REROUTE: BEYOND STATIC REPRESENTATIONS IN MULTIMODAL KNOWLEDGE GRAPH COMPLETION

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal knowledge graph completion (MMKGC) integrates structural, visual, and textual information to predict missing facts. Static candidate representations use fixed modality combinations, although evidence requirements vary across relations and head/tail roles. Missing modalities further limit the available evidence. We propose ReRoute to adapt entity representations to these requirements. Its relation-role-aware multimodal routing (RRMR) adjusts visual and textual enhancement components on both the query and candidate sides. These adjustments depend on the relation and head/tail role. Structure-aware missing-modality compensation (SMMC) aggregates available modality-specific enhancement components from one-hop neighbors in the training graph. RRMR then modulates these supplementary signals. Decomposable scoring evaluates all entities without explicitly constructing conditioned candidate representations for each query. Across four evaluation metrics on DB15K, MKG-W, and MKG-Y, ReRoute ranks first in 10 of the 12 comparisons and second in the remaining 2. On MKG-W and MKG-Y, ReRoute achieves relative MRR improvements of 5.02% and 6.93%, respectively, over the strongest baseline on each dataset. Our code and data are available at `https://anonymous.4open.science/r/ReRoute-code-47BE/`.

1 INTRODUCTION

Knowledge graphs organize entities and relations as triples for question answering, search, and recommendation, but remain incomplete (Hogan et al., 2021; Nickel et al., 2015). Multimodal knowledge graphs incorporate entity images and textual descriptions, providing complementary evidence for inferring missing facts (Zhu et al., 2024). Multimodal knowledge graph completion (MMKGC) combines structural, visual, and textual information to predict a missing head or tail entity given a known entity and relation (Zhang et al., 2025b; Zhu et al., 2025).

Existing MMKGC methods integrate multimodal information through entity encoding and fusion, then use relation representations to assess triple plausibility (Pezeshkpour et al., 2018). MyGO (Zhang et al., 2025b) organizes multimodal information into fine-grained tokens, and MoMoK (Zhang et al., 2025a) learns relation-aware entity representations via modality knowledge experts. We focus on how candidate representations adapt to relations, head/tail roles, and modality availability during all-entity prediction.

Figure 1 illustrates two challenges in constructing candidate representations. First, the modality evidence relevant to a candidate depends on the relation and its head/tail role. A static candidate representation combines visual and textual information in the same way across queries. For example, Chris Rock is a candidate head entity for "Who acted in *Bad Company*?" and a candidate tail entity for "Who does Eddie Murphy influence?", involving different evidence requirements. Second, increasing a modality's weight cannot recover missing information. One-hop neighbors in the training graph with the corresponding modality available can provide supplementary evidence. Appendix A provides further details on these examples.

We propose ReRoute, combining relation-role-aware multimodal routing (RRMR) with structure-aware missing-modality compensation (SMMC). Fine-grained entity encoding produces reusable multimodal base representations and visual and textual enhancement components. SMMC uses neighborhood evidence to compensate for missing modalities. RRMR then modulates the enhancement components on both the query and candidate sides. Shared candidate-side gates enable de-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

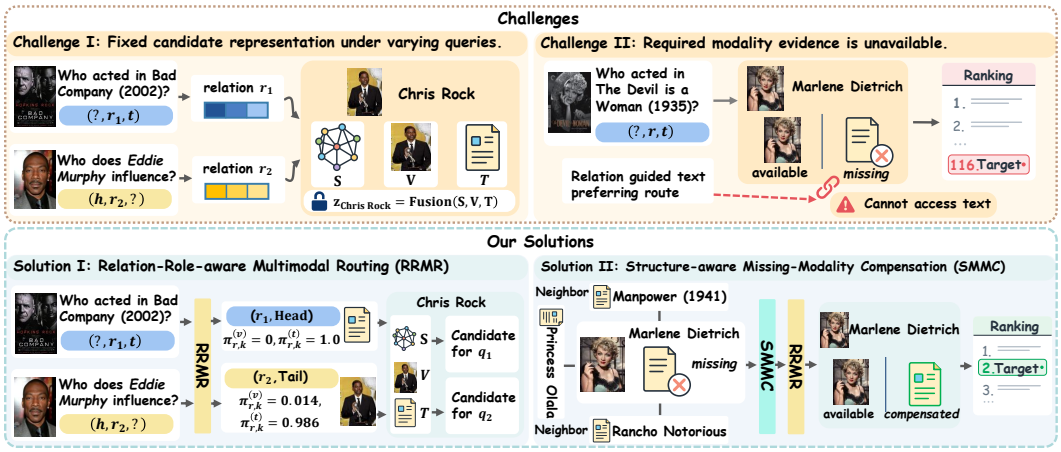


Figure 1: ReRoute addresses varying evidence requirements across relations and head/tail roles (left, RRMR) and missing modalities (right, SMMC followed by RRMR).

composable scoring without explicitly constructing conditioned candidate representations for each query. Our main contributions are:

- We propose RRMR, which uses the relation and head/tail role to modulate modality-specific enhancement components on both the query and candidate sides, with decomposable scoring over all entities.
- We propose SMMC, which compensates for missing modalities by aggregating one-hop neighbors' enhancement components via base-representation similarity, without additional trainable parameters.
- Experiments on DB15K, MKG-W, and MKG-Y demonstrate ReRoute's effectiveness. Compared with the strongest baseline on each dataset, ReRoute achieves relative MRR improvements of 5.02% on MKG-W and 6.93% on MKG-Y.

2 Related Work

Multimodal Entity Representation and Fusion. Multimodal entity representation learning integrates graph structure with visual and textual information to support link prediction (Li et al., 2023). Distributional alignment and adaptive fusion address differences in modality distributions and information content (Baltrušaitis et al., 2018; Cao et al., 2022; Gao et al., 2025). Fine-grained methods further model semantic units and interactions within and across modalities, as in MyGO (Zhang et al., 2025b) and TSAM (Li et al., 2025). Building on fine-grained entity encoding, ReRoute constructs independently adjustable visual and textual enhancement components modulated by the relation and head/tail role; further related work appears in Appendix B.

Relation- and Context-Aware Multimodal Modeling. The predictive value of modality information depends on the relation and link context (Zhao et al., 2022). Existing approaches incorporate prediction context through image selection, dynamic fusion, or expert aggregation, as in LAFA (Shang et al., 2024), NativE (Zhang et al., 2024a), and MoMoK (Zhang et al., 2025a). ReRoute jointly uses relations and head/tail roles to modulate modality contributions on both the query and candidate sides; shared candidate-side gates yield decomposable scoring over all entities.

Missing-Modality Compensation and Robust Multimodal Reasoning. Missing modalities and modality noise affect evidence completeness and reliability, respectively. Generative methods learn cross-modal associations to construct substitute representations for missing modalities (Pezeshkpour et al., 2018), while other methods improve robustness through perturbations and consistency constraints (Jian et al., 2025; Liu et al., 2026). ReRoute's SMMC applies similarity-weighted aggregation to neighbors' enhancement components in undirected one-hop neighborhoods, without additional trainable parameters. RRMR then modulates these supplementary signals.

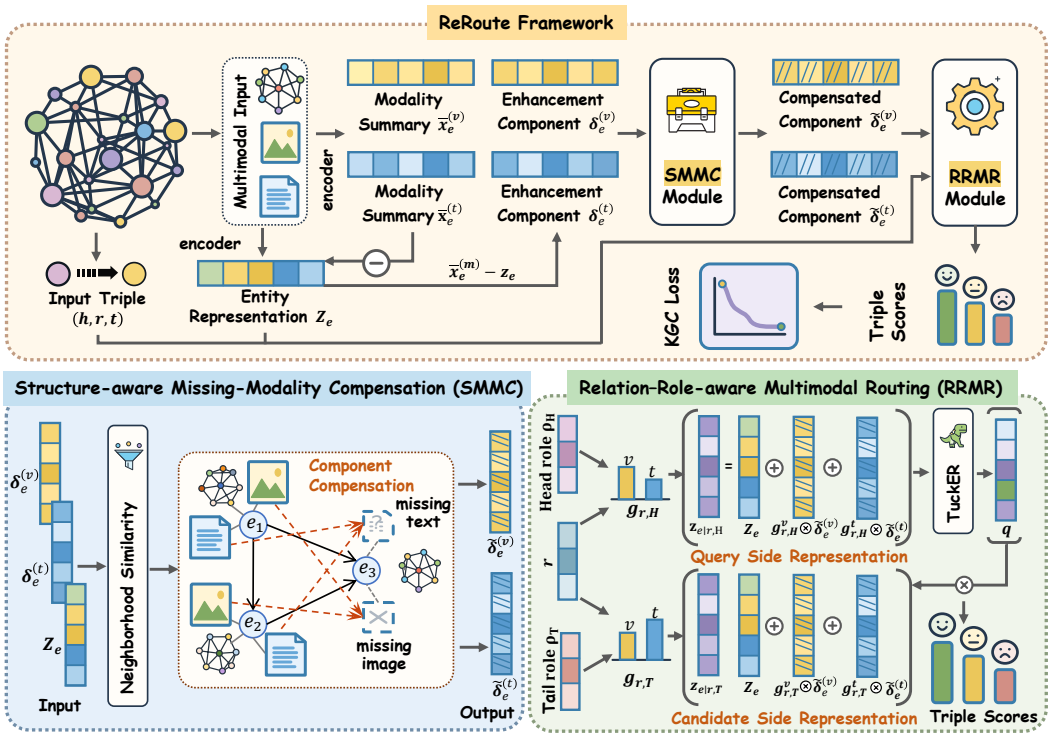


Figure 2: ReRoute constructs base representations and enhancement components through entity encoding. SMMC compensates for missing modalities, RRMR conditions both query and candidate sides, and TuckER decoding supplies query vectors for decomposable scoring.

3 Problem Definition

Let $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$ be a knowledge graph with entity set $\mathcal{E}$, relation set $\mathcal{R}$, and observed triples $\mathcal{T} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$. A triple $(h, r, t) \in \mathcal{T}$ states that relation r holds between head entity h and tail entity t. Entities may additionally have images and textual descriptions; either or both modalities may be missing. Training triples form $\mathcal{T}_{\text{train}} \subseteq \mathcal{T}$. MMKGC uses training triples and available modality information to predict a missing head or tail entity. For a tail prediction query $\xi = (h, r, ?)$ or head prediction query $\xi = (?, r, t)$, the scoring function $f_\xi : \mathcal{E} \to \mathbb{R}$ assigns $f_\xi(e)$ to each candidate $e \in \mathcal{E}$. Higher scores indicate greater plausibility of the completed triple.

Let $N = |\mathcal{E}|$ and d denote the number of entities and shared embedding dimension, respectively. The visual and textual enhancement branches are indexed by $\mathcal{M} = \{v, t\}$, and $\kappa \in \{\mathrm{H}, \mathrm{T}\}$ denotes an entity's head/tail role in the current triple. We denote entity e's multimodal base representation by $\mathbf{z}_e \in \mathbb{R}^d$ and its representation conditioned on relation r and role κ by $\mathbf{z}_{e|r,\kappa} \in \mathbb{R}^d$. Vectors are column vectors; when stacked row-wise, each entity vector is transposed into the corresponding matrix row.

4 Methodology

ReRoute (Figure 2) constructs multimodal base representations and modality-specific enhancement components, applies SMMC and RRMR, and performs query encoding and decomposable scoring over all entities. SMMC weights neighbors by base representation similarity, assuming that similar neighbors can provide relevant modality-specific enhancement components for a missing modality. RRMR separates modality allocation from the enhancement budget, allowing a strong modality preference even when the overall enhancement magnitude is small.

4.1 MULTIMODAL REPRESENTATION CONSTRUCTION

ReRoute adopts fine-grained entity encoding (Zhang et al., 2025b; Li et al., 2025) and employs BERT (Devlin et al., 2019) and BEiT (Peng et al., 2022) to tokenize textual descriptions and entity images, respectively. The resulting token embeddings are mapped to the base representation dimension. Before cross-modal entity encoding, masked mean pooling produces a summary $\overline{\mathbf{x}}_e^{(m)}$ for each modality $m \in \mathcal{M}$; an entirely missing modality has a zero summary. A Transformer (Vaswani et al., 2017) models interactions among the structural embedding and visual and textual tokens, producing the fused multimodal base representation $\mathbf{z}_e$. Each independently adjustable modality-specific enhancement component is the offset from the base representation to its modality summary, masked by modality availability:

$$\boldsymbol{\delta}_e^{(m)} = a_e^{(m)} \left(\overline{\mathbf{x}}_e^{(m)} - \mathbf{z}_e \right), \tag{1}$$

where $a_e^{(m)} \in \{0, 1\}$ indicates whether modality m is available for entity e.

4.2 STRUCTURE-AWARE MISSING-MODALITY COMPENSATION

SMMC provides supplementary signals for missing modalities whose enhancement components are zero. We construct an undirected adjacency graph from training triples and define $\mathcal{N}_m(e)$ as entity e's one-hop neighbors with modality m available. Cosine similarity between base representations determines the neighbor weights:

$$\alpha_{ej}^{(m)} = \frac{\exp(\cos(\mathbf{z}_e, \mathbf{z}_j)/\tau_{\text{att}})}{\sum_{k \in \mathcal{N}_m(e)} \exp(\cos(\mathbf{z}_e, \mathbf{z}_k)/\tau_{\text{att}})}, \qquad j \in \mathcal{N}_m(e), \tag{2}$$

where $\tau_{\text{att}} > 0$ is a temperature parameter. Let η_m denote the global missing rate, the proportion of entities missing modality m. Compensation is enabled when $\eta_m \geq \tau_{\text{miss}}$, with $\tau_{\text{miss}} = 0.1$, and applies only to entities missing that modality:

$$\widetilde{\boldsymbol{\delta}}_e^{(m)} = \begin{cases} \lambda_{\text{comp}} \displaystyle\sum_{j \in \mathcal{N}_m(e)} \alpha_{ej}^{(m)} \boldsymbol{\delta}_j^{(m)}, & a_e^{(m)} = 0 \text{ and } \eta_m \geq \tau_{\text{miss}}, \\ \boldsymbol{\delta}_e^{(m)}, & \text{otherwise}, \end{cases} \tag{3}$$

where $\lambda_{\text{comp}} \in [0, 1]$ is the compensation weight; an empty set of available neighbors yields the zero vector. Aggregation uses neighbors' enhancement components in their pre-compensation state, restricting supplementary signals to one-hop neighborhoods and preserving the enhancement components of observed modalities. Weights are computed directly from base representations without additional trainable parameters. The compensated components are stacked row-wise by entity into $\widetilde{\mathbf{D}}^{(m)} \in \mathbb{R}^{N \times d}$ for subsequent routing.

4.3 RELATION-ROLE-AWARE MULTIMODAL ROUTING

RRMR conditions modality gates on the relation and head/tail role to control visual and textual allocation and the overall enhancement magnitude on both query and candidate sides. Let $\overline{\mathbf{r}}$ and $\boldsymbol{\rho}_\kappa$ be learnable embeddings of relation r and role κ. Layer normalization and a linear mapping transform their sum into the routing output:

$$\boldsymbol{\ell}_{r,\kappa} = \mathbf{W}_g \, \text{LN}_g \left(\overline{\mathbf{r}} + \boldsymbol{\rho}_\kappa \right) + \mathbf{b}_g, \tag{4}$$

where $\mathbf{W}_g \in \mathbb{R}^{3 \times d}$ and $\mathbf{b}_g \in \mathbb{R}^3$ are learnable parameters. The outputs $\ell_{r,\kappa}^{(v)}$, $\ell_{r,\kappa}^{(t)}$, and $\ell_{r,\kappa}^{\text{bud}}$ correspond to the visual modality, textual modality, and enhancement budget, respectively. A softmax over the first two determines modality allocation; the budget component controls the overall enhancement magnitude:

$$g_{r,\kappa}^{(m)} = \beta_{\max} \sigma\big(\ell_{r,\kappa}^{\text{bud}}\big) \, \frac{\exp\Big(\ell_{r,\kappa}^{(m)}\Big)}{\sum_{m' \in \mathcal{M}} \exp\Big(\ell_{r,\kappa}^{(m')}\Big)}, \tag{5}$$

where σ is the sigmoid function and $\beta_{\max} > 0$ bounds the enhancement budget, so the gates sum to at most $\beta_{\max}$. The budget is $\beta_{r,\kappa} = \beta_{\max} \sigma(\ell_{r,\kappa}^{\text{bud}})$, and the allocation weights $\pi_{r,\kappa}^{(m)} = g_{r,\kappa}^{(m)} / \beta_{r,\kappa}$

satisfy $\pi_{r,\kappa}^{(v)} + \pi_{r,\kappa}^{(t)} = 1$. The conditioned entity representation is

$$\mathbf{z}_{e|r,\kappa} = \mathbf{z}_e + \lambda_{\text{rout}} \sum_{m \in \mathcal{M}} g_{r,\kappa}^{(m)} \widetilde{\boldsymbol{\delta}}_e^{(m)}, \tag{6}$$

where λ_{rout} is the global routing weight, set to 1 in the main model. For tail prediction $(h, r, ?)$, the known entity uses head-role gates and candidates use tail-role gates; these roles reverse for head prediction $(?, r, t)$. Routing network parameters are shared across relation–role combinations; candidates within each combination share gates but retain entity-specific enhancement components.

4.4 Query Encoding and Decomposable Scoring

Triple context encoding arranges the conditioned known entity representation, relation embedding, and learnable prediction token in head–relation–tail order. Position embeddings are added before the sequence enters a Transformer. For tail prediction $(h, r, ?)$, the head position contains $\mathbf{z}_{h|r,\text{H}}$ and the tail position contains the prediction token. For head prediction, the known tail entity and prediction token occupy the tail and head positions, respectively. TuckER (Balažević et al., 2019) models interactions between the encoded prediction and relation positions to produce the query vector $\mathbf{q}_\xi \in \mathbb{R}^d$. Let κ_ξ denote the role predicted in query ξ. Matching the query vector with a conditioned candidate representation and substituting Equation (6) gives

$$\begin{aligned} f_\xi(e) &= \mathbf{q}_\xi^\top \mathbf{z}_{e|r,\kappa_\xi} \\ &= \mathbf{q}_\xi^\top \mathbf{z}_e + \lambda_{\text{rout}} \sum_{m \in \mathcal{M}} g_{r,\kappa_\xi}^{(m)} \mathbf{q}_\xi^\top \widetilde{\boldsymbol{\delta}}_e^{(m)}. \end{aligned} \tag{7}$$

For B queries, stack the query vectors and base entity representations row-wise into $\mathbf{Q} \in \mathbb{R}^{B \times d}$ and $\mathbf{Z} \in \mathbb{R}^{N \times d}$, and let $\mathbf{g}^{(m)} \in \mathbb{R}^B$ hold the candidate-side modality gates. The score matrix over all entities is

$$\mathbf{F} = \mathbf{Q}\mathbf{Z}^\top + \lambda_{\text{rout}} \sum_{m \in \mathcal{M}} \left(\mathbf{g}^{(m)} \mathbf{1}_N^\top \right) \odot \left(\mathbf{Q} (\widetilde{\mathbf{D}}^{(m)})^\top \right), \tag{8}$$

where $\mathbf{F} \in \mathbb{R}^{B \times N}$, $\mathbf{1}_N \in \mathbb{R}^N$ is a column vector of ones, and $\odot$ denotes element-wise multiplication; each query's gates are broadcast along the candidate dimension. This equals inner-product scoring with explicitly constructed conditioned candidate representations (see Appendix C.2), requiring $O(BNd)$ time and $O(Nd + BN)$ storage while avoiding an intermediate $B \times N \times d$ tensor.

4.5 Training Objective and Inference

ReRoute treats knowledge graph completion as classification over all entities. During training, the head or tail entity of each training triple is masked with equal probability and serves as the prediction target. For query ξ_i, a softmax gives the candidate probabilities:

$$p(e \mid \xi_i) = \frac{\exp(F_{i,e})}{\sum_{e' \in \mathcal{E}} \exp(F_{i,e'})}. \tag{9}$$

For ground-truth target entity $e_i^\star$, label smoothing gives $y_{i,e}^{(\varepsilon)} = (1 - \varepsilon)\mathbb{I}[e = e_i^\star] + \varepsilon/N$, where $\varepsilon \in [0, 1)$ is the smoothing coefficient. The link prediction loss is

$$\mathcal{L}_{\text{KGC}} = -\frac{1}{B} \sum_{i=1}^{B} \sum_{e \in \mathcal{E}} y_{i,e}^{(\varepsilon)} \log p(e \mid \xi_i). \tag{10}$$

Inference reuses $\mathbf{Z}$ and $\widetilde{\mathbf{D}}^{(m)}$ across queries, generating each query's gates and vector, scoring all entities via Equation (8), and ranking them by descending score. Algorithm 1 in Appendix C.1 summarizes the complete training and inference procedure.

5 Experiments

We evaluate ReRoute on DB15K, MKG-W, and MKG-Y through six research questions:

RQ1: How does ReRoute compare with existing methods in link prediction? (Section 5.2)

RQ2: How do enhancement placement, role conditioning, the adaptive enhancement budget, and SMMC contribute to performance, and how sensitive is performance to the compensation and routing weights? (Section 5.3)

RQ3: How does RRMR adjust modality allocation and enhancement magnitude across relations and head/tail roles? (Section 5.4)

RQ4: Can SMMC mitigate performance degradation under additional missing textual information at test time? (Section 5.5)

RQ5: How does decomposable scoring compare with explicit conditioned scoring in candidate scoring time and additional peak GPU memory usage? (Section 5.6)

RQ6: How do head and tail routing outputs differ across relations? (Section 5.7)

5.1 Experimental Settings

Datasets. We use three public MMKGC benchmarks, DB15K (Liu et al., 2019), MKG-W, and MKG-Y (Xu et al., 2022), containing structural triples and entity-associated visual and textual information. Modality missing rates denote the proportions of entities with no available tokens in the corresponding modality. MKG-W has 40.27% missing visual information and MKG-Y has 17.89% missing textual information, activating SMMC for the respective modality; both missing rates on DB15K fall below its 10% threshold. Full statistics appear in Appendix D.1, Table 3.

Evaluation metrics. We score all entities for head and tail queries (Bordes et al., 2013) under the filtered ranking protocol (Zhang et al., 2025b). We report mean reciprocal rank (MRR) and Hits@K ($K \in \{1, 3, 10\}$) (Sun et al., 2019): reciprocal target ranks and top-K success rates, respectively, averaged across queries.

Baselines. Table 1 compares ReRoute with published results for three structural and 15 multimodal knowledge graph completion methods (Appendix D.2). We train and evaluate ReRoute, Backbone (without SMMC and RRMR), and all ablation variants.

Implementation details. ReRoute is implemented in PyTorch (Paszke et al., 2019) with an entity representation dimension of 256 and a Transformer feed-forward dimension of 1,024. We train with Adam (Kingma & Ba, 2015) (learning rate 4×10^{-4}, weight decay 10^{-5}), batch size 2,048, and a cosine annealing schedule with warm restarts (Loshchilov & Hutter, 2017) for 3,500 epochs, evaluating every 50 epochs. The label smoothing coefficient is $\varepsilon = 0.2$; compensation weight λ_{comp} and routing weight λ_{rout} are both 1.0 except in sensitivity experiments. The enhancement budget upper bound is $\beta_{\max} = 0.2$, and the SMMC similarity temperature is $\tau_{\mathrm{att}} = 0.1$. Table 1 reports means and standard deviations over three random seeds; the full configuration (dropout rates, token counts, and the learning rate schedule) is in Appendix D.4. Efficiency experiments use an NVIDIA A100 GPU (40 GB for the candidate scoring benchmark).

5.2 Performance Comparison (RQ1)

Table 1 shows that ReRoute ranks first on 10 of 12 dataset–metric combinations and second on the remaining two. It achieves the highest scores across all metrics on MKG-W and MKG-Y, with relative MRR improvements of 5.02% and 6.93% over TSAM, respectively. On DB15K, it achieves the highest Hits@3 and Hits@10 scores but ranks second to M-Hyper in MRR (41.12% versus 41.25%) and Hits@1.

5.3 Ablation Studies and Parameter Sensitivity (RQ2)

Table 2 evaluates six variants. **Only Query** and **Only Candidate** retain routed enhancement on the known entity or candidate side only, scoring the other side with its base representation; both keep SMMC. **w/o Role** fixes the router's head/tail role input to a shared embedding (decoder position embeddings unchanged). **w/o Adaptive Budget** fixes the budget at 1.0, bypassing the 0.2 upper bound while retaining softmax modality allocation. **w/o SMMC** disables compensation; **w/o SMMC & RRMR** removes both modules (Backbone).

Table 1: Link prediction results (%). Baselines: published results; ReRoute: mean ± standard deviation over three seeds. Bold/underlined: best/second best. Relative improvements use each column's strongest baseline; – indicates no improvement.

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
Structural knowledge graph completion methods												
TransE (NeurIPS'13)	24.86	12.78	31.48	47.07	29.19	21.06	33.20	44.23	30.73	23.45	35.18	43.37
TuckER (EMNLP'19)	33.86	25.33	37.91	50.38	30.39	24.44	32.91	41.25	37.05	34.59	38.43	41.45
RotatE (ICLR'19)	29.28	17.87	36.12	49.66	33.67	26.80	36.68	46.73	34.95	29.10	38.35	45.30
Multimodal knowledge graph completion methods												
IKRL (IJCAI'17)	26.82	14.09	34.93	49.09	32.36	26.11	34.75	44.07	33.22	30.37	34.28	38.26
RSME (ACM MM'21)	29.76	24.15	32.12	40.29	29.23	23.36	31.97	40.43	34.44	31.78	36.07	39.09
MMRNS (ACM MM'22)	32.68	23.01	37.86	51.01	35.03	28.59	37.49	47.47	35.93	30.53	39.07	45.47
OTKGE (NeurIPS'22)	23.86	18.45	25.89	34.23	34.36	28.85	36.25	44.88	35.51	31.97	37.18	41.38
VISTA (EMNLP'23)	30.42	22.49	33.56	45.94	32.91	26.12	35.38	45.61	30.45	24.87	32.39	41.53
QEB (ACM MM'23)	28.18	14.82	36.67	51.55	32.38	25.47	35.06	45.32	34.37	29.49	36.95	42.32
IMF (WWW'23)	32.25	24.20	36.00	48.19	34.50	28.77	36.62	45.44	35.79	32.95	37.14	40.63
AdaMF (COLING'24)	32.51	21.31	39.67	51.68	34.27	27.21	37.86	47.21	38.06	33.49	40.44	45.48
SNAG (COLING'25)	36.30	27.40	41.10	53.00	37.30	30.20	40.50	50.30	39.10	34.70	41.08	46.70
NativE (SIGIR'24)	34.30	25.08	39.48	51.35	36.84	29.94	40.06	49.39	39.21	35.03	41.21	46.25
MoMoK (ICLR'25)	39.57	32.38	43.45	54.14	35.89	30.38	37.54	46.13	37.91	35.09	39.20	43.20
MyGO (AAAI'25)	37.72	30.08	41.26	52.21	36.10	29.78	38.54	47.75	38.44	35.01	39.84	44.19
TSAM (TMM'25)	40.50	32.60	44.36	55.44	<u>40.07</u>	<u>33.29</u>	<u>42.53</u>	<u>52.72</u>	<u>39.80</u>	35.28	41.29	46.44
M-Hyper (ACL'26)	**41.25**	**33.64**	<u>45.01</u>	56.09	37.02	31.24	39.16	48.84	39.46	<u>36.02</u>	40.92	45.22
HFR-MKGC (AAAI'26)	35.20	23.04	43.25	<u>56.41</u>	38.62	31.37	42.12	51.66	39.00	34.41	<u>42.12</u>	<u>47.31</u>
ReRoute (Ours)	<u>41.12</u> ±0.12	<u>33.18</u> ±0.16	**45.14** ±0.05	**56.50** ±0.07	**42.08** ±0.35	**35.13** ±0.54	**44.85** ±0.31	**55.26** ±0.12	**42.56** ±0.44	**37.80** ±0.42	**44.23** ±0.45	**51.29** ±0.42
Relative improvement	–	–	+0.29%	+0.16%	+5.02%	+5.53%	+5.45%	+4.82%	+6.93%	+4.94%	+5.01%	+8.41%

Table 2: Ablation results (%), averaged over three random seeds with fixed visual/textual token limits of 16/24. Bold: best. –: w/o SMMC omitted because both missing rates on DB15K fall below the activation threshold.

Setting	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
ReRoute (Full)	**41.12**	**33.18**	**45.14**	**56.50**	**42.08**	**35.13**	**44.85**	**55.26**	**42.56**	**37.80**	**44.23**	**51.29**
Only Query	40.43	32.41	44.34	55.95	41.45	34.20	44.21	55.16	41.93	37.00	43.89	50.96
Only Candidate	39.51	31.43	43.32	55.07	41.34	34.14	44.14	55.22	41.98	37.21	43.94	50.56
w/o Role	40.58	32.52	44.49	55.87	41.08	34.03	43.62	54.88	41.84	36.82	43.77	50.94
w/o Adaptive Budget	40.87	32.83	44.86	56.08	41.58	34.48	44.33	55.12	42.08	37.23	44.07	50.88
w/o SMMC	–	–	–	–	41.45	34.33	44.25	54.86	41.90	37.16	43.66	50.75
w/o SMMC & RRMR	38.54	30.45	42.47	53.94	40.10	32.90	42.99	53.49	39.83	35.27	41.61	47.78

The full model outperforms all variants across all metrics. Adding candidate-side enhancement to Only Query raises MRR by 0.69, 0.63, and 0.63 percentage points on DB15K, MKG-W, and MKG-Y, respectively; gains over Only Candidate further support enhancement on both sides. Removing role conditioning reduces performance, while removing SMMC lowers MRR by 0.63 and 0.66 percentage points on MKG-W and MKG-Y. In w/o Adaptive Budget, the fixed budget exceeds Full's upper bound, so this comparison evaluates the combined effect of budget adaptation and magnitude control.

Parameter sensitivity. We vary λ_{comp} over $\{0, 0.3, 0.6, 1.0\}$ and λ_{rout} over $\{0.5, 0.75, 1.0, 1.25, 1.5\}$ on MKG-W and MKG-Y, retraining for each value with the other parameter fixed at 1.0.

Figure 3 shows that MRR and Hits@1 peak at 1.0 in each sweep on both datasets. Increasing λ_{comp} improves both metrics monotonically on MKG-W; on MKG-Y, both metrics initially decrease at 0.3 before improving. Both metrics decrease when λ_{rout} deviates from 1.0. These results support the defaults $\lambda_{\text{comp}} = \lambda_{\text{rout}} = 1.0$.

5.4 Analysis of Relation–Role Conditioning (RQ3)

Figure 4 shows visual allocation weights and enhancement budgets for 12 relations on MKG-W. Enhancement preferences vary across relations and between head and tail roles within the same relation. These patterns indicate that RRMR adapts both modality allocation and overall enhancement magnitude to the relation and head/tail role.

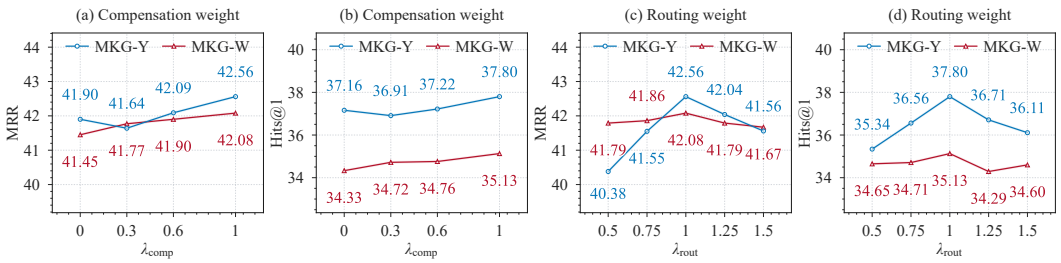


Figure 3: MRR and Hits@1 on MKG-W and MKG-Y versus (a, b) compensation weight λ_{comp} and (c, d) routing weight λ_{rout}, averaged over three random seeds.

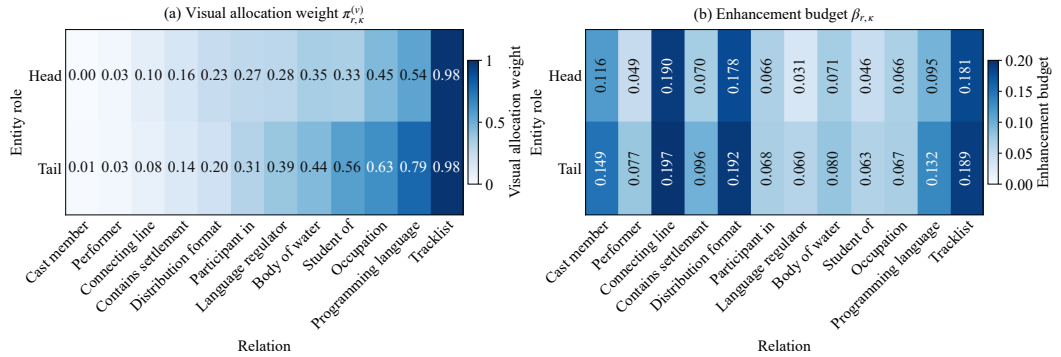


Figure 4: Relation–role conditioning on MKG-W: (a) visual allocation weight $\pi^{(v)}_{r,\kappa}$ (textual: $1 - \pi^{(v)}_{r,\kappa}$); (b) enhancement budget $\beta_{r,\kappa}$. Columns: relations; rows: head/tail roles.

5.5 Robustness to Missing Textual Information (RQ4)

After training the models on MKG-Y with its original modality availability, we freeze their parameters and randomly mask available textual information, increasing the total textual missing rate from approximately 18% to 30%, 40%, 50%, and 60%. All models share the same nested masking sets, and masked information is excluded from entity encoding and neighborhood compensation.

As shown in Figure 5, performance declines for all models as the textual missing rate increases, but ReRoute consistently achieves the highest MRR and Hits@10. At a missing rate of 60%, ReRoute outperforms the variant retaining only RRMR by 2.68 percentage points in MRR and 4.17 percentage points in Hits@10. These results indicate that SMMC can mitigate the performance degradation caused by additional missing textual information at test time.

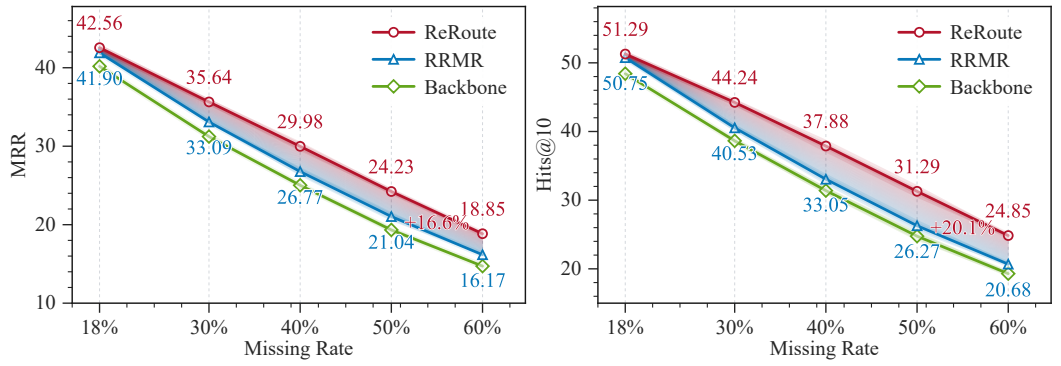


Figure 5: MRR and Hits@10 (%) under additional missing textual information at test time on MKG-Y. RRMR: ReRoute without SMMC; Backbone: without SMMC and RRMR. Shading highlights ReRoute's gains over RRMR; annotated percentages indicate relative improvements.

5.6 Candidate Scoring Efficiency (RQ5)

Figure 6 compares candidate scoring costs on MKG-Y across different numbers of candidates. With 15,000 candidates, decomposable scoring achieves a $17.6\times$ speedup over optimized explicit conditioned scoring and reduces additional peak GPU memory usage to $1/38.6$ of that required by the latter. The absolute MRR difference between the two implementations is 1.3×10^{-9}. Model-level inference on MKG-Y takes 92 μs per query for ReRoute versus 80 μs for MyGO; Appendix D.5, Table 4, reports the full comparison.

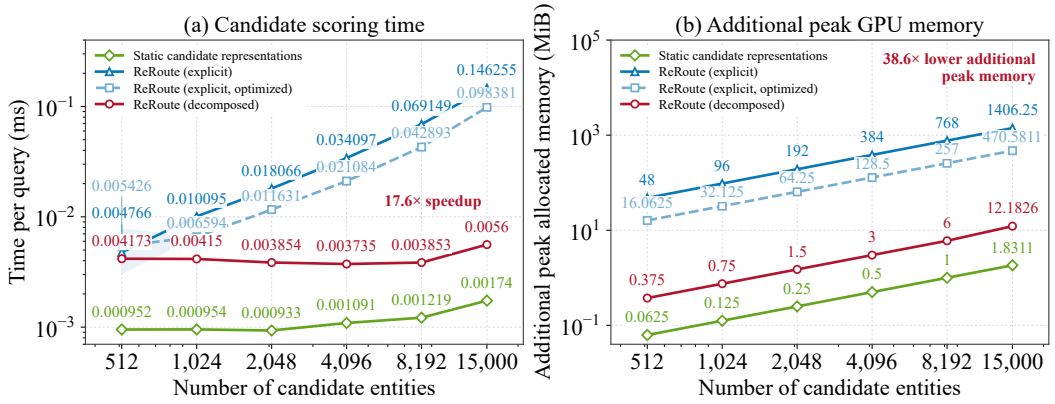


Figure 6: Candidate scoring costs on MKG-Y (NVIDIA A100, 40 GB): (a) amortized scoring time per query; (b) additional peak GPU memory. Both axes are logarithmic.

5.7 Visualization of Routing Outputs (RQ6)

Figure 7 jointly projects all $2|\mathcal{R}| = 338$ three-dimensional routing outputs $\ell_{r,\kappa}$ on MKG-W using t-SNE (Van der Maaten & Hinton, 2008). Eight relations are highlighted by sorting relations by the Euclidean distance between their head and tail routing outputs and selecting evenly across the distance range; the remaining outputs are shown at reduced opacity. Distances in the original three-dimensional space are 0.0437 for *Cause of death* and 0.0039 for *Designed by*. RRMR can retain similar routing outputs for one relation's two roles while producing more distinct outputs for another. Role conditioning allows head–tail differentiation to vary across relations. These distances quantify variation in routing outputs; a common offset to both modality logits leaves the softmax allocation unchanged. Figure 4 shows the learned allocation weights and enhancement budgets, while the w/o Role ablation tests the predictive value of role conditioning.

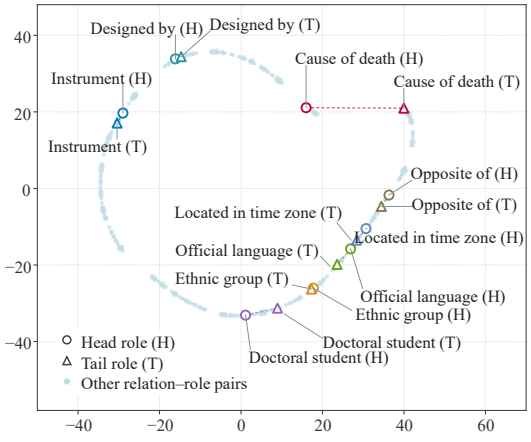


Figure 7: t-SNE of relation–role routing outputs on MKG-W. Colors: relations; circles/triangles: head/tail roles; dashed lines connect the two roles of each relation.

6 Conclusion and Future Work

ReRoute combines relation-role-aware multimodal routing and structure-aware missing-modality compensation for MMKGC. Experiments on three benchmarks demonstrate ReRoute's effectiveness and robustness to additional missing textual information. Ablation studies confirm the contribution of each module. Decomposable scoring preserves explicitly conditioned scores while reducing candidate scoring time and additional peak GPU memory. Future work will examine the reliability of supplementary neighborhood evidence when modality information is scarce.

AI USE STATEMENT

We used AI solely to assist with English grammar checking. AI was not involved in the creation or development of the manuscript's content. All content was written and revised by the authors, who take full responsibility for the authenticity and accuracy of the work.

ETHICS STATEMENT

This work uses only the publicly available DB15K, MKG-W, and MKG-Y benchmark datasets from prior studies. We did not collect new data or conduct experiments involving human participants. As these benchmarks are derived from real-world knowledge sources, they may contain biases, inaccuracies, or incomplete information, which may in turn be reflected in model predictions. The datasets were used solely for research purposes and in accordance with their original usage conditions.

REPRODUCIBILITY STATEMENT

We provide the code for reproducing all experiments at `https://anonymous.4open.science/r/ReRoute-code-47BE/`. Detailed descriptions of the training and inference procedures, theoretical proofs, datasets, evaluation protocols, implementation details, and experimental settings are provided in the main paper and Appendices C.1–D.4. Additional details on the candidate scoring benchmark and neighborhood coverage statistics are included in Appendices D.6 and D.7.

REFERENCES

Ivana Balažević, Carl Allen, and Timothy Hospedales. Tucker: Tensor factorization for knowledge graph completion. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pp. 5185–5194, 2019.

Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2): 423–443, 2018.

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

Zongsheng Cao, Qianqian Xu, Zhiyong Yang, Yuan He, Xiaochun Cao, and Qingming Huang. Otkge: Multi-modal knowledge graph embeddings via optimal transport. In *Advances in neural information processing systems*, volume 35, pp. 39090–39102, 2022.

Zhuo Chen, Yin Fang, Yichi Zhang, Lingbing Guo, Jiaoyan Chen, Jeff Z Pan, Huajun Chen, and Wen Zhang. Noise-powered multi-modal knowledge graph representation framework. In *Proceedings of the 31st international conference on computational linguistics*, pp. 141–155, 2025.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Yuxiao Gao, Fuwei Zhang, Zhao Zhang, Xiaoshuang Min, and Fuzhen Zhuang. Mixed-curvature multi-modal knowledge graph completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 11699–11707, 2025.

Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard De Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, et al. Knowledge graphs. *ACM Computing Surveys (Csur)*, 54(4):1–37, 2021.

Yue Jian, Xiangyu Luo, Zhifei Li, Miao Zhang, Yan Zhang, Kui Xiao, and Xiaoju Hou. Apkgc: Noise-enhanced multi-modal knowledge graph completion with attention penalty. In *Proceedings of the AAAI conference on artificial intelligence*, volume 39, pp. 15005–15013, 2025.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*, 2015.

Jaejun Lee, Chanyoung Chung, Hochang Lee, Sungho Jo, and Joyce Jiyoung Whang. Vista: Visual-textual knowledge graph representation learning. In *Findings of the association for computational linguistics: EMNLP 2023*, pp. 7314–7328, 2023.

Linyu Li, Zhi Jin, Yichi Zhang, Dongming Jin, Chengfeng Dou, Yuanpeng He, Xuan Zhang, and Haiyan Zhao. Towards structure-aware model for multi-modal knowledge graph completion. *IEEE Transactions on Multimedia*, 2025.

Xinhang Li, Xiangyu Zhao, Jiaxing Xu, Yong Zhang, and Chunxiao Xing. Imf: interactive multimodal fusion model for link prediction. In *Proceedings of the ACM web conference 2023*, pp. 2572–2580, 2023.

Ye Liu, Hui Li, Alberto Garcia-Duran, Mathias Niepert, Daniel Onoro-Rubio, and David S Rosenblum. Mmkg: multi-modal knowledge graphs. In *European Semantic Web Conference*, pp. 459–474, 2019.

Zhiqiang Liu, Yichi Zhang, Mengshu Sun, Lei Liang, and Wen Zhang. Collaboration of fusion and independence: Hypercomplex-driven robust multi-modal knowledge graph completion. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 27968–27983, 2026.

Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *Proceedings of the 5th International Conference on Learning Representations*, 2017.

Maximilian Nickel, Kevin Murphy, Volker Tresp, and Evgeniy Gabrilovich. A review of relational machine learning for knowledge graphs. *Proceedings of the IEEE*, 104(1):11–33, 2015.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, volume 32, 2019.

Zhiliang Peng, Li Dong, Hangbo Bao, Qixiang Ye, and Furu Wei. Beit v2: Masked image modeling with vector-quantized visual tokenizers. *arXiv preprint arXiv:2208.06366*, 2022.

Pouya Pezeshkpour, Liyan Chen, and Sameer Singh. Embedding multimodal relational data for knowledge base completion. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pp. 3208–3218, 2018.

Bin Shang, Yinliang Zhao, Jun Liu, and Di Wang. Lafa: Multimodal knowledge graph completion with link aware fusion and aggregation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 8957–8965, 2024.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

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

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, volume 30, 2017.

Di Wang, Junping Du, Zhe Xue, Meiyu Liang, Guanhua Ye, Yingxia Shao, and Haisheng Li. Hfr-mkgc: Hierarchical fusion reasoning with mllms for multi-modal knowledge graph completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 15815–15823, 2026.

Meng Wang, Sen Wang, Han Yang, Zheng Zhang, Xi Chen, and Guilin Qi. Is visual context really helpful for knowledge graph? a representation learning perspective. In *Proceedings of the 29th ACM international conference on multimedia*, pp. 2735–2743, 2021.

Xin Wang, Benyuan Meng, Hong Chen, Yuan Meng, Ke Lv, and Wenwu Zhu. Tiva-kg: A multimodal knowledge graph with text, image, video and audio. In *Proceedings of the 31st ACM international conference on multimedia*, pp. 2391–2399, 2023.

Ruobing Xie, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. Image-embodied knowledge representation learning. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pp. 3140–3146, 2017.

Derong Xu, Tong Xu, Shiwei Wu, Jingbo Zhou, and Enhong Chen. Relation-enhanced negative sampling for multimodal knowledge graph completion. In *Proceedings of the 30th ACM international conference on multimedia*, pp. 3857–3866, 2022.

Yichi Zhang, Zhuo Chen, Lingbing Guo, Yajing Xu, Binbin Hu, Ziqi Liu, Wen Zhang, and Huajun Chen. 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*, pp. 91–101, 2024a.

Yichi Zhang, Zhuo Chen, Lei Liang, Huajun Chen, and Wen Zhang. Unleashing the power of imbalanced modality information for multi-modal knowledge graph completion. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pp. 17120–17130, 2024b.

Yichi Zhang, Zhuo Chen, Lingbing Guo, Binbin Hu, Ziqi Liu, Wen Zhang, Huajun Chen, et al. Multiple heads are better than one: Mixture of modality knowledge experts for entity representation learning. In *International Conference on Learning Representations*, volume 2025, pp. 54811–54828, 2025a.

Yichi Zhang, Zhuo Chen, Lingbing Guo, Yajing Xu, Binbin Hu, Ziqi Liu, Wen Zhang, and Huajun Chen. Tokenization, fusion, and augmentation: towards fine-grained multi-modal entity representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 13322–13330, 2025b.

Yu Zhao, Xiangrui Cai, Yike Wu, Haiwei Zhang, Ying Zhang, Guoqing Zhao, and Ning Jiang. Mose: Modality split and ensemble for multimodal knowledge graph completion. In *Proceedings of the 2022 conference on empirical methods in natural language processing*, pp. 10527–10536, 2022.

Guoliang Zhu, Tao Ren, Dandan Wang, and Jun Hu. Let modalities teach each other: modal-collaborative knowledge extraction and fusion for multimodal knowledge graph completion. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pp. 6210–6222, 2025.

Xiangru Zhu, Zhixu Li, Xiaodan Wang, Xueyao Jiang, Penglei Sun, Xuwu Wang, Yanghua Xiao, and Nicholas Jing Yuan. Multi-modal knowledge graph construction and application: A survey. *IEEE Trans. Knowl. Data Eng.*, 36(2):715–735, 2024.

A Motivating Examples

This appendix elaborates on the two challenges in constructing candidate representations introduced in Section 1.

Evidence requirements vary with relation and role. In Figure 1 (left), Chris Rock is a candidate head entity for "Who acted in *Bad Company*?" and a candidate tail entity for "Who does Eddie Murphy influence?" The two queries concern acting credits and interpersonal influence, respectively. When a model reuses a single candidate representation across queries, its visual and textual information is combined in a fixed way, and prediction contexts are distinguished only through the query representation and scoring function. Conditioning candidate representations on the relation and head/tail role accommodates these varying evidence requirements; decomposable scoring avoids reconstructing them for each query.

Missing modalities cannot be restored by reweighting. Relation and role conditioning can adjust modality contributions but cannot supply missing information. When Marlene Dietrich's textual information is unavailable (Figure 1, right), increasing the textual modality weight cannot recover the missing evidence. Neighboring entities in the training graph, such as connected films, may provide supplementary evidence, but differ in relevance and modality availability. SMMC selects neighbors with available modality information and weights their enhancement components; RRMR then modulates these supplementary signals according to the relation and head/tail role.

B Detailed Related Work

This appendix supplements Section 2 with a detailed account of existing MMKGC methods.

Multimodal Entity Representation and Fusion. OTKGE (Cao et al., 2022) uses the Wasserstein distance to constrain mappings from individual modality distributions to a unified representation space, limiting disruption to within-modality structure. AdaMF-MAT (Zhang et al., 2024b) combines adaptive modality fusion with modality adversarial training, learning modality weights and generating adversarial samples to address imbalanced modality distributions. MyGO (Zhang et al., 2025b) uses fine-grained discrete tokens with cross-modal entity encoding and fine-grained contrastive learning. TSAM (Li et al., 2025) models cross-modal interactions through fine-grained modality-aware fusion and uses structure-aware contrastive learning to align representations from other modalities with graph structure.

Relation- and Context-Aware Multimodal Modeling. LAFA (Shang et al., 2024) weights entity images by link context, fuses visual and structural representations, and incorporates neighboring entities through link-aware neighborhood aggregation. NativE (Zhang et al., 2024a) uses relation-guided dual adaptive fusion to adjust modality weights according to entity information and relational context. MoMoK (Zhang et al., 2025a) learns multi-view entity representations through relation-guided aggregation of modality knowledge experts, combining expert information disentanglement with multimodal joint decision-making. These approaches incorporate prediction context through image selection, dynamic fusion, or expert aggregation.

Missing-Modality Compensation and Robust Multimodal Reasoning. MKBE (Pezeshkpour et al., 2018) uses learned embeddings and modality-specific decoders to generate missing multimodal values, including text and images. NativE (Zhang et al., 2024a) uses collaborative modality adversarial training to generate synthetic modality embeddings and address imbalanced modality information. SNAG (Chen et al., 2025) applies modality-level Gaussian noise masking alongside Transformer-based entity-level modality interactions. M-Hyper (Liu et al., 2026) retains independent modality representations and fused representations, incorporating noise-driven self-distillation into relation-aware fusion. Its unperturbed fused representations guide the learning of noisy representations to improve robustness to modality perturbations.

C Model Design and Analysis

This appendix presents the ReRoute algorithm, establishes the routing gate properties and the equivalence of decomposable scoring and explicit conditioned scoring, and analyzes candidate matching complexity.

C.1 Algorithm Overview

Algorithm 1 summarizes the complete training and inference procedure of ReRoute. For each mini-batch, the entity encoder and SMMC recompute the representations shared by the query and candidate sides using the current parameters; RRMR then produces the gates for both sides.

Algorithm 1 ReRoute: training and inference.

Require: Training triples $\mathcal{T}_{\text{train}}$, entity set $\mathcal{E}$, relation set $\mathcal{R}$, visual and textual tokens, modality availability indicators $a_e^{(m)}$; hyperparameters τ_{miss}, $\beta_{\max}$, λ_{comp}, λ_{rout}, τ_{att}.

1: Build the undirected adjacency graph from $\mathcal{T}_{\text{train}}$.
2: Compute global missing rates η_m from modality availability.
3: **for** each training epoch **do**
4: **for** each mini-batch of training triples **do**
5: Mask the head or tail entity with equal probability to form the prediction queries.
6: Encode all entities with the current parameters to obtain $\mathbf{z}_e$; stack the base representations row-wise into $\mathbf{Z}$.
7: Compute modality-specific enhancement components $\boldsymbol{\delta}_e^{(m)}$ via (1).
8: Compensate eligible missing modalities via (2)–(3); stack the compensated enhancement components row-wise into $\widetilde{\mathbf{D}}^{(m)}$ for each $m \in \mathcal{M}$.
9: Compute routing outputs $\boldsymbol{\ell}_{r,\kappa}$ and modality gates $g_{r,\kappa}^{(m)}$ via (4)–(5).
10: Condition the known entity representation via (6); encode queries into $\mathbf{Q}$ through triple context encoding and TuckER decoding.
11: Compute the score matrix $\mathbf{F}$ over all entities via (8).
12: Compute $\mathcal{L}_{\text{KGC}}$ via (9)–(10); update all trainable parameters.
13: **end for**
14: **end for**
15: **Inference:** with trained parameters fixed, compute $\mathbf{z}_e$, $\widetilde{\boldsymbol{\delta}}_e^{(m)}$, $\mathbf{Z}$, and $\widetilde{\mathbf{D}}^{(m)}$ once; for each query, generate the gates and $\mathbf{Q}$, compute $\mathbf{F}$ via (8), and rank all entities by descending score.

C.2 Proofs

Proposition 1 (gate budget and modality allocation). For every relation $r \in \mathcal{R}$ and role $\kappa \in \{\mathrm{H}, \mathrm{T}\}$, the modality gates satisfy $\sum_{m\in\mathcal{M}} g_{r,\kappa}^{(m)} = \beta_{r,\kappa} \le \beta_{\max}$, and the modality allocation weights satisfy $\pi_{r,\kappa}^{(v)} + \pi_{r,\kappa}^{(t)} = 1$.

Proof. Because $\sigma(x) = 1/(1+e^{-x}) \in (0,1)$ for every $x \in \mathbb{R}$, the enhancement budget satisfies $0 < \beta_{r,\kappa} = \beta_{\max}\sigma(\ell_{r,\kappa}^{\text{bud}}) < \beta_{\max}$. Summing Equation (5) over $m \in \mathcal{M}$ gives

$$\sum_{m\in\mathcal{M}} g_{r,\kappa}^{(m)} = \beta_{r,\kappa} \frac{\sum_{m\in\mathcal{M}} \exp\left(\ell_{r,\kappa}^{(m)}\right)}{\sum_{m'\in\mathcal{M}} \exp\left(\ell_{r,\kappa}^{(m')}\right)} = \beta_{r,\kappa} \le \beta_{\max}. \tag{11}$$

Dividing Equation (5) by $\beta_{r,\kappa}$ yields $\pi_{r,\kappa}^{(m)} = \exp(\ell_{r,\kappa}^{(m)}) / \sum_{m'\in\mathcal{M}} \exp(\ell_{r,\kappa}^{(m')})$, a two-way softmax, so $\pi_{r,\kappa}^{(v)} + \pi_{r,\kappa}^{(t)} = 1$. □

Proposition 2 (equivalence of decomposable and explicit conditioned scoring). For any B queries and all N candidate entities, each entry of the score matrix $\mathbf{F}$ in Equation (8) equals the explicitly conditioned score $f_{\xi_i}(e) = \mathbf{q}_{\xi_i}^{\top} \mathbf{z}_{e|r_i,\kappa_{\xi_i}}$ of the corresponding query–candidate pair.

Proof. For a query ξ with relation r and predicted role κ_ξ, explicit conditioned scoring substitutes Equation (6) directly into the inner product:

$$f_\xi(e) = \mathbf{q}_\xi^\top \mathbf{z}_{e|r,\kappa_\xi} = \mathbf{q}_\xi^\top \mathbf{z}_e + \lambda_{\text{rout}} \sum_{m \in \mathcal{M}} g_{r,\kappa_\xi}^{(m)} \mathbf{q}_\xi^\top \widetilde{\boldsymbol{\delta}}_e^{(m)}, \tag{12}$$

which is exactly Equation (7). The candidate-side gates $g_{r,\kappa_\xi}^{(m)}$ depend only on the relation–role combination and are shared by all candidates. The inner products $\mathbf{q}_\xi^\top \mathbf{z}_e$ and $\mathbf{q}_\xi^\top \widetilde{\boldsymbol{\delta}}_e^{(m)}$ can therefore be computed for all candidates using the shared matrices $\mathbf{Z}$ and $\widetilde{\mathbf{D}}^{(m)}$, before applying the gates. For B queries, stacking the query vectors row-wise into $\mathbf{Q} \in \mathbb{R}^{B \times d}$ and the base entity representations into $\mathbf{Z} \in \mathbb{R}^{N \times d}$ gives the base matching term $\mathbf{Q}\mathbf{Z}^\top$. Stack the compensated enhancement components of modality m row-wise into $\widetilde{\mathbf{D}}^{(m)} \in \mathbb{R}^{N \times d}$. The modality matching term $\mathbf{Q}(\widetilde{\mathbf{D}}^{(m)})^\top$ is scaled row-wise by the candidate-side gates $\mathbf{g}^{(m)} \in \mathbb{R}^B$, broadcast through the outer product $\mathbf{g}^{(m)}\mathbf{1}_N^\top$. Combining these terms reproduces Equation (8), so each entry of $\mathbf{F}$ equals the explicitly conditioned score of the corresponding query–candidate pair. The decomposition introduces no mathematical approximation; the two scoring procedures differ only in the order in which the shared gates are applied. □

Proposition 3 (compensation preserves observed modalities). For every entity $e \in \mathcal{E}$ and modality $m \in \mathcal{M}$ with $a_e^{(m)} = 1$, compensation leaves the enhancement component unchanged, $\widetilde{\boldsymbol{\delta}}_e^{(m)} = \boldsymbol{\delta}_e^{(m)}$. For an eligible missing modality, an empty neighborhood $\mathcal{N}_m(e) = \emptyset$ yields the zero vector. The compensation step introduces no additional trainable parameters.

Proof. The first two claims follow directly from the case definition in Equation (3): the first branch applies only when $a_e^{(m)} = 0$ and $\eta_m \geq \tau_{\text{miss}}$, and its empty sum evaluates to the zero vector. The aggregation weights in Equation (2) are computed from the cosine similarity of base representations and the fixed temperature τ_{att}. The compensation weight λ_{comp} is also fixed, so compensation introduces no additional trainable parameters. □

C.3 Complexity

Candidate matching multiplies the query matrix by the transposes of the base representation matrix and two modality-specific enhancement matrices, requiring $O(BNd)$ time for B queries and N candidates. Decomposable scoring stores the shared candidate representations and scores, requiring $O(Nd + BN)$ memory. The explicit conditioned scoring implementation used in Section 5.6 materializes an intermediate $B \times N \times d$ tensor of conditioned candidate representations, which the decomposable form avoids. These bounds cover candidate matching with precomputed representations, excluding entity encoding and SMMC.

D Experiment Details

D.1 Datasets and Statistics

DB15K (Liu et al., 2019) is built from DBpedia; MKG-W and MKG-Y (Xu et al., 2022) are subsets of Wikidata and YAGO, respectively. All three benchmarks contain structural triples and entity-associated visual and textual information. Table 3 reports dataset sizes and modality missing rates, calculated over all entities in each dataset. Appendix D.7 reports the availability of neighborhood evidence for entities with missing modalities.

Table 3: Dataset statistics. Modality missing rates denote the proportions of entities with no available tokens in the corresponding modality.

Dataset	#Entities	#Relations	#Train	#Valid	#Test	Visual missing (%)	Textual missing (%)
DB15K	12,842	279	79,222	9,904	9,902	2.12	0.00
MKG-W	15,000	169	34,196	4,276	4,274	40.27	5.67
MKG-Y	15,000	28	21,310	2,665	2,663	5.08	17.89

D.2 BASELINES

Section 5.2 compares ReRoute with 18 baselines. **Structural methods.** The three structural knowledge graph completion methods are TransE (Bordes et al., 2013), TuckER (Balažević et al., 2019), and RotatE (Sun et al., 2019). **Multimodal methods.** The 15 multimodal knowledge graph completion methods are IKRL (Xie et al., 2017), RSME (Wang et al., 2021), MMRNS (Xu et al., 2022), OTKGE (Cao et al., 2022), VISTA (Lee et al., 2023), QEB (Wang et al., 2023), IMF (Li et al., 2023), AdaMF (Zhang et al., 2024b), SNAG (Chen et al., 2025), NativE (Zhang et al., 2024a), MoMoK (Zhang et al., 2025a), MyGO (Zhang et al., 2025b), TSAM (Li et al., 2025), M-Hyper (Liu et al., 2026), and HFR-MKGC (Wang et al., 2026). Baseline results in Table 1 are drawn primarily from MoMoK (Zhang et al., 2025a) and M-Hyper (Liu et al., 2026), with results for TSAM and HFR-MKGC taken from their original papers (Li et al., 2025; Wang et al., 2026). We train and evaluate ReRoute. We also train and evaluate Backbone and all ablation variants for the ablation studies and robustness analysis. Backbone removes both SMMC and RRMR while retaining ReRoute's visual and textual token configurations.

D.3 EVALUATION PROTOCOL

We evaluate link prediction by scoring all candidate entities for head and tail queries and follow the filtered ranking protocol (Zhang et al., 2025b). Under the filtered setting, other known positive triples from the training, validation, and test sets are excluded when ranking the target entity (Bordes et al., 2013). We report MRR and Hits@K for $K \in \{1, 3, 10\}$ (Sun et al., 2019); MRR averages reciprocal target ranks, while Hits@K measures the proportion of targets ranked among the top K candidates. Results are averaged over head and tail prediction for all test triples. Let $\mathcal{T}_{\text{test}}$ denote the set of test triples, and let $r_{h,i}$ and $r_{t,i}$ denote the ranks of the ground-truth target entity in the head prediction and tail prediction of the i-th test triple, respectively. MRR and Hits@K are computed as

$$\text{MRR} = \frac{1}{2\,|\mathcal{T}_{\text{test}}|} \sum_{i=1}^{|\mathcal{T}_{\text{test}}|} \left(\frac{1}{r_{h,i}} + \frac{1}{r_{t,i}} \right), \tag{13}$$

$$\text{Hits@}K = \frac{1}{2\,|\mathcal{T}_{\text{test}}|} \sum_{i=1}^{|\mathcal{T}_{\text{test}}|} \Big(\mathbb{I}\left[r_{h,i} \le K\right] + \mathbb{I}\left[r_{t,i} \le K\right] \Big), \tag{14}$$

where $\mathbb{I}[\cdot]$ is the indicator function.

D.4 IMPLEMENTATION DETAILS

The entity encoder and triple context encoder have one and two Transformer layers, respectively. Each layer uses four attention heads. Dropout rates (Srivastava et al., 2014) are 0.9, 0.4, and 0.1 for structural, visual, and textual embeddings, respectively, and 0.01 for the Transformers. ReRoute, Backbone, and all ablation variants use maximum visual and textual token counts of 16 and 24, respectively. The cosine annealing schedule with warm restarts (Loshchilov & Hutter, 2017) uses an initial cycle length of 50 epochs and a cycle length multiplier of two. The gradient norm clipping threshold is 0.1.

D.5 MODEL EFFICIENCY

Table 4 reports model-level efficiency on MKG-Y, complementing the candidate scoring comparison in Section 5.6; RRMR denotes ReRoute without SMMC. On MKG-Y, ReRoute and RRMR have identical reported FLOPs and scoring memory, with similar training and inference times. Both routed models require more FLOPs, training time, and scoring memory than MyGO. Appendix D.6 details the candidate scoring benchmark used in Figure 6.

Table 4: Model efficiency on MKG-Y. Total training time is estimated for 3,500 epochs from the reported time per epoch.

Model	FLOPs (M/query)	Train (s/epoch)	Train (h)	Inference (μs/query)	Scoring Mem (MiB)
MyGO	50.80	3.12	3.03	80	8.40
RRMR	66.20	6.87	6.68	94	12.18
ReRoute	66.20	6.84	6.65	92	12.18

D.6 Candidate Scoring Benchmark

We benchmark candidate scoring on MKG-Y using an NVIDIA A100-PCIE-40GB GPU, CUDA 12.4, and PyTorch 2.5.1 in FP32. The explicit conditioned and decomposable scoring implementations use the same query vectors, relation–role gates, base representations, and compensated enhancement components, with query batch size $B = 32$ and representation dimension $d = 256$. The candidate count is varied over $N \in \{512, 1{,}024, 2{,}048, 4{,}096, 8{,}192, 15{,}000\}$, without candidate chunking. Optimized explicit conditioned scoring constructs one $B \times N \times d$ tensor of conditioned candidate representations using in-place updates, then computes batched inner products with the query vectors. Decomposable scoring instead computes the three $B \times N$ matching matrices in Equation (8) and applies the gates to the two modality-specific terms, avoiding the $B \times N \times d$ tensor.

Measurements cover candidate scoring only. Entity encoding and SMMC are computed once before measurement, and their outputs are cached on the GPU; query encoding, gate generation, and CPU–GPU transfers are excluded. After 10 warm-up runs, latency is averaged over 50 runs, with GPU synchronization immediately before and after each timed call, and divided by B to obtain the amortized time per query. Additional peak GPU memory usage is measured in a separate scoring call: after clearing the allocator cache, we record the allocated-memory baseline, reset peak-memory statistics, and subtract this baseline from the peak allocation during scoring. Thus, the reported memory excludes tensors already resident on the GPU.

With all $15{,}000$ candidates, decomposable scoring takes $5.60\,\mu$s per query and uses $12.18\,$MiB of additional peak GPU memory, compared with $98.4\,\mu$s and $470.58\,$MiB for optimized explicit conditioned scoring. These measurements yield the $17.6\times$ speedup and $1/38.6$ memory ratio reported in Section 5.6. The formulations are algebraically equivalent (Appendix C.2); in the FP32 scoring check, the maximum absolute score difference is 7.6×10^{-6}, and the top-k candidate sets agree for all checked queries at $k \in \{1, 10, 100\}$. Filtered evaluation over the full test set gives an absolute MRR difference of 1.3×10^{-9} and identical target ranks for 99.96% of queries.

D.7 SMMC Coverage

We measure SMMC coverage using one-hop neighborhoods in the undirected training graph and the original modality availability, without retraining. An entity is missing a modality if it has no available tokens for that modality; an available neighbor has at least one such token. Coverage is the proportion of entities missing a modality that have at least one available neighbor. Table 5 reports the two dataset–modality combinations that meet the SMMC activation threshold $\eta_m \geq \tau_{\text{miss}} = 0.1$: visual information on MKG-W and textual information on MKG-Y.

Table 5: SMMC coverage under the original modality availability. Missing rates use all entities as the denominator. Coverage and neighbor-count statistics use all entities missing the indicated modality, including those with no available neighbors.

Dataset	Modality	Missing entities (% of all entities)	Covered entities (% of missing entities)	Available neighbors		
				Mean	Median	Max.
MKG-W	Visual	6,041 (40.27%)	4,940 (81.77%)	2.54	1	2,089
MKG-Y	Textual	2,684 (17.89%)	2,355 (87.74%)	2.23	2	23

Most entities missing a modality for which SMMC is enabled have at least one available neighbor; those without one retain a zero enhancement component, as specified in Equation (3). On MKG-W, the maximum of 2,089 available neighbors contrasts with a median of one, indicating a skewed distribution. These statistics describe access to neighborhood evidence; its predictive contribution is evaluated by the ablations in Section 5.3.