimed actor known for his transformative roles, notably as Tony Stark in the Marvel Cinematic Universe.

"Iron Man" is a thrilling superhero film that introduces Tony Stark, a genius inventor turned armored avenger, navigating a world of high-tech warfare and personal redemption.

Fig. 1. The whole framework of our DHNS. MHLD means multiple hardness-level denoising. $\mathbf{x}_{0:T}$ and $\hat{\mathbf{x}}_{0:T}$ are the noised and the denoised embeddings in the range of time steps $[0, T]$ corresponding to an entity. $\mathbf{x}^{struc}_{T/20}$, $\mathbf{x}^{text}_{T/20}$ and $\mathbf{x}^{vis}_{T/20}$ are three modality-specific (structural/textual/visual) denoised embeddings at the time step $T/20$. $\gamma_{T/20}$ denotes the margin adaptive to the negative triples with the hardness level $HL(\frac{T}{20})$.

art MMKGC models and negative sampling strategies, demonstrating the robustness and effectiveness of our DHNS framework for MMKGC tasks.

2 Related Work

2.1 Multimodal Knowledge Graph Completion

Existing MMKGC approaches extend traditional knowledge graph embedding (KGE) techniques to handle multi-modal information [33]. KGE models represent entities and relations in continuous numerical spaces. Early models like TransE [3] propose a translational distance-based score function, where relations are represented as translation operations between entities. DistMult [35] and ComplEx [27] use bilinear models to capture both symmetric and antisymmetric relations. RotatE [25] and QuatE [36] introduce rotational and quaternion-based embeddings, respectively, to model complex relational patterns.

In particular, MMKGC models typically involve designing additional embeddings to represent multi-modal information, such as textual descriptions and images of entities, and incorporating them into the score function to evaluate the plausibility of each triple. For instance, IKRL [33] extracts visual features using a pre-trained visual encoder and combines them with structural embeddings from TransE to assess the plausibility of triples. TransAE [32] and TBKGC [20] extend

IKRL by integrating both visual and textual information. RSME [29] employs a gate mechanism to ensure that the most relevant multi-modal features are incorporated into entity embeddings. AdaMF [38] uses a generator to produce adversarial samples and a discriminator to measure their plausibility in an adversarial training framework. LAFA [24] considers the relationships between entities and different modalities, focusing on link-aware aggregation of multi-modal information. VISTA [11] designs three transformer-based encoders to incorporate visual and textual embeddings for predicting missing triples. However, these MMKGC models primarily focus on entity and relation representation learning, neglecting the importance of generating high-quality negative triples from rich multi-modal information to better guide the training process.

2.2 Negative Sampling in Knowledge Graph Embedding

Negative sampling (NS)[19] is a widely used technique in KGE that generates negative triples not present in knowledge graphs (KGs), enhancing model training by contrasting them with positive triples. Traditional strategies, such as random entity replacement, are simple but often produce false negatives or low-quality triples, leading to ambiguous training signals [4]. Bernoulli sampling [31] employs a Bernoulli distribution to replace entities to generate higher-quality negative triples. KBGAN [4] and IGAN [30] use Generative Adversarial Networks (GANs) to select harder negatives that are difficult for KGE models to distinguish from positives. NSCaching [40] utilizes additional memory to store and efficiently sample high-quality negative triples during training. SANS [1] leverages graph structure information for sampling high-quality negatives. However, these NS strategies are primarily designed for unimodal KGC models and do not leverage the multi-level semantics in multimodal information, which is crucial for generating diverse negative triples. MMRNS [34] introduces a relation-enhanced NS mechanism using knowledge-guided cross-modal attention to generate more challenging negatives from multimodal data. MANS [37] emphasizes modality-aware NS to align structural and multimodal information, enhancing negative triple quality. Despite these advances, these multimodal NS approaches remain within the sampling paradigm, lacking control over negative triple generation.

2.3 Diffusion Models

In recent years, diffusion models, particularly the Denoising Diffusion Probabilistic Model (DDPM) [9], have become pivotal in AI-generated content by progressively adding Gaussian noise to input data across time steps, forming a Markov chain based on a noise schedule. The reverse process involves training a neural network to predict and remove noise, thereby reconstructing the original data. In graph-structured data, DMNS[21] employs DDPM to generate negative nodes for link prediction, considering the query node's context, while KGDM [16] applies DDPM to estimate the probabilistic distribution of target entities for KGC. However, the application of diffusion models for NS in KGC, particularly in MMKGC, remains limited. Our work addresses this gap by leveraging diffusion models for negative sampling in MMKGC, adapting the model to MMKGs to generate diverse negative triples of varying hardness, capturing hierarchical multi-modal semantics and enhancing MMKGC model training.

3 Methodology

3.1 Preliminaries and Problem Definition

A multimodal knowledge graph (MMKG) extends the traditional knowledge graph (KG) and is defined as a tuple $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T}, \mathcal{M})$. Here, $\mathcal{E}$ and $\mathcal{R}$ denote the sets of entities and relations, and $\mathcal{T}$ is the set of triples in the form (e_h, r, e_t), where $e_h, e_t \in \mathcal{E}$ and $r \in \mathcal{R}$. The component $\mathcal{M}$ indicates multimodal information, including images and textual descriptions associated with entities.

The objective of MMKGC is to predict missing triples in $\mathcal{G}$. Given an incomplete triple query, such as $(e_h, r, ?)$ or $(?, r, e_t)$, where e_h and e_t are known entities and r is a relation, the task is to evaluate the plausibility of candidate triples (e_h, r, e_t) or (e, r, e_t) by scoring them based on their learned embeddings in $\mathcal{G}$. During training, the primary goal is to distinguish between positive triples (e_h, r, e_t) and their corresponding negative triples (e'_h, r, e_t) or (e_h, r, e'_t) where e'_h and e'_t are the corrupted entities sampled from $\mathcal{E}$. To enhance the model's discriminative capability, negative sampling aims to generate hard negative triples that are semantically similar to their positive counterparts. In this paper, we represent the original embedding of an entity or relation as $\mathbf{x}$.

3.2 Diffusion-based Hierarchical Embedding Generation

To supplement the negative triples obtained by sampling explicit entities, we propose a novel Diffusion-based Hierarchical Embedding Generation (DiffHEG) module. This DHNS module enables the generation of hierarchical entity embeddings to compose high-quality negative triples by conditioning on entities and relations as well as multimodal information. Besides, these negative triples vary multiple levels of hardness, which are regulated by diffusion time steps.

Forward Diffusion and Reverse Denoising Procedures. Following the basic architecture of DDPM, in the forward diffusion process, the input entity embedding $\mathbf{x}_0$ gradually has noise added to it, resulting in a sequence of embeddings that converge to pure Gaussian noise $\mathbf{x}_T$, in which T indicates the total time steps of the diffusion process. Specifically, the forward diffusion process can be represented by a Markov chain as follows:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{1-\alpha_t}\mathbf{x}_{t-1}, \alpha_t \mathbf{I}) \tag{1}$$

where α_t represents the variance that could be constants or learnable by some scheduling mechanisms at arbitrary time step t. The complete process of adding noise can be expressed as $q(\mathbf{x}_{1:T}|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})$. Thus, the closed form of noisy entity embedding $\mathbf{x}_t$ is calculated by

$$\mathbf{x}_t = \sqrt{\bar{\beta}_t}\mathbf{x}_0 + \sqrt{1-\bar{\beta}_t}\epsilon_t \tag{2}$$

where $\beta_t = 1 - \alpha_t$ and $\bar{\beta}_t = prod_{i=1}^{t}\beta_i$. $\epsilon_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ is the noise.

Furthermore, the reverse diffusion process iteratively denoises the noisy entity embedding $\mathbf{x}_t$ to obtain the synthetic entity embedding. Given a noisy entity embedding $\mathbf{x}_t$ at time step t, the reverse process is conditioned on the embeddings of the observed entity $\mathbf{x}_h$ and the relation $\mathbf{x}_r$, yielding:

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, PE(t), C(\mathbf{x}_e, \mathbf{x}_r)), \alpha_t\mathbf{I}) \tag{3}$$

where θ denotes the parametrization of the diffusion model. $PE(t)$ denotes the positional embedding of the time step t and will be declared in the following, and $C(\mathbf{x}_e, \mathbf{x}_r)$ indicates the condition derived from an observed entity e and the associated relation r in the triple for guiding the generation of another entity embedding to constitute a negative triple. Particularly, $\mu_\theta(\mathbf{x}_t,\ PE(t), C(\mathbf{x}_e, \mathbf{x}_r))$ indicates the mean of the Gaussian distribution and is parameterized as:

$$\mu_\theta(\mathbf{x}_t, PE(t), C(\mathbf{x}_e, \mathbf{x}_r)) = \frac{1}{\sqrt{\beta_t}}\mathbf{x}_t - \frac{\beta_t}{\sqrt{\beta_t}\sqrt{1-\bar{\beta}_t}}\epsilon_\theta(\hat{\mathbf{x}}_t, PE(t), C(\mathbf{x}_e, \mathbf{x}_r)) \tag{4}$$

in which $\epsilon_\theta(\hat{\mathbf{x}}_t, PE(t), C(\mathbf{x}_e, \mathbf{x}_r)$ is the estimated noise obtained through the following conditional denoising operation. From Eq. 3 and Eq.4, the generated entity embedding via eliminating predicted noises at the beginning and each intermediate time step in the reverse diffusion process could be rewritten as:

$$\hat{\mathbf{x}}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \hat{\mathbf{x}}_0 = \frac{1}{\sqrt{\beta_t}}\hat{\mathbf{x}}_1 - \frac{\beta_t}{\sqrt{\beta_t}\sqrt{1-\bar{\beta}_t}}\epsilon_\theta(\hat{\mathbf{x}}_1, PE(1), C(\mathbf{x}_e, \mathbf{x}_r)) \tag{5}$$

$$\hat{\mathbf{x}}_{t-1} = \frac{1}{\sqrt{\beta_t}}\hat{\mathbf{x}}_t - \frac{\beta_t}{\sqrt{\beta_t}\sqrt{1-\bar{\beta}_t}}\epsilon_\theta(\hat{\mathbf{x}}_t, PE(t), C(\mathbf{x}_e, \mathbf{x}_r)) + \alpha_t\epsilon_t \tag{6}$$

where $\hat{\mathbf{x}}_T$ implies the pure noise at the last step T, $\hat{\mathbf{x}}_0$ and $\hat{\mathbf{x}}_{t-1}$ denote the generated entity embedding at the beginning step and the intermediate time step in the range of $[1, T-1]$. Following the idea of DDPM that promotes the denoise process with more diversity, the item $\alpha_t\epsilon_t$ is utilized to add a random noise for obtaining each intermediate denoise result $\hat{\mathbf{x}}_{t-1}$.

Multimodal Conditioning and Multiple Hardness-Level Denoising. Conditional denoising is composed of two sub-modules: Multimodal Conditioning (MMC) and Multiple Hardness-Level Denoising (MHLD). The MMC module computes a conditional embedding $C(\mathbf{x}_e, \mathbf{x}_r)$ by integrating entity and relation embeddings across multiple semantic levels from structural, visual, and textual features. This integration captures rich contexts of entities in the MMKG, facilitating the denoising process. Given the various strategies for modeling triples in KGs, we use several condition calculation mechanisms, detailed as follows:

Hardmard multiplication, which is available for rotation-based KGE models such as RotatE [25] and QuatE [36], is formulated as:

$$C(\mathbf{x}_e, \mathbf{x}_r) = \mathbf{x}_e \circ \mathbf{x}_r \tag{7}$$

where ∘ means Hardmard multiplication.

Bilinear multiplication, which is suitable for bilinear interaction-based KGE models such as RESCAL [22] and DistMult [35], formulated as:

$$C(\mathbf{x}_e, \mathbf{x}_r) = \mathbf{x}_e \times \mathbf{x}_r \tag{8}$$

where × indicates element-wise multiplication.

Addition, which is available for translation-based KGE models such as TransE [3] and TransH [31], formulated as:

$$C(\mathbf{x}_e, \mathbf{x}_r) = \mathbf{x}_e + \mathbf{x}_r \tag{9}$$

The MHLD module applies a denoising transformation on the noisy entity embedding $\mathbf{x}_t$, informed by the conditional embedding $C(\mathbf{x}_e, \mathbf{x}_r)$ and the time embedding $PE(t)$. In particular, $PE(t)$ is modulated by a sinusoidal positional embedding layer that is frequently used in Transformer-based models to guarantee the temporal constraints among time steps and effectively guide the denoising at each time step t. For instance, $[PE(t)]_{2i} = sin(t/1000^{\frac{2i}{d}})$ and $[PE(t)]_{2i+1} = cos(t/1000^{\frac{2i}{d}})$ where d is the dimension of $PE(t)$. Given the inputs of $\mathbf{x}_t$, $PE(t)$, and $C(\mathbf{x}_e, \mathbf{x}_r)$, the denoising function can be represented as:

$$\epsilon_\theta(\mathbf{x}_t, PE(t), C(\mathbf{x}_e, \mathbf{x}_r)) = LayerNorm(MLP(\mathbf{x}_t, PE(t), C(\mathbf{x}_e, \mathbf{x}_r))) \tag{10}$$

where simple multi-layer perceptron (MLP) and layer normalization layer (LayerNorm) are leveraged to predict the noise with the learnable parameters θ.

The MMC module integrates triples, images, and texts to generate diverse, semantically rich negative triples. Specifically, by corrupting the tail entity of a positive triple (e_h, r, e_t), we input the head entity's structural feature $\mathbf{x}_{eh}^{struc}$, visual feature $\mathbf{x}_{eh}^{vis}$, and textual feature $\mathbf{x}_{eh}^{text}$, alongside the relation embedding $\mathbf{x}_r$, into MHLD module to compute multimodal conditions. These conditions guide the reverse process outlined in Eqs. 5-6 and 10, yielding various modality-specific embeddings $\hat{\mathbf{x}}_{et,t}^{struc}$, $\hat{\mathbf{x}}_{et,t}^{vis}$, and $\hat{\mathbf{x}}_{et,t}^{text}$ for constructing the generated negative triples. The DHNS model generates multimodal implicit entity embeddings for the corrupted tail entity by minimizing a denoising diffusion loss:

$$\begin{aligned}\mathcal{L}_{diff} =& \|\epsilon_\theta(\hat{\mathbf{x}}_{et,t}^{struc}, PE(t), C(\mathbf{x}_{eh}^{struc}, \mathbf{x}_r)) - \epsilon_t\|^2 + \|\epsilon_\theta(\hat{\mathbf{x}}_{et,t}^{vis}, PE(t), \\ & C(\mathbf{x}_{eh}^{vis}, \mathbf{x}_r)) - \epsilon_t\|^2 + \|\epsilon_\theta(\hat{\mathbf{x}}_{et,t}^{text}, PE(t), C(\mathbf{x}_{eh}^{text}, \mathbf{x}_r)) - \epsilon_t\|^2\end{aligned} \tag{11}$$

where the denoising diffusion loss is formulated as the mean squared error (MSE) between the predicted noises from structural, textual, and visual levels and the actual noise added during the forward diffusion process. This loss is optimized using the Adam optimizer with separate learning rates to ensure stability.

The entity embeddings are generated by starting from complete noise and iteratively removing the predicted noise at each time step, a process inherent to diffusion models. To control the hardness of the negative triples, we modulate the diffusion time steps. Smaller time steps yield negatives closer to positives, representing higher hardness, while larger time steps produce easily distinguishable negatives. We formalize the hardness level of the generated entity embedding $\hat{\mathbf{x}}_t$ as inversely proportional to the time step t, such as $HL(\hat{\mathbf{x}}_t) \propto \frac{1}{t}$.

To ensure diversity, we sample time steps at specific intervals, such as $t = T/20, T/10, T/5, T/2$, to obtain a set of generated entity embeddings $G^- = \{neg_t | t = T/20, T/10, T/5, T/2\}$, where $neg_t = (\hat{\mathbf{x}}_t^{struc}, \hat{\mathbf{x}}_t^{vis}, \hat{\mathbf{x}}_t^{text})$ represents the modality-specific embeddings at time step t. To balance quality, embeddings closer to the halfway point of the diffusion process are assigned higher weights.

The DiffHEG mechanism provides hierarchical control over negative triple generation, considering both hardness levels and modalities, offering a more controllable approach than random replacements. By combining diverse negative triples, we could enhance the training signal for KGE models, potentially improving their performance and generalization.

3.3 Negative Triple-Adaptive Training

Considering the varying hardness levels of negative affect the KGE model's ability to distinguish between positive and negative triples, we propose a well-designed Negative Triple-Adaptive Training (NTAT) mechanism for training KGE models based on the generated negative triples. Specifically, HTAT consists of two modules: multimodal joint scoring and Hardness-Adaptive Loss (HAL). The multimodal joint scoring computes a joint score for generated negative triples by integrating entity embeddings from structural, visual, and textual modalities. Take the generated negative triple via corrupting tail entity (h, r, neg_t) (neg_t is just a formalized symbol that represents the generated tail entity embedding at time step t in this negative triple) as instance, the multimodal joint scoring of this generated negative triple is formalized as:

$$S(h, r, neg_t) = (E(\mathbf{x}_h, \mathbf{x}_r, \hat{\mathbf{x}}_t^{struc}) + E(\mathbf{x}_h, \mathbf{x}_r, \hat{\mathbf{x}}_t^{vis}) + E(\mathbf{x}_h, \mathbf{x}_r, \hat{\mathbf{x}}_t^{text}))/3 \quad (12)$$

where $\hat{\mathbf{x}}_t^{struc}$, $\hat{\mathbf{x}}_t^{vis}$ and $\hat{\mathbf{x}}_t^{text}$ are the previously defined three modality-specific entity embeddings. $E(\cdot)$ represents the score function of any KGE model to evaluate the plausibility of a triple.

Furthermore, we introduce HAL which dynamically adjusts the margin based on the hardness level of negative triples generated by the DiffHEG module. This loss assigns smaller margins to harder negatives and larger margins to easier ones, enabling the model to learn effectively from challenging cases while maintaining robustness. The hardness-adaptive loss is defined as:

$$\mathcal{L}_{HA} = -log\sigma(\gamma_t - E(h, r, t)) - \frac{1}{|G^-|} \sum_{neg_t \in G^-} w(neg_t) \cdot log\sigma((S(h, r, neg_t) - \gamma_t) \quad (13)$$

where γ_t is the margin adaptive to the hardness level $HL(t)$, and $w(neg_t)$ are the weights assigned to each negative triple. σ represents the sigmoid function. $|G^{-}|$ is the size of the negative entity embedding set corresponding to each positive triple (h, r, t). Besides, the negative triples obtained via the sampling techniques such as randomly sampling and Bernoulli sampling could be also leveraged for training KGE models with the traditional KGC loss:

$$\mathcal{L}_{KGC} = -log\sigma(\gamma - E(h,r,t)) - \frac{1}{|S^{-}|}\sum_{t' \in S^{-}} log\sigma((S(h,r,t') - \gamma) \tag{14}$$

in which γ indicates the fixed margin. t' is a corrupted entity in the negative triple set S^{-} obtained from previous NS techniques. The total loss for training a KGE model is a combination of $\mathcal{L}_{KGC}$ and $\mathcal{L}_{HA}$, weighted by a hyper-parameter λ for a trade-off between generated and sampled negative triples:

$$\mathcal{L} = \mathcal{L}_{KGC} + \lambda \cdot \mathcal{L}_{HA} \tag{15}$$

For a comprehensive understanding of the training procedure of our DHNS framework, the pseudo-code of training DHNS is provided in Algorithm 1.

Algorithm 1: Training Procedure of DHNS

```
Input: A batch of positive triples $\mathcal{T}$ from the KG $\mathcal{G}$ and the
       corresponding pre-sampled negative triples $S^{-}$ via a previous
       NS strategy, the pre-trained visual and textual embeddings of
       entities encoded from the multimodal information $\mathcal{M}$.
Output: The MMKGC model trained via DHNS.
for each triple $(e_h, r, e_t) \in \mathcal{T}$ do
    // Training DiffHEG module
    Obtain the original modality-specific embeddings of $(e_h, r, e_t)$:
      $\mathbf{x}_{eh}^{struc}$, $\mathbf{x}_r$, $\mathbf{x}_{et}^{struc}$, $\mathbf{x}_{eh}^{vis}$, $\mathbf{x}_{et}^{vis}$, $\mathbf{x}_{eh}^{text}$, $\mathbf{x}_{et}^{text}$;
    Calculate the noised entity embeddings at the time step $t$ as in Eq. 2:
      $\mathbf{x}_{eh,t}^{struc} = \sqrt{\bar{\beta}_t}\mathbf{x}_{eh}^{struc} + \sqrt{1-\bar{\beta}_t}\epsilon_t$, $\mathbf{x}_{et,t}^{struc} = \sqrt{\bar{\beta}_t}\mathbf{x}_{et}^{struc} + \sqrt{1-\bar{\beta}_t}\epsilon_t$;
    $\mathbf{x}_{eh,t}^{vis} = \sqrt{\bar{\beta}_t}\mathbf{x}_{eh}^{vis} + \sqrt{1-\bar{\beta}_t}\epsilon_t$, $\mathbf{x}_{et,t}^{vis} = \sqrt{\bar{\beta}_t}\mathbf{x}_{et}^{vis} + \sqrt{1-\bar{\beta}_t}\epsilon_t$;
    $\mathbf{x}_{eh,t}^{text} = \sqrt{\bar{\beta}_t}\mathbf{x}_{eh}^{text} + \sqrt{1-\bar{\beta}_t}\epsilon_t$, $\mathbf{x}_{et,t}^{text} = \sqrt{\bar{\beta}_t}\mathbf{x}_{et}^{text} + \sqrt{1-\bar{\beta}_t}\epsilon_t$;
    Predict the noise at the time step $t$ following Eqs. 7-10;
    Optimize parameters $\theta$ of the diffusion model by minimizing $\mathcal{L}_{diff}$;
    // Generating hierarchical entity embeddings
    for $t = T-1, \cdots, 0$ do
        Calculate the the modality-specific embeddings as in Eqs. 5-6:
          $neg_{eh,t} = (\hat{\mathbf{x}}_{eh,t}^{struc}, \hat{\mathbf{x}}_{eh,t}^{vis}, \hat{\mathbf{x}}_{eh,t}^{text})$, $neg_{et,t} = (\hat{\mathbf{x}}_{et,t}^{struc}, \hat{\mathbf{x}}_{et,t}^{vis}, \hat{\mathbf{x}}_{et,t}^{text})$;
    Generate the set of negative entity embeddings with multiple
      hardness levels: $G^{-} = \{(neg_{eh,t}, neg_{et,t}) | t = T/20, T/10, T/5, T/2\}$;
    // Training a KGE model with NTAT module
    Calculate the hardness-adaptive loss $\mathcal{L}_{HA}$ with $G^{-}$ as in Eq. 13;
    Calculate the KGC loss $\mathcal{L}_{KGC}$ with $S^{-}$ as in Eq. 14;
    Calculate $\mathcal{L}$ as in Eq. 15 and minimize it to optimize parameters of
      the KGE model;
```

Table 1. Statistics of three MMKGC benchmark datasets.

Dataset	#Entity	#Relation	#Image	#Text	#Train	#Valid	#Test
DB15K	12842	279	12818	9078	79222	9902	9904
MKG-W	15000	169	14463	14123	34196	4276	4274
MKG-Y	15000	28	14244	12305	21310	2665	2663

4 Experiments

4.1 Experiment Settings

MMKGC Datasets. We conduct experiments on three MMKGC benchmark datasets: DB15K [15], MKGW and MKG-Y [17]. The triples stored in DB15K are extracted from DBpedia while the structural knowledge in MKG-W (Multimodal-Wikidata) and MKG-Y (Multimodal KG-YAGO) are extracted from Wikidata [28] and YAGO [8], respectively. The images in these three datasets are collected by image search engines and the textual descriptions are obtained from DBpedia. The statistics of these datasets are declared in Table 1.

Baseline Models. We select three categories of baseline approaches to compare and evaluate the performance of our DHNS framework.

(1) **Unimodal KGE models**: we select some typical KGE models learning structural features to evaluate triples, including TransE [3], TransD [10], DistMult [35], ComplEx [27], RotatE [25], PairRE [7], and GC-OTE [26].

(2) **Multimodal KGC models**: we compare our framework with some state-of-the-art MMKGC models that could learn the multimodal features (visual features and/or textual features) together with structural features to represent a triple, including IKRL [33], TBKGC [20], TransAE [32], MMKRL [18], RSME [29], VBKGC [39], OTKGE [6] and AdaMF [38].

(3) **Negative sampling-based models**: some NS-based models are selected for evaluating on both MMKGC and NS performances, including uniform sampling (Uniform) [3], Bernoulli sampling (Bern) [31], NSCaching (NSCach) [40], KBGAN [4], SANS [1], NS-KGE [13], MANS [37] and MMRNS [34].

Implementation Details. We implement the DHNS model with Pytorch and conduct the experiments on one NVIDIA GeForce 4090 GPU. For a fair comparison, the preprocess procedures of extracting visual and textual features are the same as AdaMF [38] and MMRNS [34]. The visual features are extracted via BEiT [2] and the textual features are extracted using SBERT [23]. Particularly, hyper-parameters of KGE models are fixed following state-of-the-art models for a fair comparison while those of our DiffHEG module are tuned. Specifically, the total timestep is selected from $\{20, 50, 70, 100\}$, learning rate is tuned in $\{2e^{-3}, 1e^{-4}, 5e^{-4}\}$. We employ two frequently-used metrics for evaluation: mean reciprocal rank of all the correct instances (MRR) and proportion of the correct instances ranked in the top N (HN, $N = 1, 3, 10$). Particularly, we employ the filter setting [3] that removes the candidate triples already observed in the training set for all the results. The results of baselines in Table 2 and Table 3 are obtained from [38] and the results in Table 4 are derived from [34].

4.2 Experimental Results

Main Results. The experimental results shown in Table 2 and Table 3 show that our proposed framework DHNS integrated with RotatE achieves the highest or second-highest scores in terms of all the metrics across all datasets. Specifically, DHNS consistently outperforms a variety of state-of-the-art baseline models, including unimodal KGC and MMKGC models as well as NS-based models, illustrating the superior performance of our proposed framework DHNS consisting of DiffHEG and NTAT modules on MMKGC tasks.

Table 2. Evaluation results (%) of our framework DHNS integrated with RotatE model and some state-of-the-art baseline models on MMKGC datasets DB15K and MKG-W. The best results are marked **bold** and the second-best results are underlined.

Model		DB15K				MKG-W			
		MRR	H1	H3	H10	MRR	H1	H3	H10
Unimodal KGC Models	TransE	24.86	12.78	31.48	47.07	29.19	21.06	33.20	44.23
	TransD	21.52	8.34	29.93	44.24	26.84	19.65	31.48	42.68
	DistMult	23.03	14.78	26.98	40.59	20.95	15.89	22.88	36.80
	ComplEx	27.48	18.13	31.57	45.37	28.09	21.45	30.89	44.77
	RotatE	29.28	17.87	36.12	49.66	30.79	21.98	36.42	46.73
	PairRE	31.13	21.67	36.91	51.98	33.29	25.00	38.67	46.71
	GC-OTE	31.85	22.11	36.52	51.18	33.92	26.55	35.96	46.05
MMKGC Models	IKRL	26.82	14.09	34.93	49.09	32.36	26.11	34.75	44.07
	TBKGC	28.08	15.61	37.03	49.86	31.80	25.31	33.91	44.58
	TransAE	28.09	21.25	37.17	49.17	30.01	21.23	34.91	44.72
	MMKRL	26.81	13.85	35.01	49.39	29.42	22.54	31.49	43.44
	RSME	29.76	**24.15**	32.12	49.23	29.23	23.31	31.09	40.83
	VBKGC	30.61	19.75	37.18	49.41	30.69	24.55	33.07	44.62
	OTKGE	23.86	18.45	25.89	34.23	34.36	28.85	36.25	44.88
	AdaMF	32.51	21.31	39.67	51.68	34.27	27.21	37.86	47.21
NS-based Models	KBGAN	25.73	9.91	36.95	51.93	29.47	22.21	34.87	40.64
	MANS	28.82	16.87	38.54	51.51	32.04	27.48	37.48	41.62
	MMRNS	29.67	17.89	36.86	51.01	34.47	28.93	38.63	47.48
	DHNS	**34.36**	23.34	**41.56**	**53.72**	**35.68**	**28.98**	**38.68**	**48.12**

Besides, MMKGC models generally outperform unimodal KGC and baseline NS-based models, demonstrating the importance of incorporating supplementary multimodal information for representing KG embeddings. Notably, our framework DHNS performs better than both MMKGC and NS-based models, verifying

Table 3. Evaluation results (%) of our framework DHNS and some state-of-the-art baseline models on MMKGC dataset MKG-Y.

Model		MKG-Y			
		MRR	H1	H3	H10
Unimodal KGC Models	TransE	30.73	23.45	35.18	43.37
	TransD	26.39	17.01	33.05	40.41
	DistMult	25.04	19.32	27.80	39.95
	ComplEx	28.94	23.11	31.07	43.48
	RotatE	29.96	20.70	36.38	46.49
	PairRE	32.18	25.24	37.58	44.98
	GC-OTE	32.95	26.77	36.44	44.08
MMKGC Models	IKRL	33.22	30.37	34.28	38.26
	TBKGC	33.39	30.47	33.74	37.92
	TransAE	28.10	25.31	29.19	33.03
	MMKRL	30.94	27.00	32.53	36.07
	RSME	34.44	31.78	36.07	39.79
	VBKGC	34.03	31.76	35.73	37.72
	OTKGE	35.51	31.97	37.18	41.38
	AdaMF	38.06	33.49	40.40	45.48
NS-based Models	KBGAN	29.71	22.81	34.88	40.21
	MANS	29.93	25.25	31.35	34.49
	MMRNS	33.32	30.50	35.37	45.47
	DHNS	**39.11**	**34.70**	**41.23**	**46.66**

a more effective strategy through its NS mechanism with hierarchical semantics and multiple hardness levels, together with the negative triple-adaptive training paradigm to facilitate more effective training of KGE models.

Comparison of NS Strategies. The experimental results presented in Table 4 demonstrate that our DHNS is a robust and effective NS strategy for enhancing the performance of KGE models including TransE, DistMult and RotatE for MMKGC tasks. Specifically, DHNS consistently and significantly outperforms other NS strategies when integrated with RotatE on all the datasets and metrics. Besides, DHNS achieves the highest or near-highest MRR and H10 scores across KGE models TransE and DistMult as to all the datasets.

More interestingly, DHNS consistently outperforms traditional and state-of-the-art NS strategies including MMRNS and also employs multimodal information across various KGE models and datasets, illustrating the superiority of directly generating hierarchical embeddings to compose diverse and high-quality negative triples rather than sampling as in baseline NS strategies. This suggests that DHNS could be regarded as a pluggable module to generate more high-quality negative triples and further guide the training of KGE models more effectively for distinguishing between positive and negative triples.

Table 4. Performance comparison of our NS-based model DHNS and various state-of-the-art NS strategies on DB15K, MKG-W, and MKG-Y.

KGE	NS	DB15K			MKG-W			MKG-Y		
Models	Strategies	MRR	H1	H10	MRR	H1	H10	MRR	H1	H10
TransE	Uniform	24.86	12.78	47.07	29.19	21.06	44.23	30.73	23.45	43.37
	Bern	30.11	19.04	49.86	28.98	22.99	40.11	31.12	25.69	43.61
	KBGAN	24.00	5.47	50.36	24.21	14.51	40.45	26.92	19.90	37.41
	NSCach	**33.03**	**23.31**	50.29	28.90	22.24	40.97	29.51	24.78	38.02
	SANS	26.33	13.87	48.80	30.22	23.64	44.61	27.51	22.31	42.71
	MMRNS	26.61	13.40	50.60	33.51	26.25	**47.11**	34.81	28.59	**44.76**
	DHNS	32.63	20.95	**52.86**	**33.83**	**26.46**	46.85	**36.68**	**32.12**	43.62
DistMult	Uniform	20.99	14.78	39.59	20.99	15.93	38.06	25.04	19.33	35.95
	Bern	27.05	**20.04**	40.37	25.76	**21.02**	36.16	32.00	30.42	38.11
	KBGAN	23.36	16.43	36.30	19.11	7.69	37.44	16.99	9.25	33.33
	NSCach	24.68	19.33	39.74	24.70	20.31	37.24	27.30	23.55	35.67
	SANS	23.62	16.82	39.76	23.21	18.46	32.20	23.62	16.82	39.76
	MMRNS	25.92	14.06	40.42	**25.92**	19.73	**38.20**	32.78	27.99	41.13
	DHNS	**27.20**	19.07	**43.01**	25.86	20.33	36.53	**37.01**	**38.72**	**42.53**
RotatE	Uniform	29.28	17.87	49.66	33.67	26.80	46.73	34.95	29.10	45.30
	Bern	20.46	12.83	34.37	29.65	24.58	38.80	33.63	30.39	39.29
	NSCach	20.32	12.89	34.05	32.86	27.86	41.86	32.03	29.57	36.32
	SANS	30.51	19.13	50.72	33.32	27.35	44.67	35.28	29.29	44.93
	MMRNS	29.67	17.89	51.01	34.13	27.37	47.48	35.93	30.53	45.47
	DHNS	**34.36**	**23.34**	**53.72**	**35.68**	**28.98**	**48.12**	**39.11**	**34.70**	**46.66**

Ablation Study. We conduct an ablation study to provide insights into the effectiveness of each component of our framework DHNS integrated with RotatE on the three datasets. The ablated models can be classified from two aspects:

- To verify each contribution of our developed NS strategy, the ablated models are constructed by removing the whole DiffHEG module (w/o DiffHEG) or two sub-modules namely multimodal conditioning (w/o MMC) and multiple hardness-level denoising (w/o MHLD).
- Specific to our training strategy, the ablated models are designed by removing the whole NTAT module (w/o NTAT) or the key sub-module namely the hardness-adaptive loss (w/o HAL) while maintaining the hardness-aware weights of generated negative triples.

From the results shown in Table 5, we could observe that the performance of each ablated model drops compared with the whole framework DHNS across all the datasets, illustrating the effectiveness of each contribution of our framework on MMKGC tasks. Particularly, removing DiffHEG leads to the most significant drops in performance across all datasets. For instance, the performance of the ablated model w/o DiffHEG decreases **11.2%/22.9%/10.2%** as to MRR on DB15K/MKG-W/MKG-Y. This indicates that our proposed DiffHEG module for generating negative triples plays a crucial role in improving the model's ability on MMKGC tasks. Besides, we could demonstrate that both multimodal

Table 5. Ablation study of our framework on DB15K, MKG-W, and MKG-Y datasets. The best results are **bold** and the lowest values are labeled with the superscript*.

Ablated Models	DB15K			MKG-W			MKG-Y		
	MRR	H1	H3	MRR	H1	H3	MRR	H1	H3
DHNS	**34.36**	**23.34**	**41.56**	**35.68**	**28.98**	**38.68**	**39.11**	**34.70**	**41.23**
w/o DiffHEG	30.91*	20.96*	35.85*	29.02*	23.10*	31.77*	35.50*	30.32*	36.86*
w/o MMC	32.82	22.67	37.79	31.49	26.01	33.76	35.75	33.06	39.02
w/o MHLD	31.67	21.71	37.80	30.88	25.55	32.88	36.26	32.69	39.86
w/o NTAT	32.59	21.29	37.59	30.78	25.56	32.92	36.67	33.68	39.28
w/o HAL	33.78	22.05	38.31	31.22	25.80	33.26	36.77	34.54	39.67

conditioning and multiple hardness level denoising mechanisms are effective for generating diverse and high-quality negative triples from the perspectives of multimodal semantics and multiple hardness levels.

More interestingly, compared with eliminating the NS module DiffHEG, removing the training strategy NTAT exhibits a more slight decrease in performance on MKG-Y while similar significant drops on DB15K and MKG-W. In specific, the performance of the ablated model w/o NTAT drops **5.4%**/**15.9%**/**6.7%** as to MRR on DB15K/MKG-W/MKG-Y. Furthermore, the ablated model w/o HAL shows a similar performance to w/o NTAT, indicating that HAL plays a key role in the training strategy to adaptively select the margin to improve the capability of our model in discriminating positive and negative with various hardness levels. In summary, both our proposed NS and training strategies contribute to the performance improvement of the MMKGC model.

5 Conclusion

In this paper, we propose a novel Diffusion Model-based Hierarchical Negative Sampling framework DHNS for MMKGC tasks. To address the unique challenge of uncontrollable negative sampling, especially in the context of MMKGs, we are the first to develop a Diffusion-based Hierarchical Embedding Generation (DiffHEG) module to directly generate hierarchical embeddings rather than entity sampling to compose negative triples, with multimodal semantics and varying hardness levels determined by the diffusion time steps. Then, to handle the issue of the traditional one-margin-fits-all training scheme, a Negative Triple-Adaptive Training (NTAT) strategy is designed to learn the multimodal joint scoring of the generated negative triples, and further train KGE models with a Hardness-Adaptive Loss (HAL) to improve the discrimination capability concerning the diversity among negative triples. The extensive results on three MMKGC datasets illustrate the effectiveness and superiority of our DHNS framework compared with several state-of-the-art unimodal and multimodal KGC models as well as some typical NS techniques.

References

1. Ahrabian, K., Feizi, A., Salehi, Y., Hamilton, W.L., Bose, A.J.: Structure aware negative sampling in knowledge graphs. In: EMNLP. pp. 6093–6101 (2020)
2. Bao, H., Dong, L., Piao, S., Wei, F.: BEiT: BERT pre-training of image transformers. In: ICLR (2022)
3. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. In: NeurIPS. pp. 2787–2795 (2013)
4. Cai, L., Wang, W.Y.: KBGAN: Adversarial learning for knowledge graph embeddings. In: NAACL. pp. 1470–1480 (2018)
5. Cao, J., Fang, J., Meng, Z., Liang, S.: Knowledge graph embedding: A survey from the perspective of representation spaces. ACM Comput. Surv. **56**(6) (2024)
6. Cao, Z., Xu, Q., Yang, Z., He, Y., Cao, X., Huang, Q.: Otkge: multi-modal knowledge graph embeddings via optimal transport. In: NeurIPS (2024)
7. Chao, L., He, J., Wang, T., Chu, W.: PairRE: Knowledge graph embeddings via paired relation vectors. In: ACL-IJCNLP. pp. 4360–4369 (2021)
8. F. M. Suchanek, G. Kasneci, G.W.: Yago: A core of semantic knowledge. In: Web Conference. pp. 697–706 (2007)
9. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: NeurIPS (2020)
10. Ji, G., He, S., Xu, L., Liu, K., Zhao, J.: Knowledge graph embedding via dynamic mapping matrix. In: ACL-IJCNLP. pp. 687–696 (2015)
11. Lee, J., Chung, C., Lee, H., Jo, S., Whang, J.: VISTA: Visual-textual knowledge graph representation learning. In: Findings of EMNLP. pp. 7314–7328 (2023)
12. Lee, J., Wang, Y., Li, J., Zhang, M.: Multimodal reasoning with multimodal knowledge graph. In: ACL. pp. 10767–10782 (2024)
13. Li, Z., Ji, J., Fu, Z., Ge, Y., Xu, S., Chen, C., Zhang, Y.: Efficient non-sampling knowledge graph embedding. In: Web Conference. p. 1727–1736 (2021)
14. Liang, W., Meo, P.D., Tang, Y., Zhu, J.: A survey of multi-modal knowledge graphs: Technologies and trends. ACM Comput. Surv. **56**(11) (2024)
15. Liu, Y., Li, H., Garcia-Duran, A., Niepert, M., Onoro-Rubio, D., Rosenblum, D.S.: Mmkg: Multi-modal knowledge graphs. In: Hitzler, P., Fernández, M., Janowicz, K., Zaveri, A., Gray, A.J., Lopez, V., Haller, A., Hammar, K. (eds.) The Semantic Web. pp. 459–474 (2019)
16. Long, X., Zhuang, L., Li, A., Wei, J., Li, H., Wang, S.: Kgdm: A diffusion model to capture multiple relation semantics for knowledge graph embedding. In: AAAI (2024)
17. Lu, X., Wang, L., Jiang, Z., He, S., Liu, S.: Mmkrl: A robust embedding approach for multi-modal knowledge graph representation learning. Applied Intelligence **52**, 7480–7497 (2021)
18. Lu, X., Wang, L., Jiang, Z., He, S., Liu, S.: Mmkrl: A robust embedding approach for multi-modal knowledge graph representation learning. Applied Intelligence **52**, 7480 – 7497 (2021)
19. Madushanka, T., Ichise, R.: Negative sampling in knowledge graph representation learning: A review. arXiv preprint arXiv:2402.19195 (2024)
20. Mousselly-Sergieh, H., Botschen, T., Gurevych, I., Roth, S.: A multimodal translation-based approach for knowledge graph representation learning. In: SEM. pp. 225–234 (2018)
21. Nguyen, T.K., Fang, Y.: Diffusion-based negative sampling on graphs for link prediction. In: Web Conference. p. 948–958 (2024)

22. Nickel, M., Tresp, V., Kriegel, H.P.: A three-way model for collective learning on multi-relational data. In: ICML. pp. 809–816 (2011)
23. Reimers, N., Gurevych, I.: Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In: EMNLP-IJCNLP. pp. 3982–3992 (2019)
24. Shang, B., Zhao, Y., Liu, J., Wang, D.: Lafa: Multimodal knowledge graph completion with link aware fusion and aggregation. In: AAAI. pp. 8957–8965 (2024)
25. Sun, Z., Deng, Z.H., Nie, J.Y., Tang, J.: RotatE: Knowledge graph embedding by relational rotation in complex space. In: ICLR (2019)
26. Tang, Y., Huang, J., Wang, G., He, X., Zhou, B.: Orthogonal relation transforms with graph context modeling for knowledge graph embedding. In: ACL. pp. 2713–2722 (2020)
27. Trouillon, T., Welbl, J., Riedel, S., Éric Gaussier, Bouchard, G.: Complex embeddings for simple link prediction. In: ICML. pp. 2071–2080 (2016)
28. Vrandečić, D., Krötzsch, M.: Wikidata: a free collaborative knowledgebase. Commun. ACM **57**(10), 78–85 (2014)
29. Wang, M., Wang, S., Yang, H., Zhang, Z., Chen, X., Qi, G.: Is visual context really helpful for knowledge graph? a representation learning perspective. In: ACM MM. pp. 2735–2743 (2021)
30. Wang, P., Li, S., Pan, R.: Incorporating gan for negative sampling in knowledge representation learning. In: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (2018)
31. Wang, Z., Zhang, J., Feng, J., Chen, Z.: Knowledge graph embedding by translating on hyperplanes. In: AAAI. pp. 1112–1119 (2014)
32. Wang, Z., Li, L., Li, Q., Zeng, D.: Multimodal data enhanced representation learning for knowledge graphs. In: IJCNN. pp. 1–8 (2019)
33. Xie, R., Liu, Z., Luan, H., Sun, M.: Image-embodied knowledge representation learning. In: IJCAI. pp. 3140–3146 (2017)
34. Xu, D., Xu, T., Wu, S., Zhou, J., Chen, E.: Relation-enhanced negative sampling for multimodal knowledge graph completion. In: ACM MM. p. 3857–3866 (2022)
35. Yang, B., Yih, W., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases. In: ICLR (2015)
36. Zhang, S., Tay, Y., Yao, L., Liu, Q.: Quaternion knowledge graph embeddings. In: NeurIPS. pp. 2731–2741 (2019)
37. Zhang, Y., Chen, M., Zhang, W.: Modality-aware negative sampling for multimodal knowledge graph embedding. In: IJCNN. pp. 1–8 (2023)
38. Zhang, Y., Chen, Z., Liang, L., Chen, H., Zhang, W.: Unleashing the power of imbalanced modality information for multi-modal knowledge graph completion. In: LREC-COLING. pp. 17120–17130 (2024)
39. Zhang, Y., Zhang, W.: Knowledge graph completion with pre-trained multimodal transformer and twins negative sampling. ArXiv (2022)
40. Zhang, Y., Yao, Q., Shao, Y., Chen, L.: Nscaching: Simple and efficient negative sampling for knowledge graph embedding. In: ICDE. pp. 614–625 (2019)
41. Zhu, X., Li, Z., Wang, X., Jiang, X., Sun, P., Wang, X., Xiao, Y., Yuan, N.J.: Multi-modal knowledge graph construction and application: A survey. IEEE TKDE **36**(2), 715–735 (2024)