

Article

RAPC: Relation Answer Space Prototype Calibration for Multimodal Knowledge Graph Completion

Shuhan Wang ^{1,2,†}, Aizihairijiang Yusufu ^{1,3,†}, Jiang Liu ³ , Chao Ma ^{1,2}, Abidan Ainiwaer ⁴ and Aizierguli Yusufu ^{1,2,*}

¹ School of Computer Science, Xinjiang Normal University, Urumqi 830054, China; hang8318a@gmail.com (S.W.); azhar520abida@gmail.com (A.Y.); machaochao66@gmail.com (C.M.)

² Xinjiang Engineering Technology Research Center for Smart Education and Application, Urumqi 830054, China

³ School of Cyber Science and Engineering, Wuhan University, Wuhan 430072, China; wugangliujiang@gmail.com

⁴ Journalism & Communication College, Xinjiang University, Urumqi 830046, China; abidan@xju.edu.cn

* Correspondence: 107622013010043@xjnu.edu.cn

† These authors contributed equally to this work.

Abstract

Multimodal knowledge graph completion (MMKGC) aims to predict missing entities by exploiting structural, visual, and textual information and is important for semantic retrieval, recommendation, and intelligent question answering. Existing relation-aware methods usually use relational context to adjust modality weights or fuse multimodal scores but rarely exploit the historical answer distribution of each directed relation as explicit ranking evidence. To address this limitation, we propose RAPC, a Relation Answer Space Prototype Calibration framework. For each query, RAPC obtains multimodal prior scores from structural, visual, textual, and image–text cross-modal branches, retrieves the top- k query-relevant anchors from the historical answer space of the corresponding directed relation, and aggregates them into prototype evidence. This evidence is injected through a prior-preserving selective calibration mechanism, while relation-aware hard negative training improves discrimination between true answers and similar false candidates. Experiments on DB15K, MKG-W, and MKG-Y show that RAPC achieves clear gains on DB15K and MKG-W and obtains the best MRR and Hits@1 on MKG-Y. Compared with the best external results, RAPC improves MRR and Hits@1 by 2.64 and 3.57 percentage points on DB15K and by 3.02 and 3.72 percentage points on MKG-W. These results show that relation-level historical answer distributions provide useful explicit evidence for top-ranked entity prediction.

Keywords: multimodal knowledge graph completion; relation answer space; answer anchors; prototype calibration; selective calibration; hard negative mining



Academic Editor: Christos Bouras

Received: 14 June 2026

Revised: 4 July 2026

Accepted: 7 July 2026

Published: 10 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

Knowledge graphs (KGs) usually represent entities and their relations in the form of triples (h, r, t) and have been widely used in semantic retrieval, intelligent question answering, recommender systems, and other tasks [1]. However, conventional KGs mainly rely on structured relations and are limited in capturing the visual and textual semantics associated with entities. Multimodal knowledge graphs (MMKGs) further incorporate external information such as images and text, providing richer semantic sources for entity

representation [2]. Since missing entities are common in real-world graphs, multimodal knowledge graph completion (MMKGC) has become an important task for improving graph completeness. Given a query $(h, r, ?)$ or $(?, r, t)$, MMKGC aims to predict the missing entity [3].

Accurate MMKGC is important not only for improving the completeness of multimodal knowledge graphs but also for supporting practical applications that require reliable entity ranking. For example, in semantic retrieval, recommendation, and intelligent question answering, a system often needs to rank a large number of candidate entities and return a small set of reliable top-ranked answers. Therefore, improving top-rank precision is practically meaningful, especially when structural triples alone are insufficient and multimodal information is needed to distinguish visually or semantically similar entities. From a theoretical perspective, this task also raises the question of how relation-level historical evidence in the training graph can be used together with multimodal representations for more reliable candidate ranking.

Existing MMKGC methods mainly focus on multimodal representation learning and modality fusion. Early methods used visual or textual features as supplements to structural embeddings, while later studies explored adaptive fusion, cross-modal alignment, modality interaction, and relation-aware modeling. In these methods, relational context is often used to adjust modality weights, select modality experts, or fuse prediction scores. However, most of them focus on modality utilization under the current relation and pay less attention to the historical answer distribution encoded by the relation in the training graph. For each directed relation, its historical answer entities form a relation-specific answer space, which indicates where plausible answers usually appear. Thus, relations can provide not only modality guidance but also answer region-level prior evidence for candidate ranking.

To address this problem, we propose RAPC, a Relation Answer Space Prototype Calibration framework. RAPC treats the historical answer entities of a directed relation as a searchable answer space and selects answer anchors that are consistent with the multimodal prior ranking of the current query. The resulting prototype evidence provides relation-specific answer region information for candidate ranking. To avoid disturbing the original multimodal ranking, RAPC uses a prior-preserving selective calibration strategy to inject prototype information only into key local candidate regions. Relation-aware hard negative training is also introduced to distinguish similar false candidates.

The main contributions of this paper are summarized as follows:

1. We propose a relation answer space-driven view for multimodal knowledge graph completion. Different from existing relation-aware MMKGC methods that mainly use relations to guide modality fusion or score aggregation, we show that each directed relation also encodes a historical answer distribution in the training graph. RAPC explicitly organizes such historical answer entities as a relation answer space, which provides answer region-level evidence for candidate ranking.
2. We design a query-relevant prototype calibration mechanism to exploit the relation answer space. Instead of directly using the whole answer space as a static prior or a candidate constraint, RAPC retrieves the top- k answer anchors according to the multimodal prior ranking of the current query and aggregates them into query-specific prototype evidence. A prior-preserving selective calibration strategy is then used to inject this evidence into local ranking regions without fully replacing the original multimodal ranking.
3. Experiments are conducted on three public MMKGC benchmark datasets, namely DB15K, MKG-W, and MKG-Y. These datasets contain structured triples together with entity-associated visual and textual features and are commonly used to evaluate models that combine structural, visual, and textual information. The results show

that RAPC achieves strong performance on DB15K and MKG-W. On MKG-Y, RAPC achieves the best MRR and Hits@1, while its advantage is mainly reflected in top-rank precision. Further analyses examine simple answer space priors, anchor selection, seen and unseen answers, relation frequency, and calibration behavior.

Conceptually, RAPC reflects a data-centric rather than a purely model-centric view of MMKGC. Instead of repeatedly upgrading the modality fusion architecture while keeping the use of the training graph unchanged, RAPC extracts a higher-quality, relation-conditioned slice of the available training data, namely the historical relation answer space, and uses it as explicit ranking evidence. This view complements model-centric improvements and is consistent with the argument that both data-centric and model-centric strategies are important for building compact and robust AI solutions [4].

2. Related Work

2.1. Knowledge Graph Completion

Knowledge graph completion (KGC) aims to infer missing facts from observed triples. Existing methods usually map entities and relations into continuous vector spaces and use scoring functions to evaluate candidate triples [5]. Conventional KGC methods include distance- or transformation-based models, such as TransE [6], TransD [7], RotatE [8], and PairRE [9], as well as semantic matching-based models, such as DistMult [10], ComplEx [11], and TuckER [12]. These methods effectively exploit graph structure and are commonly used as structural baselines in MMKGC. However, they mainly rely on triples and cannot fully use entity-associated visual and textual information.

2.2. Multimodal Knowledge Graph Completion

MMKGC introduces entity-associated modalities, such as images and text, into structured triples. Early methods, including IKRL [3], TBKGC [13], TransAE [14], and RSME [15], mainly used visual or textual information as supplements to structural embeddings. Recent surveys have summarized the development of multimodal knowledge graphs and MMKGC from the perspectives of multimodal construction, representation learning, completion tasks, benchmarks, and application scenarios [16–18].

Recent studies further explore modality fusion, relation-aware modeling, missing-modality robustness, and training strategy optimization. For example, MKGformer introduces a hybrid transformer with multi-level visual–textual fusion [19]; VISTA learns visual–textual knowledge graph representations with stronger cross-modal interaction [20]; LAFA uses link-aware fusion and aggregation to reduce the influence of irrelevant visual information [21]; NativE and MACO address modality diversity, imbalance, and missing-modality problems in more realistic MMKG settings [22,23]; and SNAG improves robust multimodal representation learning by using noise-powered training [24]. In addition, several works focus on training strategies, such as MMRNS [25] and diffusion-based hierarchical negative sampling [26], while generative perspectives, such as DiffusionCom, have also been explored for MMKGC [27]. AdaMF-MAT addresses modality imbalance through adaptive fusion and modality adversarial training [28]; MyGO improves fine-grained multimodal entity representation [29]; and MoMoK uses relation-guided modality experts [30].

Although these methods improve MMKGC from different perspectives, most of them still focus on representation learning, modality fusion, or training strategy design. In contrast, RAPC explicitly models the historical answer distribution of each directed relation as a searchable relation answer space and uses it as query-conditioned ranking evidence.

2.3. Difference from Relation Priors and Candidate Constraints

Although RAPC uses the historical answer entities of each directed relation, it is different from relation-specific candidate priors, type-constrained ranking, and conventional prototype-based reranking. First, RAPC does not use the relation answer space as a hard candidate mask. All entities in the candidate set are still scored and ranked, and entities outside the relation answer space are not removed during inference. Therefore, the relation answer space is used as an evidence source rather than a candidate constraint. Second, RAPC is not a simple relation-specific prior. A static prior or bias assigns similar preference to entities that appear in the historical answer space of a relation. In contrast, RAPC first obtains a multimodal prior ranking for the current query and then dynamically retrieves query-relevant anchors from the relation answer space. Thus, different queries under the same directed relation may select different anchors and construct different prototype evidence. Third, RAPC is different from conventional prototype-based classification or reranking. It does not directly replace the multimodal score with a prototype score. Instead, the prototype score is used as complementary evidence and is injected through a prior-preserving selective calibration mechanism. This design keeps the basic ranking structure produced by multimodal representations while allowing reliable local regions to absorb relation answer space evidence. In summary, RAPC uses the relation answer space as query-conditioned ranking evidence. Its key idea is not to constrain the candidate set but to dynamically retrieve answer region prototypes and selectively calibrate the multimodal prior ranking. The empirical comparisons in Table 1 further distinguish RAPC from simple relation priors, hard answer space constraints, and static prototype reranking.

Table 1. Sanity checks against simple relation answer space priors. All metric values are reported in percentage form. Bold values indicate the best results.

Setting	DB15K		MKG-W		MKG-Y	
	MRR	Hits@1	MRR	Hits@1	MRR	Hits@1
Prior score reference	37.37	30.42	35.09	29.84	35.64	32.19
Prior + relation answer space bias	36.98	29.94	35.32	29.92	35.86	32.30
Hard answer space constraint	29.28	23.52	26.52	22.51	27.70	25.00
Static relation answer space mean prototype	36.43	30.32	33.49	28.22	37.33	33.84
Full RAPC	42.21	35.95	39.12	34.10	38.89	35.81

3. Problem Definition

Let $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T}, \mathcal{M})$ be a multimodal knowledge graph, where $\mathcal{E}$ denotes the entity set, $\mathcal{R}$ denotes the relation set, $\mathcal{T} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ denotes the set of observed triples, and $\mathcal{M}$ denotes the set of modalities associated with entities. For an entity $e \in \mathcal{E}$, its pre-extracted feature under modality $m \in \mathcal{M}$ is denoted as x_e^m . This study considers four types of information: structural, visual, textual, and image–text cross-modal information.

The goal of multimodal knowledge graph completion is to predict the missing entity from the candidate entity set $\mathcal{E}$ given a query $(h, r, ?)$ or $(?, r, t)$. To handle head entity prediction and tail entity prediction in a unified manner, we introduce an inverse relation r^{-1} for each original relation $r \in \mathcal{R}$ and convert head entity prediction $(?, r, t)$ into the tail prediction form $(t, r^{-1}, ?)$. Therefore, all queries can be uniformly represented as $q = (u, \bar{r}, ?)$, where $u \in \mathcal{E}$ is the known entity and $\bar{r} \in \mathcal{R} \cup \mathcal{R}^{-1}$ is a directed relation. The model computes a query-specific score $s_q(e)$ for each candidate entity $e \in \mathcal{E}$ so that the true answer y is ranked as high as possible among all candidates.

Based on this formulation, we define the relation answer space $A_{\bar{r}}$ for each directed relation $\bar{r}$. For tail entity prediction, the answer space consists of all entities that appear as tail entities of the relation in the training set. For head entity prediction, the answer space consists of all entities that appear as head entities of the relation in the training set:

$$A_{\bar{r}} = \begin{cases} \{t \mid \exists h \in \mathcal{E}, (h, r, t) \in \mathcal{T}_{\text{train}}\}, & \bar{r} = r, \\ \{h \mid \exists t \in \mathcal{E}, (h, r, t) \in \mathcal{T}_{\text{train}}\}, & \bar{r} = r^{-1} \end{cases} \quad (1)$$

It should be noted that $A_{\bar{r}}$ is not the set of true answers for the current query but a candidate region formed by the historical answer entities of the directed relation in the training graph. This paper studies how to jointly exploit multimodal entity representations and the relation answer space $A_{\bar{r}}$ to learn the final scoring function $s_{\text{final}}(q, e)$ so that the true answer entity obtains a higher rank in the candidate entity set.

4. Method

This section introduces the proposed Relation Answer Space Prototype Calibration framework, namely RAPC. Given a query $q = (u, \bar{r}, ?)$, where u denotes the known entity and $\bar{r}$ denotes a directed relation, RAPC first obtains a multimodal prior ranking based on structural, visual, textual, and image–text cross-modal information. It then selects query-relevant answer anchors from the answer space of the current directed relation and constructs prototype evidence. Finally, a prior-preserving selective calibration mechanism is used to generate the final ranking. The overall framework is shown in Figure 1.

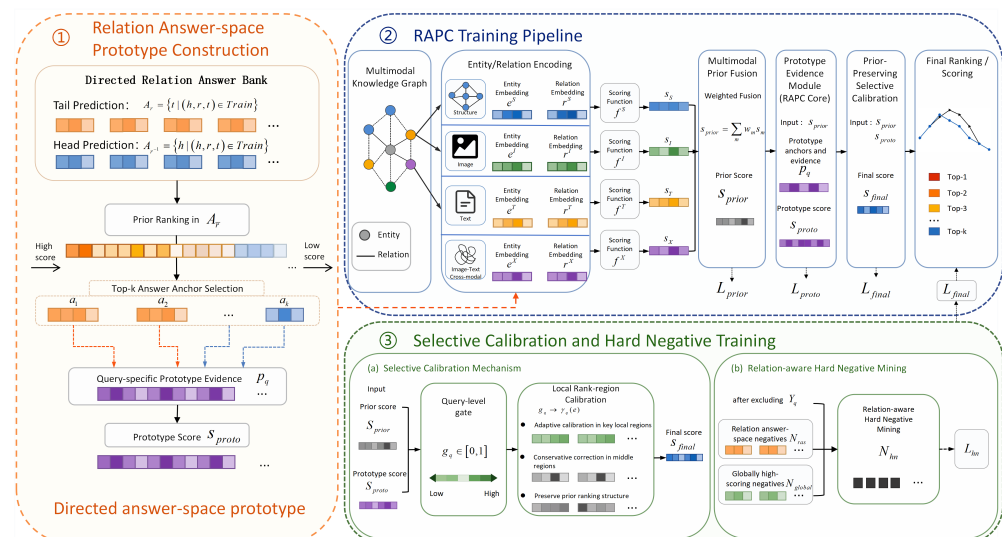


Figure 1. Overall framework of RAPC. The model first obtains a multimodal prior ranking, then selects query-relevant answer anchors from the directed relation answer space to construct prototype evidence, and finally injects this evidence into the final candidate ranking through a selective calibration mechanism. Colors are used to distinguish different modules and processing stages, while the ellipses indicate omitted intermediate candidates rather than missing content.

4.1. Multimodal Prior Ranking

For each entity $e \in \mathcal{E}$, RAPC constructs entity representations from four branches, namely the structural, visual, textual, and image–text cross-modal branches:

$$e^S, e^I, e^T, e^X \quad (2)$$

where S , I , T , and X denote the structural branch, visual branch, textual branch, and image–text cross-modal branch (X branch), respectively. The structural branch is obtained from learnable entity embeddings. The visual and textual branches are obtained by mapping pre-extracted features through projection layers. The image–text cross-modal branch is

used to model the joint semantics between visual and textual representations. RAPC also learns relation representations for each directed relation $\bar{r}$ in the four branches:

$$r_{\bar{r}}^S, \quad r_{\bar{r}}^I, \quad r_{\bar{r}}^T, \quad r_{\bar{r}}^X \quad (3)$$

Given a query $q = (u, \bar{r}, ?)$, the model computes the matching score of each candidate entity e in each branch. Following previous complex-valued embedding methods [8,11], we adopt a complex-valued scoring form. For branch $m \in \{S, I, T, X\}$, the entity and relation representations are written as

$$e^m = (e_{\text{re}}^m, e_{\text{im}}^m), \quad r_{\bar{r}}^m = (r_{\text{re}}^m, r_{\text{im}}^m) \quad (4)$$

The candidate score of the m -th branch is defined as

$$s_m(q, e) = \langle u_{\text{re}}^m r_{\text{re}}^m - u_{\text{im}}^m r_{\text{im}}^m, e_{\text{re}}^m \rangle + \langle u_{\text{re}}^m r_{\text{im}}^m + u_{\text{im}}^m r_{\text{re}}^m, e_{\text{im}}^m \rangle \quad (5)$$

RAPC then fuses the scores from the four branches with weighted summation to obtain the multimodal prior score of the current query:

$$s_{\text{prior}}(q, e) = \sum_{m \in \{S, I, T, X\}} \omega_m s_m(q, e) \quad (6)$$

where ω_m denotes the fusion weight of the m -th branch. $s_{\text{prior}}(q, e)$ represents the basic ranking result obtained only from multimodal representations. The relation answer space evidence introduced later is calibrated based on this prior ranking.

4.2. Query-Relevant Relation Answer Space Prototype

Based on the directed relation answer space defined above, RAPC pre-constructs an answer entity bank $A_{\bar{r}}$ for each directed relation $\bar{r}$. Instead of directly using the mean representation of the entire answer space, RAPC dynamically selects query-relevant answer anchors from $A_{\bar{r}}$ according to the multimodal prior ranking of the current query, thereby constructing a query-specific relation answer space prototype.

Given a query $q = (u, \bar{r}, ?)$, RAPC first uses the multimodal prior score $s_{\text{prior}}(q, e)$ to select the top- k entities from the current relation answer space $A_{\bar{r}}$ as answer anchors:

$$B_q = \text{TopK}_{a \in A_{\bar{r}}} s_{\text{prior}}(q, a) \quad (7)$$

where $B_q = \{a_1, a_2, \dots, a_k\}$ denotes the anchor set corresponding to the current query. These anchors come from the historical answer region of the current directed relation and are also consistent with the multimodal prior ranking of the current query. Therefore, they can provide query-relevant candidate region evidence.

To distinguish the importance of different anchors, RAPC computes normalized weights according to their prior scores:

$$\alpha_i = \frac{\exp(s_{\text{prior}}(q, a_i) / \tau)}{\sum_{j=1}^k \exp(s_{\text{prior}}(q, a_j) / \tau)} \quad (8)$$

where τ is the temperature coefficient.

For each modality branch $m \in \{S, I, T, X\}$, the selected answer anchors are aggregated into a query-relevant prototype:

$$p_q^m = \sum_{i=1}^k \alpha_i a_i^m \quad (9)$$

where a_i^m denotes the representation of anchor entity a_i in the m -th branch.

After obtaining the prototype representation, RAPC combines the known entity, relation, and query-relevant prototype to form a prototype-enhanced query representation:

$$z_q^m = \phi_m([u^m; r_{\bar{r}}^m; p_q^m]) \quad (10)$$

where $[\cdot; \cdot]$ denotes vector concatenation, and $\phi_m(\cdot)$ is the nonlinear mapping function of the m -th branch. Based on this representation, the prototype branch score of candidate entity e is defined as

$$s_{\text{proto}}^m(q, e) = \langle z_q^m, e^m \rangle \quad (11)$$

Finally, the prototype scores from different branches are fused through relation-conditioned weights to obtain the overall prototype evidence score:

$$s_{\text{proto}}(q, e) = \sum_{m \in \{S, I, T, X\}} \pi_m(\bar{r}) s_{\text{proto}}^m(q, e) \quad (12)$$

where $\pi_m(\bar{r})$ denotes the prototype evidence weight of the m -th branch under the current directed relation. Through the above process, RAPC transforms the historical answer distribution in the relation answer space into prototype evidence that can be used for the current query, providing complementary information for subsequent ranking calibration.

Score Normalization

Since s_{prior} and s_{proto} are produced by different scoring paths, their numerical scales may differ. To reduce scale mismatch before calibration, we apply query-wise score normalization to each score vector over all candidate entities:

$$\hat{s}(q, e) = \frac{s(q, e) - \mu_q}{\sigma_q + \varepsilon} \quad (13)$$

where μ_q and σ_q denote the mean and standard deviation of the candidate scores for query q , and ε is a small constant for numerical stability. In implementation, branch scores are normalized before score fusion and then rescaled by a validation-selected factor. This operation makes the prior and prototype scores more comparable and stabilizes the residual term $s_{\text{proto}}(q, e) - s_{\text{prior}}(q, e)$ used in selective calibration.

4.3. Prior-Preserving Selective Calibration

The relation answer space prototype provides complementary evidence for candidate entity ranking. However, its reliability can be affected by the size of the relation answer space, the quality of selected anchors, and the candidate distribution. If $s_{\text{proto}}(q, e)$ is directly used to replace $s_{\text{prior}}(q, e)$, the relatively stable multimodal prior ranking may be disrupted. Therefore, RAPC adopts a prior-preserving selective relation calibration strategy, denoted as SRC, which injects prototype evidence while preserving the structure of the prior ranking.

In candidate ranking, a learned scoring function may assign high scores to candidates that are semantically plausible but factually incorrect. This failure mode is the ranking-level counterpart of plausible-but-incorrect outputs observed in probabilistic AI models more generally [31]. In MMKGC, such candidates often appear in local ranking regions where entities have similar structural or multimodal scores. SRC is therefore designed to control how prototype evidence is injected into these local regions so that useful relation answer space evidence can correct the ranking without aggressively disturbing the multimodal prior.

First, RAPC estimates the reliability of prototype evidence at the query level and obtains a calibration coefficient $g_q \in [0, 1]$. The basic calibrated score is defined as

$$s_{\text{calib}}(q, e) = s_{\text{prior}}(q, e) + g_q(s_{\text{proto}}(q, e) - s_{\text{prior}}(q, e)) \quad (14)$$

For reproducibility, we provide the functional form of the query-level gate g_q . For each query q , RAPC constructs a 10-dimensional feature vector

$$\mathbf{x}_q = [b_q, H_q, a_q^{\max}, a_q^{\text{margin}}, \sigma_q^{\text{prior}}, \sigma_q^{\text{proto}}, \mu_q^{\Delta}, \sigma_q^{\Delta}, o_q, \Delta_q^{\text{top}}].$$

Here, $b_q = \log(1 + |A_f|) / \log(1 + |\mathcal{E}|)$ denotes the normalized size of the current directed relation answer space. Let B_q be the selected anchor set, and let α_i be the normalized weight of anchor a_i . The anchor weight entropy is computed as

$$H_q = -\frac{1}{\log |B_q|} \sum_{a_i \in B_q} \alpha_i \log(\alpha_i + \epsilon),$$

where ϵ is a small numerical constant. The features $a_q^{\max}$ and a_q^{margin} denote the largest anchor weight and the difference between the largest anchor weight and the mean anchor weight, respectively.

For score statistics and top-overlap calculation, RAPC uses the top- K_g candidates, where $K_g = 32$ in implementation. Let $T_q^{\text{prior}} = \text{Top}_{K_g}(s_{\text{prior}})$ and $T_q^{\text{proto}} = \text{Top}_{K_g}(s_{\text{proto}})$. The terms σ_q^{prior} and σ_q^{proto} are the standard deviations of the prior and prototype scores over T_q^{prior} and T_q^{proto} , respectively. The residual statistics μ_q^{Δ} and σ_q^{Δ} are computed over T_q^{prior} using $\Delta(q, e) = s_{\text{proto}}(q, e) - s_{\text{prior}}(q, e)$. The overlap feature is defined as

$$o_q = \frac{|T_q^{\text{prior}} \cap T_q^{\text{proto}}|}{K_g},$$

and Δ_q^{top} denotes the difference between the highest prototype score and the highest prior score.

The gate network is a two-layer MLP:

$$\mathbf{h}_q = \text{Dropout}(\text{GELU}(W_1 \text{LN}(\mathbf{x}_q) + \mathbf{b}_1)),$$

$$\delta_q = \tanh(W_2 \mathbf{h}_q + b_2),$$

where LN denotes LayerNorm. The hidden dimension is 16 in the reported experiments. Let η_t denote the prototype warm-up factor at epoch t :

$$\eta_t = \begin{cases} 0, & t < t_0, \\ \min\left(1, \frac{t-t_0+1}{T_w}\right), & t \geq t_0, \end{cases}$$

where t_0 is the prototype start epoch and T_w is the number of prototype warm-up epochs. The final query-level gate is computed as

$$g_q = \text{clip}(\eta_t(g_0 + \rho \delta_q), 0, 1),$$

where g_0 is the base gate strength and ρ controls the adaptive gate range. The parameters of the gate network are optimized jointly with the other trainable parameters of RAPC under the overall training objective. The resulting g_q is expanded to all candidates of the current query and then adjusted by the candidate-level SRC rule below.

RAPC then performs selective correction only within local candidate regions. The motivation is that different ranking regions have different reliability. Top-ranked candidates usually receive stronger structural and multimodal support and therefore can receive prototype-based calibration with a relatively larger strength. In contrast, middle-ranking candidates are less stable because their prior scores are often close to each other; a small prototype-induced perturbation may change their relative order and push similar false candidates upward. Therefore, RAPC applies conservative correction to this region. For candidates jointly supported by the prior ranking, prototype ranking, and relation answer space, the prototype evidence is regarded as more reliable, and a larger calibration strength is allowed. Let $R_{\text{top}} = \text{Top}_{n_b}(s_{\text{prior}})$ denote the high-confidence candidate set in the multimodal prior ranking, and let $R_{\text{mid}} = \text{Top}_{n_d}(s_{\text{prior}}) \setminus R_{\text{top}}$ denote the easily disturbed middle-ranking candidate set, where $n_d > n_b$. Meanwhile, the candidate set jointly supported by the prior ranking, prototype ranking, and relation answer space is defined as

$$C_{\text{support}} = \text{Top}_{n_p}(s_{\text{prior}}) \cap \text{Top}_{n_{\text{proto}}}(s_{\text{proto}}) \cap A_{\bar{r}} \quad (15)$$

For each candidate entity e , RAPC constructs a candidate-level calibration coefficient:

$$\gamma_q(e) = \text{clip}(g_q + \beta_b \mathbb{I}(e \in R_{\text{top}}) - \beta_d \mathbb{I}(e \in R_{\text{mid}}) + \beta_s \mathbb{I}(e \in C_{\text{support}}), \gamma_{\min}, \gamma_{\max}) \quad (16)$$

where β_b , β_d , and β_s control the calibration strength for prior high-confidence candidates, the conservative suppression strength for middle-ranking candidates, and the calibration strength for jointly supported candidates, respectively. $\gamma_{\min}$ and $\gamma_{\max}$ are used to limit the calibration range. The final score is defined as

$$s_{\text{final}}(q, e) = s_{\text{prior}}(q, e) + \gamma_q(e)(s_{\text{proto}}(q, e) - s_{\text{prior}}(q, e)) \quad (17)$$

In this process, candidates in R_{top} retain relatively high calibration strength, candidates in R_{mid} receive more conservative correction, and candidates in C_{support} receive a larger calibration weight because they are jointly supported by the prior ranking, prototype ranking, and relation answer space. Therefore, SRC does not perform unconstrained reranking over all candidates. Instead, it controls the injection strength of prototype evidence within selected local ranking regions.

4.4. Training Objective

The training objective of RAPC consists of the final ranking loss, the multimodal prior auxiliary loss, the prototype evidence auxiliary loss, the relation-aware hard negative loss, and a regularization term. For a query q , let Y_q denote the set of true answers. We adopt the multi-positive cross-entropy loss for the final score, prior score, and prototype score. Taking the final score as an example, the loss is defined as

$$\mathcal{L}_{\text{final}} = -\log \frac{\sum_{y \in Y_q} \exp(s_{\text{final}}(q, y))}{\sum_{e \in \mathcal{E}} \exp(s_{\text{final}}(q, e))} \quad (18)$$

$\mathcal{L}_{\text{prior}}$ and $\mathcal{L}_{\text{proto}}$ are computed in the same form based on s_{prior} and s_{proto} , respectively. They are used to maintain the stability of the multimodal prior ranking and enhance the independent discriminative ability of the prototype branch.

To improve the model's ability to distinguish similar false candidates, RAPC introduces relation-aware hard negatives. For a query $q = (u, \bar{r}, ?)$, during training, the true answer set Y_q is first excluded, and hard negatives are then mined according to the current final score s_{final} . The hard negatives consist of relation answer space negatives and global high-scoring negatives:

$$N_{\text{hn}}(q) = N_{\text{ras}}(q) \cup N_{\text{global}}(q) \quad (19)$$

where

$$N_{\text{ras}}(q) = \text{Top}_{k_r} \{s_{\text{final}}(q, e) \mid e \in A_{\bar{r}}, e \notin Y_q\} \quad (20)$$

$$N_{\text{global}}(q) = \text{Top}_{k_g} \{s_{\text{final}}(q, e) \mid e \in \mathcal{E} \setminus Y_q\} \quad (21)$$

In the experiments, we adopt a hybrid mining strategy. High-scoring negative samples are selected from both the relation answer space and the global candidate set, and duplicated candidates are removed. These candidates are close to the true answers either within the relation answer region or in the final ranking, making them more suitable than random negatives for training the model's local discriminative ability.

Based on the above hard negatives, the relation-aware hard negative loss is defined as

$$\mathcal{L}_{\text{hn}} = -\log \frac{\sum_{y \in Y_q} \exp(s_{\text{final}}(q, y) / \tau_{\text{hn}})}{\sum_{y \in Y_q} \exp(s_{\text{final}}(q, y) / \tau_{\text{hn}}) + \sum_{e^- \in N_{\text{hn}}(q)} \exp(s_{\text{final}}(q, e^-) / \tau_{\text{hn}})} \quad (22)$$

where τ_{hn} is the temperature coefficient. This objective encourages true answers to be ranked higher than relation-related hard negatives in the final ranking.

Finally, the overall optimization objective of RAPC is

$$\mathcal{L} = \mathcal{L}_{\text{final}} + \lambda_1 \mathcal{L}_{\text{prior}} + \lambda_2 \mathcal{L}_{\text{proto}} + \lambda_3 \mathcal{L}_{\text{hn}} + \lambda_4 \mathcal{L}_{\text{reg}} \quad (23)$$

where $\mathcal{L}_{\text{reg}}$ is the regularization term, and $\lambda_1, \lambda_2, \lambda_3, \lambda_4$ are the corresponding loss weights.

By jointly optimizing the above objectives, RAPC can improve the effectiveness of relation answer space prototype evidence while preserving the stability of the multimodal prior ranking and further enhance the model's ability to distinguish hard candidates.

5. Experiments

To evaluate the effectiveness of RAPC, we conduct experiments on three public multimodal knowledge graph completion datasets, namely DB15K, MKG-W, and MKG-Y. The experiments are designed to answer the following questions: whether RAPC outperforms existing structure-only unimodal and multimodal KGC methods, whether relation answer space evidence and query-relevant anchors bring the main performance gains, and whether selective calibration and relation-aware hard negatives can further improve the ranking results.

5.1. Experimental Settings

5.1.1. Datasets

We use DB15K, MKG-W, and MKG-Y as the experimental datasets. DB15K is derived from DBpedia [32], while MKG-W and MKG-Y are derived from Wikidata [33] and YAGO [34], respectively. All three datasets contain structured triples as well as entity-associated visual and textual features and are commonly used public benchmark datasets for MMKGC.

As shown in Table 2, the three datasets differ in both relation scale and split size. DB15K contains more relation types, while MKG-Y contains fewer relations, which is further related to the broader relation answer spaces reported in Table 3.

Table 2. Statistics of the experimental datasets.

Dataset	Source	#Ent.	#Rel.	Train	Valid	Test
DB15K	DBpedia	12,842	279	79,222	9902	9904
MKG-W	Wikidata	15,000	169	34,196	4276	4274
MKG-Y	YAGO	15,000	28	21,310	2665	2663

Table 3. Statistics of directed relation answer spaces. Coverage denotes the proportion of test answers covered by the training relation answer space. Entropy denotes the average normalized entropy of relation-specific answer distributions, normalized by $\log |A_{\bar{r}}|$.

Dataset	#Rel.	#Dir. Rel.	Avg. $ A_{\bar{r}} $	Median $ A_{\bar{r}} $	Max $ A_{\bar{r}} $	Coverage	Entropy
DB15K	279	558	134.60	13	10,365	65.03%	0.815
MKG-W	169	338	108.26	8	3689	56.62%	0.735
MKG-Y	28	56	406.91	46	3551	67.86%	0.932

5.1.2. Baselines

We compare RAPC with three categories of methods. The first category includes structure-only unimodal KGC methods, including TransE [6], TransD [7], DistMult [10], ComplEx [11], RotatE [8], PairRE [9], and GC-OTE [35]. The second category includes classical multimodal KGC methods, including IKRL [3], TBKGC [13], TransAE [14], MMKRL [36], RSME [15], VBKGC [37], and OTKGE [38]. The third category includes recent strong MMKGC methods, including AdaMF [28], MMRNS [25], MyGO [29], and MoMoK [30]. These methods cover major technical directions such as structural modeling, modality fusion, relation-aware modeling, and negative sampling enhancement. We only include baselines whose results are publicly reported or reproducible under the same benchmark datasets and filtered evaluation protocol. Therefore, the comparison focuses on methods with directly comparable results rather than on methods whose complete results under the same setting are unavailable.

For external baselines, we report the results available in the original publications or official released results under the same benchmark splits and filtered evaluation protocol, unless otherwise stated. These baselines were not uniformly re-implemented within the RAPC codebase, and they were not forced to use RAPC's ComplEx-style scoring backbone, inverse relation augmentation, embedding dimension, or hyperparameter schedule. Therefore, Table 4 should be interpreted as a benchmark-level comparison under the same dataset splits and filtered evaluation protocol rather than as a fully controlled architectural ablation.

Table 4. Main results of RAPC and comparable baselines on DB15K, MKG-W, and MKG-Y. Bold values indicate the best results, and underlined values indicate the second-best results.

Model		DB15K				MKG-W				MKG-Y			
		MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10	MRR	Hits@1	Hits@3	Hits@10
Unimodal KGC	TransE	24.86	12.78	31.48	47.07	29.19	21.06	33.20	44.23	30.73	23.45	35.18	43.37
	TransD	21.52	8.34	29.93	44.24	25.56	15.88	32.99	40.18	26.39	17.01	33.60	40.31
	DistMult	23.03	14.78	26.28	39.59	20.99	15.93	22.28	30.86	25.04	19.33	27.80	35.95
	ComplEx	27.48	18.37	31.57	45.37	24.93	19.09	26.69	36.73	28.71	22.26	32.12	40.93
	RotatE	29.28	17.87	36.12	49.66	33.67	26.80	36.68	46.73	34.95	29.10	38.35	45.30
	PairRE	31.13	21.62	35.52	49.30	34.40	28.24	36.71	46.04	32.01	25.53	35.84	43.89
	GC-OTE	31.85	22.11	36.52	51.18	33.92	26.55	35.96	46.05	32.95	26.77	36.44	44.08
Multimodal KGC	IKRL	26.82	14.09	34.93	49.09	32.36	26.11	34.75	44.07	33.22	30.37	34.28	38.26
	TBKGC	28.40	15.61	37.03	49.86	31.48	25.31	33.98	43.24	33.99	30.47	35.27	40.07
	TransAE	28.09	21.25	31.17	41.17	30.00	21.23	34.91	44.72	28.10	25.31	29.10	33.03
	MMKRL	26.81	13.85	35.07	49.39	30.10	22.16	34.09	44.69	36.81	31.66	39.79	45.31
	RSME	29.76	24.15	32.12	40.29	29.23	23.36	31.97	40.43	34.44	31.78	36.07	39.09
	VBKGC	30.61	19.75	37.18	49.44	30.61	24.91	33.01	40.88	37.04	33.76	38.75	42.30
	OTKGE	23.86	18.45	25.89	34.23	34.36	28.85	36.25	44.88	35.51	31.97	37.18	41.38
	MMRNS	32.68	23.01	37.86	51.01	35.03	28.59	37.49	47.47	35.93	30.53	39.07	45.47
	AdaMF	32.51	21.31	39.67	51.68	34.27	27.21	37.86	47.21	38.06	33.49	40.44	45.48
	MyGO	37.72	30.08	41.26	52.21	36.10	29.78	38.54	47.75	38.44	35.01	39.84	44.19
	MoMoK	39.57	32.38	43.45	54.14	35.89	30.38	37.54	46.13	37.91	35.09	39.20	43.20
Ours	RAPC	42.21	35.95	45.14	54.32	39.12	34.10	40.63	49.24	38.89	35.81	39.69	44.99

To isolate the contribution of the proposed relation answer space mechanism under controlled conditions, we additionally report the internal prior score reference in Table 1. This reference uses the same RAPC backbone, multimodal inputs, inverse relation augmentation, embedding dimension, training protocol, and evaluation protocol as the full

RAPC model but removes relation answer space evidence, query-relevant anchor retrieval, prototype aggregation, and selective calibration.

5.1.3. Overfitting Control

To reduce overfitting, RAPC uses several regularization and validation-based strategies. First, dropout and L2 weight decay are applied to the projection, prototype mapping, and scoring modules. Second, all main hyperparameters, including fusion weights, SRC parameters, and training schedule settings, are selected using the validation set. Third, the best checkpoint is selected according to validation MRR, and the test set is used only for final evaluation, not for model selection or hyperparameter tuning. Fourth, learning-rate warm-up and cosine decay are used to stabilize optimization. In addition, we report results over multiple random seeds in Table 5 to characterize the stability of the final model rather than to claim that multi-seed stability itself eliminates overfitting.

Table 5. Stability analysis over multiple random seeds. All metric values are reported in percentage form as mean $\pm$ standard deviation.

Dataset	MRR	Hits@1	Hits@3	Hits@10
DB15K	41.80 $\pm$ 0.41	35.49 $\pm$ 0.42	44.74 $\pm$ 0.35	53.96 $\pm$ 0.31
MKG-W	38.70 $\pm$ 0.42	33.59 $\pm$ 0.39	40.10 $\pm$ 0.47	48.72 $\pm$ 0.46
MKG-Y	38.60 $\pm$ 0.29	35.83 $\pm$ 0.12	39.57 $\pm$ 0.35	44.57 $\pm$ 0.49

5.1.4. Evaluation Metrics and Implementation Details

We evaluate all methods under the filtered setting and report MRR, Hits@1, Hits@3, and Hits@10. In filtered evaluation, for each query, other entities that form known true triples with the same query in the training, validation, or test set are removed from the candidate ranking, except for the target answer being evaluated. This prevents other correct entities from being incorrectly treated as negative candidates.

Let $\mathcal{Q}$ denote the set of evaluation queries, and let rank_q denote the filtered rank of the true answer for query q . MRR is defined as

$$\text{MRR} = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \frac{1}{\text{rank}_q}. \quad (24)$$

Hits@ N measures the proportion of queries whose true answer is ranked within the top N positions:

$$\text{Hits@}N = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathbb{I}(\text{rank}_q \leq N), \quad (25)$$

where $N \in \{1, 3, 10\}$. MRR is the mean reciprocal rank of the correct entity. It assigns greater weight to correct answers ranked near the top and is therefore sensitive to improvements in top-ranked predictions. Hits@1 measures exact top-ranked prediction accuracy, whereas Hits@3 and Hits@10 measure whether the correct entity appears within the top three and top ten candidates, respectively. All reported metric values are multiplied by 100 and shown in percentage form.

RAPC is implemented in PyTorch 2.1.0 with structural, visual, textual, and image–text cross-modal branches. Visual and textual features are pre-extracted features provided by the datasets and projected into a unified space. For each relation, we build both forward and inverse relation answer spaces to unify head and tail prediction. Unless otherwise specified, the number of answer anchors is set to $k = 8$. The best checkpoint is selected on the validation set and evaluated on the test set. For relation-aware hard negatives, we construct a candidate pool from both the relation answer space and global candidates and select up to 32 hard negatives according to s_{final} after removing duplicates.

These settings are used for all reported RAPC results unless otherwise specified. The number of answer anchors is set to $k = 8$ by default because it provides a balance between prototype representativeness and noise control. A very small k may fail to cover the local answer region of the current directed relation, whereas a very large k may introduce weakly related historical answers and reduce prototype discriminability. This choice is further supported by the top- k sensitivity analysis in Figure 2, where $k = 8$ gives stable and competitive performance across the evaluated settings. The candidate region parameters n_b , n_d , n_p , and n_{proto} are used to define local ranking regions for selective calibration rather than to remove candidates from the ranking list. The SRC parameters and fusion weights are selected on the validation set and then kept fixed during test evaluation. The main implementation details and hyperparameters are summarized in Table 6.

Table 6. Implementation details and main hyperparameters of RAPC.

Item	Setting
Branches	Structural, visual, textual, and image–text cross-modal branches
Embedding dimension	256 for DB15K and MKG-W, and 300 for MKG-Y
Entity/relation scoring	Complex-valued scoring function following ComplEx-style matching
Visual/textual features	Pre-extracted features provided by the datasets; no additional external supervision is used
Projection layer	Gated projection layer with LayerNorm, GELU, dropout, and linear transformation
Cross-modal branch	Concatenation of projected visual and textual representations followed by a gated projection layer
Prototype mapping ϕ_m	Two-layer MLP with LayerNorm, GELU, dropout, and linear output layer
Prior fusion weights ω_m	Branch weights selected on the validation set and kept fixed during test evaluation
Prototype fusion weights $\pi_m(\bar{r})$	Relation-conditioned weights generated from the directed relation representation and normalized by softmax
Number of anchors	$k = 8$ by default
Candidate regions	$n_b = 10$, $n_d = 50$, $n_p = 50$, and $n_{\text{proto}} = 10$
Query gate g_q	Computed from answer space size, anchor weight entropy and concentration, prior/prototype score statistics, and top-ranked candidate overlap
SRC parameters	$\beta_b = 0.14$, $\beta_d = 0.03$, $\beta_s = 0.12$, $\gamma_{\min} = 0.25$, $\gamma_{\max} = 0.58$
Hard negatives	Up to 32 negatives mined from relation answer space and global high-scoring candidates
Score normalization	Query-wise normalization with validation-selected scaling factor 1.5
Optimizer and learning rates	AdamW; structural branch 2.0×10^{-4} , modal and prototype branches 2.4×10^{-4}
Batch size and regularization	Training batch size 256, evaluation batch size 128, dropout 0.18, weight decay 1×10^{-5}
Training schedule	Warm-up with cosine learning-rate decay; best checkpoints selected on the validation set
Evaluated checkpoints	Epoch 180 on DB15K, epoch 85 on MKG-W, and epoch 170 on MKG-Y
Hardware	NVIDIA GeForce RTX 3090 GPU (NVIDIA Corporation, Santa Clara, CA, USA)

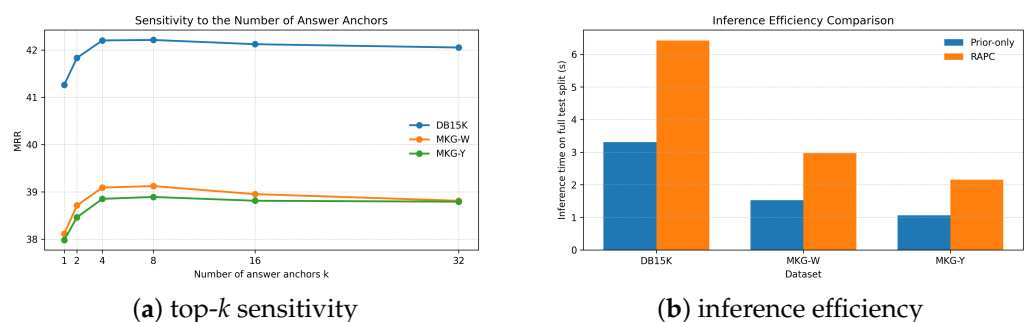


Figure 2. Parameter sensitivity and inference efficiency analysis of RAPC on DB15K, MKG-W, and MKG-Y. (a) shows the MRR results under different numbers of answer anchors k , and (b) compares the inference time of the prior-only model and RAPC on the full test set.

5.2. Main Results

Table 4 presents the main comparison results. For clarity, the strongest comparable baseline is selected separately for each dataset and each metric as the best external result reported in Table 4 under the same benchmark setting. On DB15K, MoMoK gives the

strongest previous MRR and Hits@1, so RAPC improves MRR and Hits@1 by 2.64 and 3.57 percentage points, respectively. On MKG-W, MyGO gives the strongest previous MRR, while MoMoK gives the strongest previous Hits@1; therefore, RAPC improves MRR and Hits@1 by 3.02 and 3.72 percentage points, respectively. These gains suggest that relation answer space evidence is most useful when the historical answer region is sufficiently informative for ranking.

The results on MKG-Y show a different pattern. RAPC obtains the best MRR and Hits@1, whereas AdaMF gives slightly higher Hits@3 and Hits@10. This pattern is consistent with the answer space statistics reported in Table 3: MKG-Y has broader and more diffuse relation answer spaces, which can make the selected anchors less discriminative. Therefore, RAPC mainly improves top-rank precision, while its advantage becomes weaker for wider ranking ranges on this dataset.

5.2.1. Stability over Random Seeds

To further examine the stability of RAPC, we retrain the model with multiple random seeds and report the mean and standard deviation on the test set. The results are shown in Table 5. The mean performance is close to the single-run results in Table 4, and the standard deviations are small on all three datasets, showing that RAPC is stable under different random initializations.

5.2.2. Significance Test

Since most published baselines only provide aggregate results and do not release query-level ranks under the same setting, paired significance tests against all external baselines are not available. We therefore conduct a bootstrap significance test between RAPC and the internal prior baseline using query-level reciprocal ranks. For each dataset, we resample test queries 1000 times and compute the MRR difference between RAPC and the prior baseline. The improvement of RAPC over the prior baseline is significant on all three datasets, with $p < 0.001$.

5.2.3. Sanity Checks Against Simple Relation Answer Space Priors

To examine whether RAPC can be reduced to simple relation answer space priors, we conduct several sanity check baselines in Table 1. These baselines do not remove modules from the full RAPC training pipeline. Instead, they test simple ways of using the relation answer space, including using only the prior score, adding a constant answer space bias, applying a hard answer space constraint, and using a static relation-level prototype. The prior score reference is included only to show the ranking performance before any answer space operation.

The results in Table 1 show that simple uses of the relation answer space cannot explain the gains of RAPC. Adding a constant relation answer space bias brings little or even negative improvement over the prior score reference, and the hard answer space constraint performs much worse because true answers may appear outside the historical answer space. The static relation answer space mean prototype is also clearly inferior to full RAPC, showing that a relation-level prototype alone is not sufficient. These results indicate that RAPC does not rely on a simple relation answer space prior, a hard answer space constraint, or a static prototype. Instead, its gains come from query-relevant anchor retrieval and selective calibration.

5.2.4. Anchor Selection Diagnostics

To further examine whether the gains come from query-relevant relation answer space anchors, we compare the default relation top- k anchor selection with two diagnostic variants. Global top- k anchors select anchors from the whole entity set according to the

prior score, while random RAS anchors randomly sample anchors from the directed relation answer space with uniform weights.

As shown in Table 7, random RAS anchors perform clearly worse than relation top- k anchors on all three datasets. This result shows that simply sampling historical answers from the directed relation answer space is not sufficient. Global top- k anchors obtain competitive results, but they are still consistently lower than relation top- k anchors. This suggests that high-prior entities alone cannot fully replace relation-aware anchor retrieval. Overall, the results support that RAPC benefits from selecting query-relevant anchors within the directed relation answer space.

Table 7. Anchor selection diagnostics under the same trained checkpoint. MRR values are reported in percentage form. Bold values indicate the best results.

Setting	DB15K	MKG-W	MKG-Y
Relation top- k anchors	42.21	39.12	38.89
Global top- k anchors	42.12	38.86	38.80
Random RAS anchors	40.55 $\pm$ 0.02	37.04 $\pm$ 0.01	38.25 $\pm$ 0.01

5.3. Ablation Study

To analyze the role of each core component in RAPC, we design four ablation settings.

w/o RAS Evidence: This setting removes relation answer space evidence and does not select answer anchors from the directed relation answer space. It is used to verify whether the relation answer space serves as the core evidence source of RAPC.

w/o Query-Specific Anchors: This setting is a trained RAPC ablation. It keeps the relation answer space, the RAPC training objectives, and the selective calibration module but replaces query-specific top- k anchors with a relation-level mean prototype. It is different from the static mean prototype sanity check in Table 1 because this variant is trained and evaluated within the full RAPC framework.

w/o SRC: This setting keeps the prototype evidence but removes selective relation-aware calibration, using only ordinary query-level calibration. It is used to verify whether prior-preserving selective calibration helps stably absorb prototype evidence.

w/o Relation-Aware Hard Negatives: This setting removes the relation-aware hard negative training objective. It is used to verify whether hard negatives mined from the relation answer space and the global candidate set can enhance the discriminative ability of the model during training. RAHN denotes relation-aware hard negatives. The corresponding ablation results are summarized in Table 8.

Table 8. Ablation results of RAPC on the three datasets. Bold values indicate the best results.

Setting	DB15K		MKG-W		MKG-Y	
	MRR	Hits@1	MRR	Hits@1	MRR	Hits@1
w/o RAS Evidence	37.37	30.42	35.09	29.84	35.64	32.19
w/o Query-Specific Anchors	40.76	33.74	37.12	31.40	38.02	35.19
w/o SRC	41.95	35.59	38.78	33.68	38.62	35.36
w/o RAHN	42.10	35.89	38.88	33.97	38.07	34.91
Full RAPC	42.21	35.95	39.12	34.10	38.89	35.81

The w/o RAS Evidence result is identical to the prior score reference in Table 1 and is repeated here to provide a component-level ablation comparison.

Removing RAS Evidence leads to the largest performance drop, reducing MRR by 4.84, 4.03, and 3.25 points on DB15K, MKG-W, and MKG-Y, respectively. This confirms that the main source of improvement is the relation answer space rather than the multimodal prior alone. Replacing query-specific anchors with a relation-level mean prototype also

hurts performance, especially on MKG-W, where MRR drops by 2.00 points. Therefore, the model benefits not only from using historical answer entities but also from selecting anchors conditioned on the current query. Removing SRC and RAHN causes smaller but consistent degradation, suggesting that these two modules mainly stabilize calibration and improve local discrimination among hard candidates.

5.3.1. Fine-Grained Analysis of SRC

We further perform a fine-grained analysis of SRC by removing each local calibration component during inference. The results are shown in Table 9.

Table 9. Fine-grained ablation results of selective relation calibration. All metric values are reported in percentage form.

Variant	DB15K		MKG-W		MKG-Y	
	MRR	Hits@1	MRR	Hits@1	MRR	Hits@1
Full SRC	42.21	35.95	39.12	34.10	38.89	35.81
w/o R_{top}	42.05	35.77	39.06	34.04	38.80	35.72
w/o R_{mid} suppression	42.19	35.93	39.08	34.07	38.92	35.87
w/o $C_{support}$	42.18	35.94	39.09	34.14	38.71	35.53
fixed g_q	42.20	35.94	38.89	33.89	38.94	35.87
w/o clip	42.20	35.94	39.11	34.19	38.89	35.83

Overall, these results suggest that SRC should be interpreted as a bounded calibration module rather than an aggressive reranking strategy. Some local rules have dataset-dependent effects, but the complete SRC design provides a stable trade-off between absorbing prototype evidence and preserving the multimodal prior ranking.

5.3.2. Score and Calibration Distribution

To examine whether prototype calibration causes unstable score shifts, we visualize the distributions of s_{prior} , s_{proto} , s_{final} , and the calibration coefficient. As shown in Figure 3, the prototype score generally has a wider distribution, while the final score remains close to the prior score after selective calibration. The calibration coefficient is also bounded within a moderate range across the three datasets. This indicates that RAPC injects prototype evidence conservatively rather than replacing the multimodal prior ranking.

To further analyze the working mechanism of RAPC, we conduct additional analyses from seven perspectives: seen-answer and unseen-answer generalization, relation answer space statistics, answer space size, relation frequency, the number of answer anchors, hyperparameter sensitivity, and a case study.

5.3.3. Seen-Answer and Unseen-Answer Analysis

Since RAPC uses the historical answer entities of each directed relation, we further analyze whether its improvement simply comes from memorizing answers that have appeared in the training graph. We divide test queries into seen-answer and unseen-answer groups according to whether the target entity belongs to the training relation answer space A_r .

As shown in Table 10, RAPC obtains larger gains on seen-answer queries than on unseen-answer queries. This is expected because seen answers are covered by the historical answer region and can receive more direct prototype support. More importantly, RAPC still improves over the multimodal prior on unseen-answer queries across all three datasets, showing that it does not work as a hard answer space memorization mechanism. Instead, the relation answer space provides complementary evidence for calibration, while candidates outside A_r can still be ranked by the multimodal prior. These results also reveal a

limitation of RAPC: its advantage is stronger when the relation answer space has partial but meaningful coverage and becomes weaker when the true answer is outside the historical answer region.

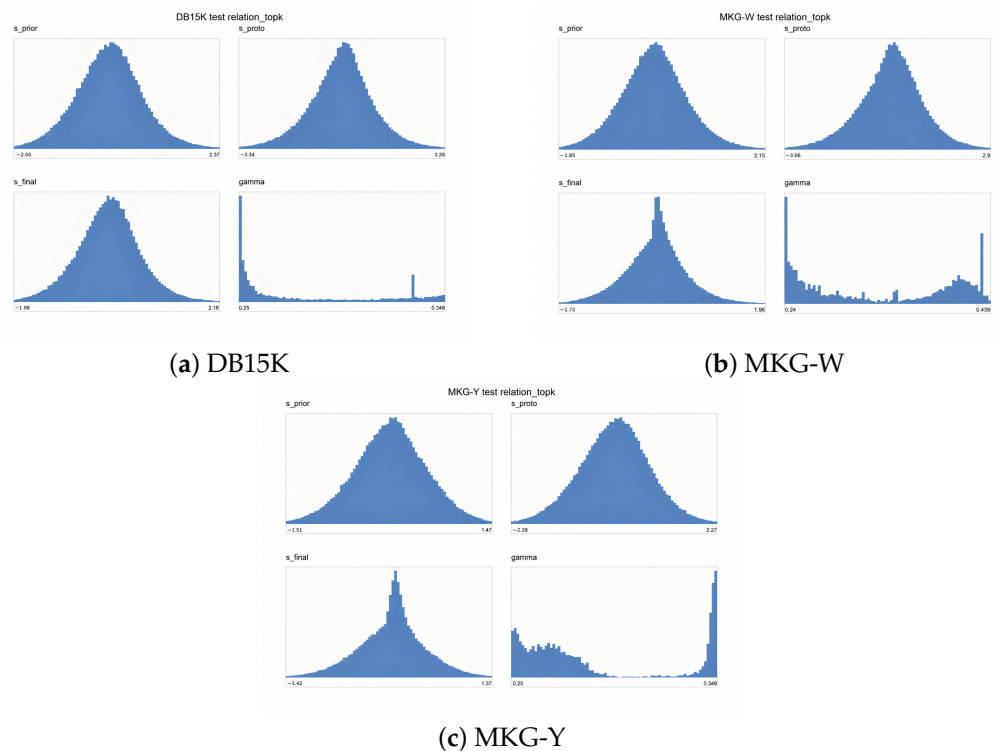


Figure 3. Score and calibration distribution analysis on DB15K, MKG-W, and MKG-Y.

Table 10. Seen-answer and unseen-answer analysis. All metric values are reported in percentage form, and gains are reported in percentage points.

Dataset	Group	<i>n</i>	Prior MRR	RAPC MRR	Gain	Prior Hits@1	RAPC Hits@1	Gain
DB15K	Seen	12,879	41.90	48.73	+6.83	34.02	41.94	+7.92
DB15K	Unseen	6925	28.95	30.07	+1.12	23.73	24.81	+1.08
MKG-W	Seen	4840	43.75	50.34	+6.59	37.95	45.06	+7.11
MKG-W	Unseen	3708	23.79	24.47	+0.68	19.26	19.79	+0.53
MKG-Y	Seen	3614	37.78	41.95	+4.16	33.73	38.38	+4.65
MKG-Y	Unseen	1712	31.12	32.43	+1.31	28.94	30.38	+1.44

5.3.4. Statistics of Relation Answer Spaces

To better understand the dataset-level behavior of RAPC, we summarize the statistics of directed relation answer spaces in Table 3.

For each directed relation $\bar{r}$, we compute the normalized entropy of its empirical answer distribution as

$$H(\bar{r}) = -\frac{1}{\log |A_{\bar{r}}|} \sum_{e \in A_{\bar{r}}} p_{\bar{r}}(e) \log p_{\bar{r}}(e), \quad (26)$$

where $p_{\bar{r}}(e)$ denotes the empirical frequency of entity e appearing as an answer of directed relation $\bar{r}$ in the training set. When $|A_{\bar{r}}| \leq 1$, the entropy is set to 0. The entropy reported in Table 3 is averaged over the directed relations in each dataset.

As shown in Table 3, MKG-Y contains far fewer relations than DB15K and MKG-W but has the largest average and median answer space sizes. It also has the highest normalized answer space entropy, indicating that its relation-specific answer distributions are more diffuse. This helps explain why RAPC obtains smaller gains on some MKG-Y metrics:

when the answer space is broad and less concentrated, selected anchors may contain more weakly related historical answers and provide less discriminative prototype evidence.

5.3.5. Analysis of Relation Answer Space Size

We divide test queries into four quantile groups according to the size of the directed relation answer space $|A_{\bar{r}}|$ and report the MRR improvement in the final RAPC score over the multimodal prior score. The results are shown in Table 11.

Table 11. MRR improvement of RAPC over the multimodal prior ranking under different relation answer space sizes. Values are reported in percentage points, and the number of queries in each group is shown in parentheses.

Dataset	q0–q25	q25–q50	q50–q75	q75–q100
DB15K	+10.39 ($n = 5005$)	+5.40 ($n = 4914$)	+2.76 ($n = 5217$)	+0.60 ($n = 4668$)
MKG-W	+7.66 ($n = 2188$)	+6.27 ($n = 2235$)	+1.28 ($n = 2217$)	+0.12 ($n = 1908$)
MKG-Y	+3.27 ($n = 1599$)	+2.74 ($n = 1151$)	+5.25 ($n = 1447$)	+1.11 ($n = 1129$)

As shown in Table 11, RAPC gains more on small and medium answer spaces, while the improvement becomes smaller in the largest groups. For example, the MRR gain drops from 10.39 to 0.60 points on DB15K and from 7.66 to 0.12 points on MKG-W. This suggests that relation answer spaces provide useful evidence, but overly broad answer spaces may contain more weakly related anchors and reduce prototype discriminability. The positive but non-monotonic trend on MKG-Y also supports the need for selective calibration.

5.3.6. Long-Tail Relation Frequency Analysis

We analyze RAPC under different relation frequency groups to examine its behavior on long-tail and frequent relations. Test queries are divided into four quantile groups according to the training frequency of their directed relations. The results are shown in Table 12.

Table 12. Long-tail relation frequency analysis. MRR values are reported in percentage form, and gains are reported in percentage points.

Dataset	Freq. Group	n	Mean Train Freq.	Prior MRR	RAPC MRR	Gain
DB15K	q0–q25	4951	312.9	42.68	48.61	+5.94
DB15K	q25–q50	4951	1305.0	50.54	54.34	+3.79
DB15K	q50–q75	4951	3061.6	30.49	33.78	+3.29
DB15K	q75–q100	4951	8285.0	25.77	32.10	+6.32
MKG-W	q0–q25	2137	288.0	41.82	48.63	+6.81
MKG-W	q25–q50	2137	923.7	38.43	42.13	+3.70
MKG-W	q50–q75	2137	1790.7	28.44	32.55	+4.11
MKG-W	q75–q100	2137	5815.2	31.68	33.15	+1.47
MKG-Y	q0–q25	1332	765.2	28.45	31.31	+2.86
MKG-Y	q25–q50	1332	2646.4	62.72	68.90	+6.18
MKG-Y	q50–q75	1331	3423.5	45.72	48.46	+2.74
MKG-Y	q75–q100	1331	6131.0	5.64	6.86	+1.22

As shown in Table 12, RAPC consistently improves over the multimodal prior across different relation frequency groups. This indicates that relation answer space evidence is not limited to a specific frequency range. The gain becomes smaller in the highest-frequency group of MKG-Y, which is consistent with the observation that overly broad answer spaces may introduce weakly related anchors and reduce prototype discriminability. This also suggests that relation frequency alone does not determine prediction difficulty; high-frequency relations may still be difficult when their answer spaces are broad and semantically diffuse.

5.3.7. Analysis of the Number of Answer Anchors and Inference Efficiency

The number of answer anchors k controls how many historical answers are used to construct prototype evidence. We compare different k values and evaluate inference overhead against the prior-only model. Timing is measured on an NVIDIA GeForce RTX 3090 GPU, including forward scoring and filtered ranking on the full test set, excluding model loading. Each result is averaged over three runs.

As shown in Figure 2a, the model performs relatively poorly when $k = 1$ on most datasets, indicating that a single historical answer anchor is insufficient to represent the potential answer region under the current relation. As k increases to 4 or 8, the model performance improves and reaches the best result when $k = 8$. When k further increases to 16 or 32, the performance slightly decreases, suggesting that too many anchors may introduce weakly related historical answer entities. Therefore, we set the default number of answer anchors to $k = 8$.

As shown in Figure 2b, the inference time of RAPC is about 1.94 to 2.03 times that of the prior-only model. The additional overhead mainly comes from answer anchor selection, prototype construction, and selective calibration, but the full test set evaluation still remains on the order of seconds. In terms of complexity, answer anchor selection has a complexity of approximately $O(|A_r| \log k)$, prototype aggregation has a complexity of $O(kd)$, and final candidate scoring remains $O(|\mathcal{E}|d)$. Therefore, RAPC improves ranking quality while keeping the additional computation within an acceptable range.

5.3.8. Hyperparameter Sensitivity

We further evaluate the sensitivity of RAPC to key calibration hyperparameters, including $\gamma_{\max}$, β_d , and the anchor temperature τ . The results are shown in Table 13.

Table 13. Hyperparameter sensitivity analysis on DB15K and MKG-W. All metric values are reported in percentage form.

Hyperparameter	Value	DB15K MRR	DB15K Hits@1	MKG-W MRR	MKG-W Hits@1
$\gamma_{\max}$	0.40	42.19	35.93	38.94	33.90
	0.50	42.22	35.95	38.99	33.94
	0.58	42.21	35.95	39.12	34.10
	0.70	42.20	35.94	39.11	34.11
	0.90	42.20	35.94	39.11	34.11
β_d	0.00	42.19	35.93	39.08	34.07
	0.01	42.21	35.95	39.09	34.07
	0.03	42.21	35.95	39.12	34.10
	0.05	42.20	35.94	39.13	34.14
	0.08	42.17	35.90	39.14	34.16
τ	0.50	41.85	35.43	38.97	33.96
	0.75	42.09	35.79	39.09	34.07
	1.00	42.21	35.95	39.12	34.10
	1.50	42.42	36.25	39.06	33.98
	2.00	42.53	36.41	38.98	33.86

As shown in Table 13, RAPC is relatively stable under different calibration and anchor temperature settings. The results for $\gamma_{\max}$ and β_d show only small fluctuations, suggesting that the selective calibration module is not highly sensitive to these parameters. The anchor temperature τ has a moderate influence, and different datasets may favor slightly different temperatures. Although a few individual test metrics are slightly higher under non-default values, all hyperparameters are selected according to validation performance and kept fixed during test evaluation.

5.3.9. Case Study

To provide a more intuitive understanding of how RAPC changes candidate ranking, we present three representative cases in Table 14. The table reports whether the target answer is covered by the training relation answer space, the size of the answer space, and the rank of the target entity under the prior, prototype, and final scores.

Table 14. Case study of candidate ranking changes. A smaller rank indicates a better position of the target entity.

Case	Dataset	Seen	$ A_r $	Prior Rank	Prototype Rank	Final Rank
Large RAPC gain	DB15K	Yes	10	11,530	1	4
Unseen-answer	MKG-W	No	42	13,197	1	16
Broad space with stable ranking	MKG-Y	Yes	3551	1	1	1

As shown in Table 14, RAPC can substantially improve the target rank when the selected prototype evidence is highly consistent with the true answer region. In the large-gain case from DB15K, the prior ranking places the target entity at a low position, while the prototype evidence ranks it first, and the final score promotes it to the top-ranked region. For the unseen-answer case from MKG-W, although the target entity is not included in the training relation answer space, RAPC still improves its rank from 13197 to 16, showing that the method does not rely on hard answer space masking. The broad space case from MKG-Y shows that even under a large relation answer space, RAPC can preserve the top-ranked result when the multimodal prior and prototype evidence are already consistent. These examples illustrate that RAPC uses relation answer space evidence as complementary ranking evidence while preserving the multimodal prior when it is already confident.

5.4. Limitations

Although RAPC improves candidate ranking by using relation answer space evidence, it still has several limitations. First, its effectiveness depends on the quality and coverage of the relation answer space. When the answer space is overly broad, sparse, or weakly concentrated, the selected anchors may contain weakly related historical answers and provide less discriminative prototype evidence. Second, RAPC obtains larger gains on seen-answer queries than on unseen-answer queries because seen answers are directly covered by the training relation answer space. Although unseen-answer queries can still be ranked by the multimodal prior, the corrective effect of prototype evidence is weaker in this case. Third, the anchor retrieval, prototype construction, and selective calibration modules introduce additional inference overhead compared with the prior-only model. Finally, the current framework is mainly evaluated in a transductive MMKGC setting, and its extension to inductive or dynamically evolving multimodal knowledge graphs remains an important direction for future work.

6. Conclusions

This paper has presented RAPC, a Relation Answer Space Prototype Calibration framework for multimodal knowledge graph completion. RAPC uses the historical answer entities of each directed relation as a relation answer space, retrieves query-relevant anchors, and injects the resulting prototype evidence through prior-preserving selective calibration. Experiments on DB15K and MKG-W show clear gains over strong MMKGC baselines. On MKG-Y, RAPC achieves the best top-rank metrics, namely MRR and Hits@1, although its advantage becomes weaker for wider ranking ranges.

Theoretically, this study shows that relation-level historical answer distributions can be explicitly modeled as ranking evidence, complementing existing MMKGC methods that mainly focus on multimodal representation learning and modality fusion. Practically, RAPC

can benefit applications such as semantic retrieval and recommendation, where relation-conditioned answer space evidence may help promote reliable candidates and suppress semantically plausible-but-incorrect alternatives. For example, in a recommendation-oriented multimodal knowledge graph, RAPC can use historical answers under relations such as item category, brand, or compatible product to promote relation-consistent candidates and suppress visually or textually similar but relation-inconsistent entities.

Future work will focus on more robust answer space modeling, adaptive anchor selection for broad or long-tail relations, and extension to inductive or dynamically evolving multimodal knowledge graphs.

Author Contributions: A.Y. (Aizihairijiang Yusufu): Conceptualization, Methodology, and Software; J.L.: Formal Analysis, Visualization, and Writing—Reviewing and Editing; S.W.: Data Curation and Writing—Original Draft Preparation; A.A.: Investigation and Resources; C.M.: Validation and Investigation; A.Y. (Aizierguli Yusufu): Supervision. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Xinjiang Uygur Autonomous Region Innovation Environment (Talent and Base) Construction Special Program, Natural Science Program (Special Training for Ethnic Minority Scientific and Technological Talents) (Grant No. 2025D03032); the Undergraduate Education and Teaching Research and Reform Project of Universities in the Xinjiang Uygur Autonomous Region (Grant No. XJGXPTJG-2026077); the Youth Top-notch Talent Support Program of Xinjiang Normal University (Grant No. XJNUQB2022-22); the Tender Project of the Engineering Research Center of Smart Education of Xinjiang Normal University (Grant No. XJNU-ZHJY202403); the National Natural Science Foundation of China (Grant No. 61662081); and the National Social Science Foundation of China (Grant No. 14AZD11).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets analyzed in this study are publicly available benchmark datasets described in Section 5.1. The processed data and experimental outputs are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

KG	Knowledge graph
KGC	Knowledge graph completion
MMKG	Multimodal knowledge graph
MMKGC	Multimodal knowledge graph completion
RAS	Relation answer space
RAPC	Relation answer space prototype calibration
SRC	Selective relation calibration
RAHN	Relation-aware hard negatives

References

1. Hogan, A.; Blomqvist, E.; Cochez, M.; d’Amato, C.; de Melo, G.; Gutierrez, C.; Kirrane, S.; Labra Gayo, J.E.; Navigli, R.; Neumaier, S.; et al. Knowledge Graphs. *ACM Comput. Surv.* **2021**, *54*, 71:1–71:37.
2. Liu, Y.; Li, H.; Garcia-Duran, A.; Niepert, M.; Onoro-Rubio, D.; Rosenblum, D.S. MMKG: Multi-Modal Knowledge Graphs. In Proceedings of the 16th European Semantic Web Conference, Portorož, Slovenia, 2–6 June 2019; pp. 459–474.
3. Xie, R.; Liu, Z.; Luan, H.; Sun, M. Image-Embodied Knowledge Representation Learning. In Proceedings of the 26th International Joint Conference on Artificial Intelligence, Melbourne, Australia, 19–25 August 2017; pp. 3140–3146.

4. Hamid, O.H. Data-Centric and Model-Centric AI: Twin Drivers of Compact and Robust Industry 4.0 Solutions. *Appl. Sci.* **2023**, *13*, 2753. <https://doi.org/10.3390/app13052753>.
5. Wang, Q.; Mao, Z.; Wang, B.; Guo, L. Knowledge Graph Embedding: A Survey of Approaches and Applications. *IEEE Trans. Knowl. Data Eng.* **2017**, *29*, 2724–2743.
6. Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; Yakhnenko, O. Translating Embeddings for Modeling Multi-Relational Data. *Adv. Neural Inf. Process. Syst.* **2013**, *26*, 2787–2795.
7. Ji, G.; He, S.; Xu, L.; Liu, K.; Zhao, J. Knowledge Graph Embedding via Dynamic Mapping Matrix. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, Beijing, China, 26–31 July 2015; pp. 687–696.
8. Sun, Z.; Deng, Z.H.; Nie, J.Y.; Tang, J. RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. In Proceedings of the International Conference on Learning Representations, New Orleans, LA, USA, 6–9 May 2019.
9. Chao, L.; He, J.; Wang, T.; Chu, W. PairRE: Knowledge Graph Embeddings via Paired Relation Vectors. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics, Online, 1–6 August 2021; pp. 4360–4369.
10. Yang, B.; Yih, W.; He, X.; Gao, J.; Deng, L. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In Proceedings of the International Conference on Learning Representations, San Diego, CA, USA, 7–9 May 2015.
11. Trouillon, T.; Welbl, J.; Riedel, S.; Gaussier, E.; Bouchard, G. Complex Embeddings for Simple Link Prediction. In Proceedings of the 33rd International Conference on Machine Learning, New York, NY, USA, 20–22 June 2016; pp. 2071–2080.
12. Balažević, I.; Allen, C.; Hospedales, T.M. TuckER: Tensor Factorization for Knowledge Graph Completion. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Hong Kong, China, 3–7 November 2019; pp. 5185–5194.
13. Mousselly-Sergieh, H.; Botschen, T.; Gurevych, I.; Roth, S. A Multimodal Translation-Based Approach for Knowledge Graph Representation Learning. In Proceedings of the 7th Joint Conference on Lexical and Computational Semantics, New Orleans, LA, USA, 5–6 June 2018; pp. 225–234.
14. Wang, Z.; Li, L.; Li, Q.; Zeng, D. Multimodal Data Enhanced Representation Learning for Knowledge Graphs. In Proceedings of the International Joint Conference on Neural Networks, Budapest, Hungary, 14–19 July 2019; pp. 1–8.
15. 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 Proceedings of the 29th ACM International Conference on Multimedia, Virtual Event, China, 20–24 October 2021; pp. 2735–2743.
16. Liang, W.; De Meo, P.; Tang, Y.; Zhu, J. A Survey of Multi-Modal Knowledge Graphs: Technologies and Trends. *ACM Comput. Surv.* **2024**, *56*, 273:1–273:41.
17. Chen, Z.; Zhang, Y.; Fang, Y.; Geng, Y.; Guo, L.; Chen, X.; Li, Q.; Zhang, W.; Chen, J.; Zhu, Y.; Li, J.; Liu, X.; Pan, J.Z.; Zhang, N.; Chen, H. Knowledge Graphs Meet Multi-Modal Learning: A Comprehensive Survey. *arXiv* **2024**, arXiv:2402.05391.
18. Liu, T. Multi-Modal Knowledge Graph Completion: A Survey. In Proceedings of the International Conference on Image Processing, Machine Learning and Pattern Recognition, Nanjing, China, 6–8 December 2024; pp. 116–121.
19. Chen, X.; Zhang, N.; Li, L.; Deng, S.; Tan, C.; Xu, C.; Huang, F.; Si, L.; Chen, H. Hybrid Transformer with Multi-Level Fusion for Multimodal Knowledge Graph Completion. In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, 11–15 July 2022; pp. 904–915.
20. Lee, J.; Chung, C.; Lee, H.; Jo, S.; Whang, J.J. VISTA: Visual-Textual Knowledge Graph Representation Learning. In Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, 6–10 December 2023; pp. 7314–7328.
21. Shang, B.; Zhao, Y.; Liu, J.; Wang, D. LAFA: Multimodal Knowledge Graph Completion with Link Aware Fusion and Aggregation. *Proc. Aaai Conf. Artif. Intell.* **2024**, *38*, 8957–8965.
22. Zhang, Y.; Chen, Z.; Guo, L.; Xu, Y.; Hu, B.; Liu, Z.; Zhang, W.; Chen, H. NativE: Multi-Modal Knowledge Graph Completion in the Wild. In Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, Washington, DC, USA, 14–18 July 2024; pp. 91–101.
23. Zhang, Y.; Chen, Z.; Zhang, W. MACO: A Modality Adversarial and Contrastive Framework for Modality-Missing Multi-Modal Knowledge Graph Completion. In *Natural Language Processing and Chinese Computing*; Springer: Cham, Switzerland, 2023; pp. 123–134.
24. Chen, Z.; Fang, Y.; Zhang, Y.; Guo, L.; Chen, J.; Pan, J.Z.; Chen, H.; Zhang, W. Noise-powered Multi-modal Knowledge Graph Representation Framework. In Proceedings of the 31st International Conference on Computational Linguistics, Abu Dhabi, United Arab Emirates, 19–24 January 2025; pp. 141–155.
25. Xu, D.; Xu, T.; Wu, S.; Zhou, J.; Chen, E. Relation-Enhanced Negative Sampling for Multimodal Knowledge Graph Completion. In Proceedings of the 30th ACM International Conference on Multimedia, Lisboa, Portugal, 10–14 October 2022; pp. 3857–3866.
26. Niu, G.; Zhang, X. Diffusion-Based Hierarchical Negative Sampling for Multimodal Knowledge Graph Completion. In *Database Systems for Advanced Applications (DASFAA 2025)*; Lecture Notes in Computer Science; Springer: Singapore, 2026; Volume 15988, pp. 479–495. https://doi.org/10.1007/978-981-95-3906-2_34.

27. Huang, W.; Li, P.; Liang, M.; Hou, X.; Li, Y.; Du, J.; Xue, Z.; Guan, Z. DiffusionCom: Structure-Aware Multimodal Diffusion Model for Multimodal Knowledge Graph Completion. In *Proceedings of the 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Barcelona, Spain, 4–8 May 2026; pp. 12452–12456. <https://doi.org/10.1109/ICASSP55912.2026.11463648>.
28. Zhang, Y.; Chen, Z.; Liang, L.; Chen, H.; Zhang, W. Unleashing the Power of Imbalanced Modality Information for Multimodal Knowledge Graph Completion. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation*, Torino, Italy, 20–25 May 2024; pp. 17120–17130.
29. Zhang, Y.; Chen, Z.; Guo, L.; Xu, Y.; Hu, B.; Liu, Z.; Zhang, W.; Chen, H. Tokenization, Fusion, and Augmentation: Towards Fine-Grained Multi-Modal Entity Representation. *Proc. Aaai Conf. Artif. Intell.* **2025**, *39*, 13322–13330.
30. Zhang, Y.; Chen, Z.; Guo, L.; Xu, Y.; Hu, B.; Liu, Z.; Zhang, W.; Chen, H. Multiple Heads Are Better Than One: Mixture of Modality Knowledge Experts for Entity Representation Learning. In *Proceedings of the International Conference on Learning Representations*, Singapore, 24–28 April 2025.
31. Hamid, O.H. Beyond Probabilities: Unveiling the Delicate Dance of Large Language Models (LLMs) and AI-Hallucination. In *Proceedings of the 2024 IEEE Conference on Cognitive and Computational Aspects of Situation Management*, Montreal, QC, Canada, 7–10 May 2024; pp. 85–90. <https://doi.org/10.1109/CogSIMA61085.2024.10553755>.
32. Lehmann, J.; Isele, R.; Jakob, M.; Jentzsch, A.; Kontokostas, D.; Mendes, P.N.; Hellmann, S.; Morsey, M.; Van Kleef, P.; Auer, S.; et al. DBpedia—A Large-Scale, Multilingual Knowledge Base Extracted from Wikipedia. *Semant. Web* **2015**, *6*, 167–195.
33. Vrandečić, D.; Krötzsch, M. Wikidata: A Free Collaborative Knowledgebase. *Commun. ACM* **2014**, *57*, 78–85.
34. Suchanek, F.M.; Kasneci, G.; Weikum, G. YAGO: A Core of Semantic Knowledge. In *Proceedings of the 16th International Conference on World Wide Web*, Banff, AB, Canada, 8–12 May 2007; pp. 697–706.
35. Tang, Y.; Huang, J.; Wang, G.; He, X.; Zhou, B. Orthogonal Relation Transforms with Graph Context Modeling for Knowledge Graph Embedding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online, 5–10 July 2020; pp. 2713–2722.
36. Lu, X.; Wang, L.; Jiang, Z.; He, S.; Liu, S. MMKRL: A Robust Embedding Approach for Multi-Modal Knowledge Graph Representation Learning. *Appl. Intell.* **2022**, *52*, 7480–7497.
37. Zhang, Y.; Zhang, W. Knowledge Graph Completion with Pre-Trained Multimodal Transformer and Twins Negative Sampling. *arXiv* **2022**, arXiv:2209.07084.
38. Cao, Z.; Xu, Q.; Yang, Z.; He, Y.; Cao, X.; Huang, Q. OTKGE: Multi-Modal Knowledge Graph Embeddings via Optimal Transport. In *Advances in Neural Information Processing Systems*, New Orleans, LA, USA, 28 November–9 December 2022; Volume 35.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.