# Beyond Embedding Fusion: LLM-Driven Structural Enhancement for Multimodal Knowledge Graph Completion

Anonymous ACL submission

## Abstract

Multimodal knowledge graph completion (MKGC) aims to improve structural reasoning by incorporating visual and textual information. However, existing approaches rely heavily on embedding fusion, where multimodal features are compressed and fused into a unified vector before structural prediction. This compression-then-fusion paradigm inevitably reduces the rich semantics carried by raw modalities, treating them as auxiliary cues rather than sources of explicit structural knowledge. As a result, current MKGC methods often fail to capture deeper relational semantics implied in texts and images. To address this limitation, we propose LLM-SE (**L**arge **L**anguage **M**odel–driven **S**tructural **E**nhancement), a generate-then-disentangle framework that transforms raw multimodal signals into explicit structural triplets instead of collapsing them into unified embeddings. LLM-SE includes two main modules: (1) Multimodal Triplet Generation, which performs the generation step by leveraging large multimodal models to extract meaningful triplets from texts and images; and (2) Dual-View Complex module, a disentanglement mechanism that separates origin triplets from LLM-generated deep triplets, enabling the model to adaptively capture stable and exploratory knowledge. Extensive experiments on multiple MKGC benchmarks show that LLM-SE consistently outperforms state-of-the-art models across all metrics. The code and data are available at https://anonymous.4open.science/r/LLM-SE.

## 1 Introduction

Knowledge graph (KG) completion is a pivotal area within the broader field of knowledge representation and reasoning. Since the inception of knowledge graphs, exemplified by Google's KG in 2012, they have become essential for structuring and interrelating vast amounts of data across domains (Liu et al., 2024; Xu et al., 2024). Knowledge graphs have numerous applications, including recommendation system (Wang et al., 2025) and natural language understanding (Liu et al., 2018; Jiang et al., 2025). However, the real-world KGs are usually incomplete because their source documents and web information are usually deficient. Knowledge graph completion addresses this challenge by predicting missing links and entities, thereby enhancing its effectiveness across various applications.

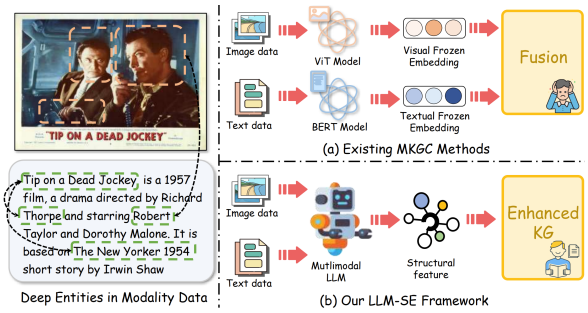



Figure 1: An intuition of existing MKGC methods and LLM-SE. Our method directly extracts deep information from multimodal features to enhance KG.

Multimodal knowledge graph completion (MKGC) extends the traditional KG completion task by incorporating additional entity-related modalities such as textual descriptions, images, and other auxiliary data sources. By leveraging these heterogeneous signals, MKGC aims to improve the prediction of missing entities and relations. This naturally raises a central research question: *How can multimodal information be more effectively exploited to enhance knowledge graph completion?*

As illustrated in Figure 1(a), existing MKGC approaches primarily follow a compression-then-fusion paradigm: multimodal information is compressed into frozen embeddings and subsequently fused into a unified representation. For example, OTKGE (Cao et al., 2022) employs optimal transport to align modalities into a unified embedding

space. VISTA (Lee et al., 2023) applies Transformer architectures in both encoder and decoder stages to fuse multimodal signals before structural learning. MyGO (Zhang et al., 2025b) proposes a fine-grained fusion mechanism that more tightly integrates visual and textual semantics into a joint embedding. Despite their architectural differences, these methods share a common limitation: multimodal information experiences partial loss through compression.

Therefore, we propose **LLM-SE**, a new generate-then-disentangle framework for multimodal knowledge graph completion. LLM-SE consists of two complementary components. The multimodal triplet generation module leverages large language models to generate deep triplets from visual and textual features, followed by deduplication and pruning operations to optimize KG. The dual-view complex module models triplets from different knowledge sources as two distinct views, enabling the model to differentiate varying levels of reliability and provenance.

Our main contributions are summarized as follows:

- To the best of our knowledge, this is the first work that departs from compression-then-fusion designs and introduces a generate-then-disentangle framework for MKGC task.
- We introduce a multimodal triplet generation module that employs LLMs to construct semantically rich deep triplets from visual and textual modalities.
- We propose a dual-view complex module that separates surface triplets from LLM-generated deep triplets through two distinct views, differentiating knowledge from different sources of credibility.
- We conduct extensive experiments on multiple benchmark datasets, demonstrating that LLM-SE consistently outperforms state-of-the-art methods.

## 2 Problem Formulation

A surface MKG $\mathcal{G}_{surf} = (\mathcal{E}, \mathcal{R}, \mathcal{T}, \mathcal{I}, \mathcal{D})$ consists of an entity set $\mathcal{E}$, a relation set $\mathcal{R}$, the tuples $\mathcal{T}$, a set of images $\mathcal{I}$, and a set of text descriptions $\mathcal{D}$. Each triplet in $\mathcal{T}$ is represented as $\mathcal{T} = \{(h, r, t) | h, t \in \mathcal{E}, r \in \mathcal{R}\}$. Entity $h$ is related to entity $t$ through relation $r$. Besides, $\mathcal{I}, \mathcal{D}$ correspond to the collections of images and textual descriptions for each entity $e$. The MKGC task aims to predict the missing component in the triplets, where the missing component can be either the head entity in the triplet $(h, r, ?)$ or the tail entity in the triplet $(?, r, t)$.

## 3 Methodology

As shown in Figure 2, the proposed LLM-SE framework comprises two key components: multimodal triplet generation and dual-view complex module.

### 3.1 Multimodal Triplet Generation

In order to obtain deeper knowledge, we carry out the following four steps: triplet extraction, relation alignment, graph deduplication, and graph pruning.

#### 3.1.1 Triplet Extraction

Triplet extraction serves as the core component of the structural enhancement process, transforming unstructured multimodal descriptions into explicit relational facts. We design a six-step prompting strategy that guides the LLM to gradually infer triplets. Specifically:

**(1) Main Entity Prioritization.** Deep triplets are constrained to include surface entities.

**(2) Joint Multimodal Reasoning.** Text is treated as the primary information source, while images are incorporated as complementary evidence.

**(3) Entity Formatting Constraints.** Entity names are required to be explicit and well-defined, avoiding vague expressions (e.g., "two people").

**(4) Textual and Visual Verifiability.** All entities must be verifiable through textual descriptions or visual evidence.

**(5) Relation Coherence.** Relations are restricted to well-defined expressions, with preference given to be selected from origin relations $\mathcal{R}$.

**(6) Category-Aware Image Reasoning.** We provide example of how to extract triplets from images of certain categories (e.g. poster).

We construct prompt $T_{extract}(\cdot)$ from the textual $D_e$ and visual information $I_e$ of the entity $e$:

$$(\hat{h}, \hat{r}, \hat{t})_n = LLM(T_{extract}(I_e, D_e, \mathcal{R})) \quad (1)$$

where $(\hat{h}, \hat{r}, \hat{t})_n$ denotes generated triplets. Detailed prompts are provided in Appendix D.

#### 3.1.2 Relation Alignment

After building the initial deep triplets, deep relations have many textual differences but are semantically one relation. We used a two-step LLM prompt

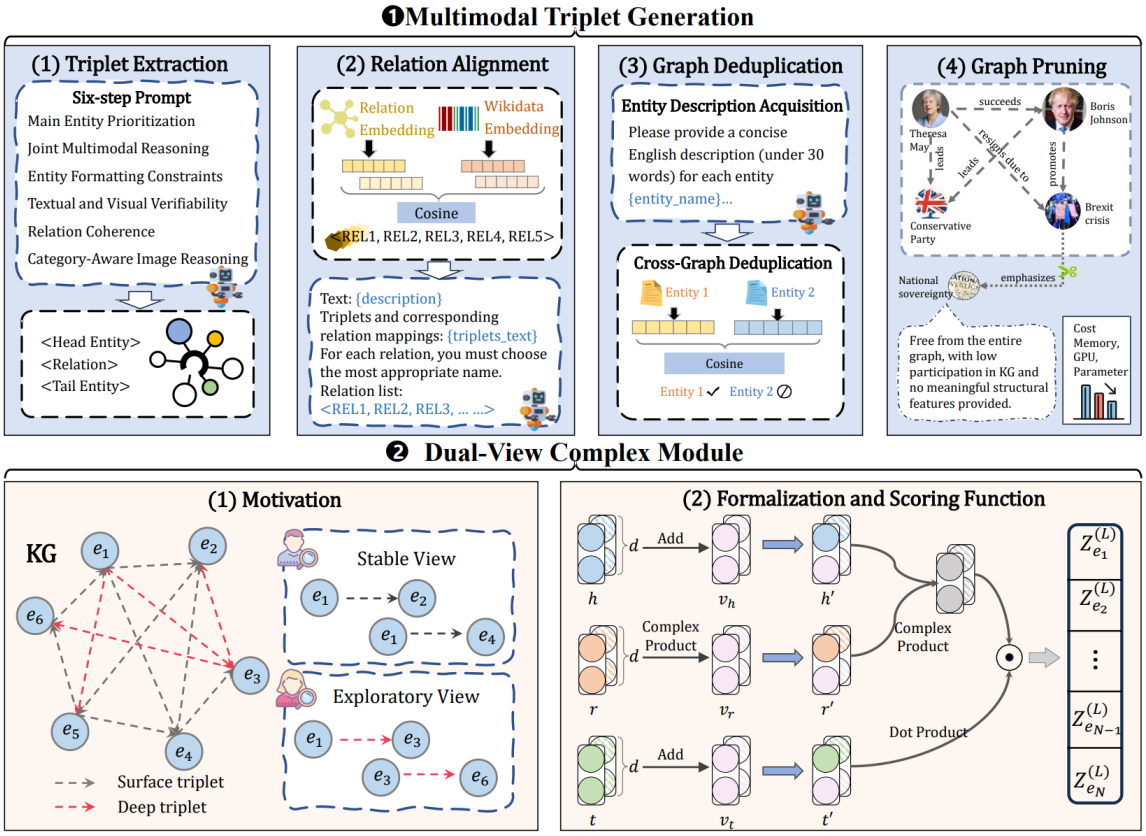



Figure 2: The framework of our proposed LLM-SE.

strategy. After extracting relation information from the text in the first step, we use FAISS index (Johnson et al., 2021) to retrieve specification relation names $\mathcal{R}_{wiki}$ from Wikidata KG and rank them based on cosine similarity. This index is built using pretrained Contriever embeddings (Izacard et al., 2022). We select the top-5 relations, denoted as $top5(\mathcal{R}_{wiki})$, that are most similar to $\hat{r}$. We construct these 5 candidate relations, entity descriptions, and triplets as prompts:

$$\bar{r} = LLM(T_{align}(D_e, \hat{h}, \hat{r}, \hat{t}, top5(\mathcal{R}_{wiki}))) \quad (2)$$

where $\bar{r}$ denotes the aligned representation of $\hat{r}$.

### 3.1.3 Graph Deduplication

This stage aims to refine the subgraphs extracted in the previous step into standardized and explicit forms, thereby integrating the separated subgraphs in the deep knowledge graph.

**Entity Description Acquisition.** Many entities contain only short surface forms, resulting in shallow semantic signals that hinder accurate graph merging. For example, entities such as "United States" and "USA" are lexically distant despite referring to the same concept. Therefore, we leverage recent advances in LLMs to generate concise, entity-focused descriptions. We design a prompt $T_{description}(\cdot)$ that guides the LLM to produce semantically informative text for each entity $e$.

$$\hat{D}_e = LLM(T_{description}(e)) \quad (3)$$

The resulting $\hat{D}_e$ enriched descriptions provide a stronger semantic basis for subsequent merging and alignment.

**Cross-Graph Deduplication.** The next step is to eliminate redundancy and integrate the surface KG with the deep KG. Redundancy may occur both within the deep graph and across the deep and surface graphs, as semantically equivalent entities can appear under different lexical forms. To this end, we perform similarity-based entity merging, where entities from the deep graph are deduplicated or aligned to their surface counterparts. Formally, the merging operation is triggered when the cosine similarity between two entity embeddings satisfies $\cos(\mathbf{e}_i, \mathbf{e}_j) \geq \lambda_e$:

$$\text{merge}(e_i, e_j) = \begin{cases} e_i, & e_i, e_j \in \mathcal{G}_{deep} \\ e_j, & e_i \in \mathcal{G}_{deep},\ e_j \in \mathcal{G}_{surf} \end{cases} \quad (4)$$

where $\mathbf{e}_i, \mathbf{e}_j$ denote the textual embeddings of entities, $\lambda_e$ is the entity threshold, $\mathcal{G}_{deep}$ represents the

deep KG, and $\cos(\cdot,\cdot)$ denotes the cosine similarity function. Correspondingly, we also use a similar threshold $\lambda_r$ and method to handle relations.

#### 3.1.4 Graph Pruning

The generation process introduces a large number of low-degree entities and triplets. Many of these newly create nodes had degree one and remain isolated near the periphery of the graph. During backpropagation, the representation updates of these newly generated entities are almost entirely determined by a single neighboring entity–relation pair, resulting in highly unstable and uninformative gradients. To mitigate this issue, we apply a degree-based pruning strategy and remove deep entities with degree one along with their associated deep triplets.

### 3.2 Dual-View Complex Module

The new knowledge graph $\mathcal{G}' = (\mathcal{E}', \mathcal{R}', \mathcal{T}')$ consists of a new entity set $\mathcal{E}'$, a relation set $\mathcal{R}'$, and the tuples $\mathcal{T}'$.

#### 3.2.1 Motivation

After augmenting the KG with LLM-generated triplets, the resulting graph becomes a mixture of reliable original triplets and potentially noisy LLM-generated ones. Therefore, we propose a dual-view complex module that disentangles knowledge from two distinct views according to their credibility. **Stable View:** Corresponding to the original, surface, and high-confidence triplets. **Exploratory View:** Corresponding to LLM-generated, deep, and lower-confidence triplets. Then, we identify three fundamental logical patterns that a robust model should capture:

**Property 1:** *The same entity pair may correspond to different relations under different views.*

**Property 2:** *Certain triples remain valid across different views.*

**Property 3:** *A triple valid in one view may become invalid in another.*

We show that the LLM-SE is theoretically capable of modeling these patterns (Appendix C.7).

#### 3.2.2 Formalization and Scoring Function

ComplEx (Trouillon et al., 2016) is a widely used and robust baseline model. Therefore, we choose to conduct research and development based on ComplEx. Let $\mathbf{h}, \mathbf{r}, \mathbf{t} \in \mathbb{C}^d$ be the complex embeddings of the head, relation, and tail, respectively, where $d$ is the embedding dimension. We split these complex vectors into their real and imaginary parts: $\mathbf{h} = (Re(\mathbf{h}), Im(\mathbf{h}))$, and similarly for $\mathbf{r}$ and $\mathbf{t}$.

View embedding $\mathbf{v} \in \mathbb{C}^{3d}$ is a high-dimensional vector that encodes the source (surface or deep) of a triplet. This vector is segmented to provide view specific adjustments for the head, tail, and relation representations: $\mathbf{v}_h, \mathbf{v}_r, \mathbf{v}_t \in \mathbb{C}^d$.

The head and tail embeddings are adjusted by adding a view-specific bias:

$$\begin{aligned}\mathbf{h}' &= (Re(\mathbf{h}) + Re(\mathbf{v}_h), Im(\mathbf{h}) + Im(\mathbf{v}_h)) \\ \mathbf{t}' &= (Re(\mathbf{t}) + Re(\mathbf{v}_t), Im(\mathbf{t}) + Im(\mathbf{v}_t))\end{aligned} \tag{5}$$

Additionally, the relation embedding is modulated by $\mathbf{v}_r$, which acts as a view-specific relation modulator. The view-specific relation representation $\mathbf{r}'$ is computed as a complex multiplication:

$$\begin{aligned}\mathbf{r}' = (&Re(\mathbf{r}) \odot Re(\mathbf{v}_r) - Im(\mathbf{r}) \odot Im(\mathbf{v}_r)), \\ &Re(\mathbf{r}) \odot Im(\mathbf{v}_r) + Im(\mathbf{r}) \odot Re(\mathbf{v}_r))\end{aligned} \tag{6}$$

where $\odot$ denotes the Hadamard product.

The score for a triplet $(h, r, t)$ under view $v$ is formulated as a modified complex inner product:

$$\begin{aligned}&\phi(h, r, t, v) \\ =&\left\langle (Re(\mathbf{h}') \odot Re(\mathbf{r}') - Im(\mathbf{h}') \odot Im(\mathbf{r}')), Re(\mathbf{t}') \right\rangle \\ +&\left\langle (Im(\mathbf{h}') \odot Re(\mathbf{r}') + Re(\mathbf{h}') \odot Im(\mathbf{r}')), Im(\mathbf{t}') \right\rangle\end{aligned} \tag{7}$$

where $\langle \cdot, \cdot \rangle$ denotes the dot product. The two parts constitute the standard ComplEx scoring function. For the loss function, mathematically, we express it as:

$$\mathcal{L} = -log(\frac{exp(\phi(h, r, t, v))}{\sum_{t' \neq t \cap t \in \mathcal{E}'} exp(\phi(h, r, t, v))}) \tag{8}$$

In addition, we use N3 (Lacroix et al., 2018) regularization and Adagrad optimizer to train our model for robust inference.

## 4 Experiments

### 4.1 Experiment Setup

**Dataset.** We conduct experiments on three publicly available MKG datasets: MKG-W (Xu et al., 2022), MKG-Y (Xu et al., 2022), and DB15K (Liu et al., 2019). Detailed information about the datasets used is provided in Table 1.

**Data Leakage.** During the filtering stage, we impose strict constraints by filtering only original knowledge to ensure that potentially noisy triples do not lead to false performance improvements.

Table 1: Statistic of Datasets.

| | $\mathcal{E}$ | $\mathcal{E}'$ | $\mathcal{R}$ | $\mathcal{R}'$ | #train | #train′ | #valid | #test | Image | Text |
|---|---|---|---|---|---|---|---|---|---|---|
| MKG-W | 15000 | 25404 | 169 | 1005 | 34196 | 119751 | 4276 | 4274 | 14463 | 14123 |
| MKG-Y | 15000 | 25482 | 28 | 980 | 21310 | 102541 | 2665 | 2663 | 14244 | 12305 |
| DB15K | 12842 | 21392 | 279 | 1611 | 79222 | 157207 | 9902 | 9904 | 12818 | 9078 |

Table 2: Comprehensive link prediction results across different datasets. The best results are in **bold**, and the second-best are underlined.

| Model | MKG-W | | | | MKG-Y | | | | DB15K | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MRR | Hits@1 | Hits@3 | Hits@10 | MRR | Hits@1 | Hits@3 | Hits@10 | MRR | Hits@1 | Hits@3 | Hits@10 |
| **Uni-modal KGC Methods** | | | | | | | | | | | | |
| **TransE** (Bordes et al., 2013) | 0.292 | 0.211 | 0.332 | 0.442 | 0.307 | 0.235 | 0.352 | 0.434 | 0.249 | 0.128 | 0.315 | 0.471 |
| **DistMult** (Yang et al., 2015) | 0.210 | 0.159 | 0.223 | 0.309 | 0.250 | 0.193 | 0.278 | 0.360 | 0.230 | 0.148 | 0.263 | 0.396 |
| **ComplEx** (Trouillon et al., 2016) | 0.249 | 0.191 | 0.267 | 0.367 | 0.287 | 0.223 | 0.321 | 0.409 | 0.275 | 0.184 | 0.316 | 0.454 |
| **RotatE** (Sun et al., 2019) | 0.337 | 0.268 | 0.367 | 0.467 | 0.350 | 0.291 | 0.384 | 0.453 | 0.293 | 0.179 | 0.361 | 0.500 |
| **PairRE** (Chao et al., 2021) | 0.344 | 0.282 | 0.367 | 0.460 | 0.320 | 0.255 | 0.358 | 0.439 | 0.311 | 0.216 | 0.359 | 0.493 |
| **Multi-modal KGC Methods** | | | | | | | | | | | | |
| **TBKGC** (Sergieh et al., 2018) | 0.315 | 0.253 | 0.340 | 0.432 | 0.340 | 0.305 | 0.353 | 0.401 | 0.284 | 0.156 | 0.370 | 0.499 |
| **TransAE** (Wang et al., 2019) | 0.300 | 0.212 | 0.349 | 0.447 | 0.281 | 0.253 | 0.291 | 0.330 | 0.281 | 0.213 | 0.312 | 0.412 |
| **IKRL** (Xie et al., 2017) | 0.324 | 0.261 | 0.348 | 0.441 | 0.332 | 0.304 | 0.343 | 0.383 | 0.268 | 0.141 | 0.349 | 0.491 |
| **OTKGE** (Cao et al., 2022) | 0.344 | 0.289 | 0.363 | 0.449 | 0.355 | 0.320 | 0.372 | 0.414 | 0.239 | 0.185 | 0.259 | 0.342 |
| **MMKRL** (Lu et al., 2022) | 0.301 | 0.222 | 0.341 | 0.447 | 0.368 | 0.317 | 0.398 | 0.453 | 0.268 | 0.139 | 0.351 | 0.494 |
| **IMF** (Li et al., 2023) | 0.345 | 0.288 | 0.366 | 0.454 | 0.358 | 0.330 | 0.371 | 0.406 | 0.323 | 0.242 | 0.360 | 0.482 |
| **VISTA** (Lee et al., 2023) | 0.329 | 0.261 | 0.354 | 0.456 | 0.305 | 0.249 | 0.324 | 0.415 | 0.304 | 0.225 | 0.336 | 0.459 |
| **AdaMF-MAT** (Zhang et al., 2024b) | 0.359 | 0.290 | 0.390 | 0.484 | 0.386 | 0.343 | 0.406 | 0.458 | 0.351 | 0.253 | 0.411 | 0.529 |
| **MyGO** (Zhang et al., 2025b) | 0.361 | 0.298 | 0.385 | 0.478 | 0.384 | 0.350 | 0.398 | 0.442 | 0.377 | 0.301 | 0.413 | 0.522 |
| **MoMoK** (Zhang et al., 2025a) | 0.359 | 0.304 | 0.375 | 0.461 | 0.379 | 0.351 | 0.392 | 0.432 | 0.396 | 0.324 | 0.435 | 0.541 |
| **NativE** (Zhang et al., 2024a) | 0.366 | 0.296 | - | - | 0.390 | 0.348 | - | - | 0.372 | 0.280 | - | 0.541 |
| **LLM-SE (ours)** | **0.477** | **0.405** | **0.519** | **0.607** | **0.506** | **0.446** | **0.542** | **0.613** | **0.441** | **0.351** | **0.488** | **0.604** |
| *improve*$\Delta$ | 30.3% | 33.2% | 33.1% | 25.4% | 29.7% | 27.1% | 33.5% | 33.8% | 11.4% | 8.3% | 12.2% | 11.6% |

Moreover, the knowledge encoded in the LLM and the provided multimodal input does not constitute data leakage, as no external supervision from the evaluation sets is introduced.

**Baselines.** For unimodal KGC methods, we adopt 5 representative models: TransE (Bordes et al., 2013), DistMult (Yang et al., 2015), ComplEx (Trouillon et al., 2016), RotatE (Sun et al., 2019), PairRE (Chao et al., 2021). For multimodal KGC methods, we employ 11 competitive models, including TBKGC (Sergieh et al., 2018), TransAE (Wang et al., 2019), IKRL (Xie et al., 2017), OTKGE (Cao et al., 2022), MMKRL (Lu et al., 2022), IMF (Li et al., 2023), VISTA (Lee et al., 2023), AdaMF-MAT (Zhang et al., 2024b), MyGO (Zhang et al., 2025b), MoMoK (Zhang et al., 2025a), and NativE (Zhang et al., 2024a).

**Parameters Setting.** Learning rates are set as 1e-1. Batch sizes are set as 1024. The entity and relation thresholds $\lambda_e, \lambda_r$ are set to 0.8 and 0.7, respectively. The embedding dimension is fixed at 512. The LLM we used is Qwen-VL-235B-A22B-Instruct. The best model is selected via early stopping on the validation set. We show the details of the evaluation protocol in Appendix B.

## 4.2 Main Results

From Table 2, LLM-SE consistently outperforms all competing methods across the three datasets, yielding substantial gains in all evaluation metrics.

For uni-modal KGC baselines, their performance is generally inferior to multimodal approaches. In most datasets, the multimodal variants outperform their uni-modal counterparts, indicating that multimodal information provides complementary signals. Recent MKGC methods remain suboptimal due to excessive information loss introduced by aggressive feature compression. For example, VISTA (Lee et al., 2023) extends multimodal fusion to include relational modalities and employs Transformer encoders; Similarly, MyGO (Zhang et al., 2025b) introduces fine-grained multimodal fusion with contrastive training strategies. But all of them still operate on heavily compressed embeddings. In contrast, LLM-SE achieves superior performance by adopting a novel generation-then-disentangle paradigm, yielding substantial improvements on MKG-W and MKG-Y (30%), and relatively smaller gains on DB15K (10%). This discrepancy arises because DB15K contains fewer textual and visual features, resulting in a limited number of generated deep triplets.

Table 3: Results of ablation study. The best results are presented in boldface.

| Category | MKG-W | | | | MKG-Y | | | | DB15K | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MRR | Hits@1 | Hits@3 | Hits@10 | MRR | Hits@1 | Hits@3 | Hits@10 | MRR | Hits@1 | Hits@3 | Hits@10 |
| (1) w/o Image | 0.462 | 0.391 | 0.498 | 0.590 | 0.493 | 0.435 | 0.529 | 0.592 | 0.436 | 0.349 | 0.482 | 0.595 |
| (2) w/o Text | 0.378 | 0.319 | 0.402 | 0.489 | 0.393 | 0.356 | 0.414 | 0.459 | 0.408 | 0.326 | 0.448 | 0.560 |
| (3) w/o CGD | 0.472 | 0.402 | 0.507 | 0.599 | 0.422 | 0.365 | 0.441 | 0.510 | 0.407 | 0.323 | 0.448 | 0.562 |
| (4) w/o Pruning | 0.471 | 0.396 | 0.510 | 0.597 | 0.501 | 0.440 | 0.534 | 0.608 | 0.433 | 0.345 | 0.478 | 0.595 |
| (5) w/o DVC | 0.431 | 0.357 | 0.474 | 0.558 | 0.442 | 0.367 | 0.495 | 0.562 | 0.420 | 0.324 | 0.475 | 0.592 |
| **LLM-SE (ours)** | **0.477** | **0.405** | **0.519** | **0.607** | **0.506** | **0.446** | **0.542** | **0.613** | **0.441** | **0.351** | **0.488** | **0.604** |

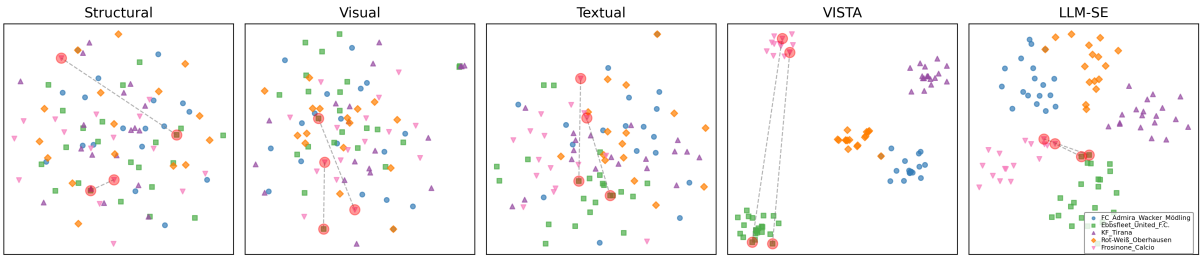



Figure 3: Visualization of low-dimensional player embeddings. Each node represents a player, and the five colors correspond to different team affiliations. The red hollow circles mark the some players, and the gray dashed lines indicate the distances between them.

### 4.3 Ablation Study

To further evaluate the effectiveness of LLM-SE and its components, we conduct a systematic ablation study, as shown in Table 3. Experiments (1) and (2) remove the image and text modalities, respectively. Performance drops in both cases, with a notably larger degradation when textual features are removed, indicating the dominant role of text in these datasets. Experiment (3) removes the Cross-Graph Deduplication (CGD) module, resulting in substantial performance declines on MKG-Y and DB15K, but only a minor drop on MKG-W. This is attributed to the more consistent entity semantics generated by the LLM on MKG-W, which reduces the reliance on deduplication. Experiment (4) shows that removing graph pruning causes moderate performance degradation across all datasets, suggesting that pruning deep entities not only reduces model size but also benefits performance. Finally, Experiment (5) replaces the dual-view complex module (DVC) with ComplEx, leading to an approximately 10% performance drop, which highlights the effectiveness of dual-view modeling.

## 5 Discussion

### 5.1 Entity Embedding Visualization

To further assess the quality of the learned representations, we visualize the embeddings of football players using t-SNE on the MKG-Y dataset, as shown in Figure 3. The first three subfigures illustrate the structural, visual, and textual patterns. The fourth and fifth subfigures present the representations produced by the VISTA and LLM-SE.

For a single modality, players from different teams tend to appear in overlapping regions, making team boundaries difficult to discern. In contrast, both VISTA and LLM-SE draw players from the same team closer together and more clearly separate players from different teams, demonstrating their effectiveness in multimodal information integration. However, the clusters generated by VISTA, while internally compact, are excessively distant from each other, resulting excessive separation that does not reflect real-world continuity. For example, the red hollow circles in the figure highlight players from two teams who share the same on-field position (guards). Their positional roles imply that they should not be placed far apart solely based on team affiliation. In this case, LLM-SE produces more reasonable representations than VISTA, yielding a distribution that better reflects the underlying semantic relationships.

### 5.2 Effectiveness of the Multimodal Triplet Generation Module

To verify the effectiveness of the multimodal triplet generation (MTG) module, we conduct experiments on several uni-modal baseline models using the enhanced knowledge graph dataset. We evaluate and compare the performance of each baseline model before and after integrating MTG while

Table 4: Results of different models with multimodal triplet generation. The best results are in bold.

| Model | MKG-W | | | | MKG-Y | | | | DB15K | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MRR | Hits@1 | Hits@3 | Hits@10 | MRR | Hits@1 | Hits@3 | Hits@10 | MRR | Hits@1 | Hits@3 | Hits@10 |
| **LLM-SE (ours)** | **0.477** | **0.405** | **0.519** | **0.607** | **0.506** | **0.446** | **0.542** | **0.613** | **0.441** | **0.351** | **0.488** | **0.604** |
| ComplEx (w/ MTG) | 0.431 | 0.357 | 0.474 | 0.558 | 0.442 | 0.367 | 0.495 | 0.562 | 0.420 | 0.324 | 0.475 | 0.592 |
| ComplEx | 0.249 | 0.191 | 0.267 | 0.367 | 0.287 | 0.223 | 0.321 | 0.409 | 0.275 | 0.184 | 0.316 | 0.454 |
| DistMult (w/ MTG) | 0.426 | 0.347 | 0.471 | 0.564 | 0.425 | 0.341 | 0.488 | 0.557 | 0.405 | 0.311 | 0.475 | 0.595 |
| DistMult | 0.210 | 0.159 | 0.223 | 0.309 | 0.250 | 0.193 | 0.278 | 0.360 | 0.230 | 0.148 | 0.263 | 0.396 |
| RotatE (w/ MTG) | 0.430 | 0.347 | 0.483 | 0.567 | 0.435 | 0.351 | 0.501 | 0.569 | 0.398 | 0.299 | 0.468 | 0.581 |
| RotatE | 0.337 | 0.268 | 0.367 | 0.467 | 0.350 | 0.291 | 0.384 | 0.453 | 0.293 | 0.179 | 0.361 | 0.500 |

keeping all parameter configurations unchanged. As shown in Table 4, MTG consistently improves performance across multiple datasets, demonstrating its effectiveness and generality.

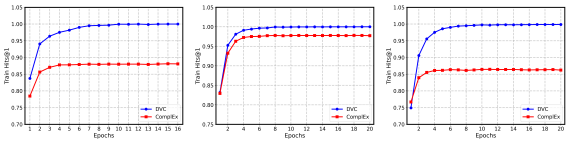
Specifically, MTG enhances all evaluation metrics for the three decoder baselines on all datasets. Notably, the MRR metric exhibits substantial gains. For example, on the MKG-W dataset, OTKGE improves ComplEx from 0.249 to 0.344 through multimodal feature fusion, whereas directly enhancing structural features with LLM-generated triplets further boosts performance to 0.431. The considerable improvements across different decoders indicate that enriching structural information rather than compressing multimodal features may provide a more effective solution for MKGC task.

### 5.3 Training Dynamics under Dual-View Mechanism

To verify the effectiveness of the proposed DVC module, we analyze the training dynamics of ComplEx and DVC on enhanced KGs, as shown in Figures 4 and 5. We observe that the ComplEx model fails to stably fit the training data on all three datasets enhanced with deep triplets, as the training metric Hits@1 stops improving before reaching 100%. This is because ComplEx assumes a single-view semantic space, while the enhanced KGs contain inherently conflicting triples, which violates the three logical properties in Section 3.2.1. In contrast, DVC introduces a dual-view mechanism that explicitly disentangles stable knowledge from exploratory information. Moreover, evaluations on the test set further demonstrate that this improved training behavior does not lead to overfitting, but instead yields stronger adaptability and robustness in the presence of noisy data.

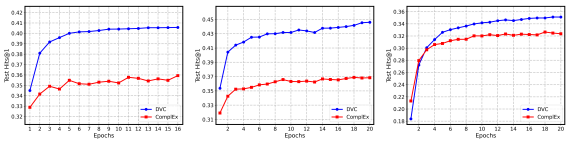
### 5.4 Case Study

To validate how our LLM-driven data augmentation optimizes reasoning, we conduct a detailed case study comparing LLM-SE and NativE in Table 5.



(a) MKG-W (b) MKG-Y (c) DB15K

Figure 4: Comparison on the Hits@1 vs. epochs of DVC and ComplEx in train datasets.



(a) MKG-W (b) MKG-Y (c) DB15K

Figure 5: Comparison on the Hits@1 vs. epochs of DVC and ComplEx in test datasets.

**Case 1: (The Belly of an Architect, cast member, ?)**

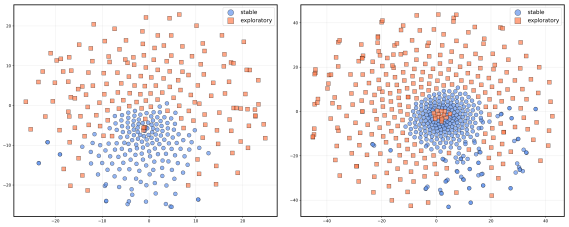
The baseline NativE struggles with data sparsity, forced to traverse a 4-hop path involving irrelevant nodes like *"United Kingdom"* to reach the target *"Brian Dennehy"*. In contrast, LLM-SE identifies a direct 1-hop connection. This indicates that our LLM-driven module successfully extracted the explicit triplet (*The Belly of an Architect*, *cast_member*, *Brian Dennehy*) directly from the movie's description and injected this missing edge into the graph. By turning a potential multi-hop reasoning problem into a simple 1-hop retrieval, LLM-SE significantly improves accuracy.

**Case 2: (The Wild, composer, ?)**

LLM-SE infers the composer *"Alan Silvertri"* via a highly plausible collaborative chain: *"The Wild"* shares the lead actor *"Kiefer Sutherland"* with the movie *"Young Guns II"*, which was composed by *"Alan Silvertri"*. This suggests a strong latent correlation in the production process. Conversely, NativE generates a long path by relying on the generic high-degree node *"film"*. Relying on

Table 5: Comparison between LLM-SE and NativE on different queries.

| Query | LLM-SE | | NativE | |
|---|---|---|---|---|
| | Correct entity ranking position & Top-5 ranked candidate entities | Shortest_path & Hops | Correct entity ranking position & Top-5 ranked candidate entities | Shortest_path & Hops |
| (The Balley of an Architect, cast member, ?) Ground Truth: Brian Dennehy | 1, (Brian Dennehy ✓, Peter Greenaway, Chloe Grace Moretz, John Gielgud, Ciaran Hinds) | The Balley of an Architect → Brian Dennehy | 12, (Peter Greenaway, John Gielgud, Ciaran Hinds, Richard Bohringer, Michael Gambon) | The Balley of an Architect → United Kindom → Tony Blair → Yale University → Brian Dennehy |
| (The Wild, composer, ?) Ground Truth: Alan Silvertri | 3, (Ed Dector, John Debney, Alan Silvertri ✓, David Newman, Randy Edelman) | The Wild → Kiefer Sutherland → Young Guns II → Alan Silvertri | 17, (John Debney, Randy Edelman, David Newman, Christophe Beck, Danny Elfman) | The Wild → Ed Dector → There's Something About Mary → film → The Quick and the Dead → Alan Silvertri |




(a) MKG-W (b) DB15K

Figure 6: Visual representation of relation embedding from different views and datasets.

the broad concept of *"film"* creates a weak semantic link, whereas LLM-SE leverages strong entity-to-entity correlations enriched by our enhancement.

Across all the cases, LLM-SE consistently outperforms NativE, ranking correct entities higher and uncovering shorter reasoning paths.

### 5.5 Dual-View Effect on Relation Embeddings

To investigate the impact of the dual-view mechanism, we visualize embeddings of the original relations from the MKG-W and DB15K datasets under two distinct views. As illustrated in Figure 6, we apply t-SNE to project the embeddings into a low-dimensional space, where blue circles denote relations in the stable view and orange squares denote relations in the exploratory view. Our relation embeddings $\mathbf{r}$ are combined with two views $\mathbf{v}_r$ via complex multiplication, corresponding to distinct rotations and scalings in each view (Trouillon et al., 2016). Consequently, for DB15K, some relations remain close under both views (central), indicating that the dual-view transformations induce consistent shifts, while others are separated (periphery), reflecting adaptive shifts. This indicates that the dual-view mechanism introduces controllable semantic shifts, adaptively allocating representational capacity to balance stability and exploration.

Table 6: Resource consumption comparison among LLM-SE, NativE and MoMoK.

| Metrics | LLM-SE | NativE | MoMoK |
|---|---|---|---|
| 1. Trainable Parameters (M) ↓ | 28.07 | 39.96 | 86.96 |
| 2. GPU Memory (GB) ↓ | 0.75 | 9.64 | 3.65 |
| 3. TPS ↑ | 15363.25 | 2242.82 | 8012.70 |
| 4. GFLOPS ↓ | 0.83 | 1.81 | 17.84 |

### 5.6 Resource Consumption

To systematically evaluate the computational efficiency of our proposed LLM-SE framework, we conduct a comparative analysis with two baseline models: NativE and MoMoK. Specifically, we evaluate the number of trainable parameters, GPU memory consumption, transactions per second (TPS), and GFLOPS.

As shown in Table 6, LLM-SE demonstrates superior resource efficiency, achieving the lowest consumption across all four evaluated resources while also delivering the fastest runtime. This lightweight characteristic stems from the architectural design of LLM-SE, which models structural information in a uni-modal manner, thereby avoiding the substantial computational overhead typically introduced by multimodal fusion during training and inference.

## 6 Conclusions

In this work, we propose a simple yet effective approach that departs from conventional compression-then-fusion designs and introduces a generate-then-disentangle framework for MKGC. Instead of compressing heterogeneous modalities into unified embeddings, our method directly transforms multimodal data into structured representations, thereby explicitly performing knowledge generation at the structural level. LLM-SE directly enhances MKGC performance and achieve state-of-the-art results on three benchmark datasets.

## Limitations

Although our framework achieves strong performance, it is not without limitations. First, the use of LLMs introduces overhead. Because we invoke the LLM once for each existing entity, KGs with a large number of entities require substantial token consumption, which increases computational cost. Second, the effectiveness of our approach is influenced by domain specificity. When the multimodal information of a KG differ substantially from the corpora on which the LLM was pre-trained, performance may degrade. This often requires manual intervention, such as incorporating domain-specific terminology into in-context examples to improve LLM understanding.

## References


Antoine Bordes, Nicolas Usunier, Alberto Garcia-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems*, page 2787–2795.

Zongsheng Cao, Qianqian Xu, Zhiyong Yang, Yuan He, Xiaochun Cao, and Qingming Huang. 2022. OTKGE: multi-modal knowledge graph embeddings via optimal transport. In *Advances in Neural Information Processing Systems*.

Linlin Chao, Jianshan He, Taifeng Wang, and Wei Chu. 2021. PairRE: Knowledge graph embeddings via paired relation vectors. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 4360–4369.

Xiang Chen, Ningyu Zhang, Lei Li, Shumin Deng, Chuanqi Tan, Changliang Xu, Fei Huang, Luo Si, and Huajun Chen. 2022. Hybrid transformer with multi-level fusion for multimodal knowledge graph completion. In *Proceedings of International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 904–915. ACM.

Tim Dettmers, Minervini Pasquale, Stenetorp Pontus, and Sebastian Riedel. 2018. Convolutional 2D knowledge graph embeddings. In *Proceedings of AAAI Conference on Artificial Intelligence*, pages 1811–1818.

Alberto García-Durán and Mathias Niepert. KBlrn: End-to-end learning of knowledge base representations with latent, relational, and numerical features. In *Proceedings of Conference on Uncertainty in Artificial Intelligence*, pages 372–381.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*, 2022.

Jinhao Jiang, Kun Zhou, Xin Zhao, Yang Song, Chen Zhu, Hengshu Zhu, and Ji-Rong Wen. 2025. Kg-agent: An efficient autonomous agent framework for complex reasoning over knowledge graph. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pages 9505–9523.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2021. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547.

Timothée Lacroix, Nicolas Usunier, and Guillaume Obozinski. 2018. Canonical tensor decomposition for knowledge base completion. In *International Conference on Machine Learning*, volume 80, pages 2869–2878.

Jaejun Lee, Chanyoung Chung, Hochang Lee, Sungho Jo, and Joyce Jiyoung Whang. 2023. VISTA: visual-textual knowledge graph representation learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7314–7328.

Xinhang Li, Xiangyu Zhao, Jiaxing Xu, Yong Zhang, and Chunxiao Xing. 2023. IMF: interactive multimodal fusion model for link prediction. In *Proceedings of International Conference on World Wide Web*, pages 2572–2580.

Chunyu Liu, Wei Wu, Siyu Wu, Lu Yuan, Rui Ding, Fuhui Zhou, and Qihui Wu. 2024. Social-enhanced explainable recommendation with knowledge graph. *IEEE Transactions on Knowledge and Data Engineering*, 36(2):840–853.

Ye Liu, Hui Li, Alberto García-Durán, Mathias Niepert, Daniel Oñoro-Rubio, and David S. Rosenblum. 2019. MMKG: multi-modal knowledge graphs. In *Proceedings of the 16th Extended Semantic Web Conference (ESWC)*, volume 11503 of *Lecture Notes in Computer Science*, pages 459–474.

Zhenghao Liu, Chenyan Xiong, Maosong Sun, and Zhiyuan Liu. 2018. Entity-duet neural ranking: Understanding the role of knowledge graph semantics in neural information retrieval. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 2395–2405.

Xinyu Lu, Lifang Wang, Zejun Jiang, Shichang He, and Shizhong Liu. 2022. MMKRL: A robust embedding approach for multi-modal knowledge graph representation learning. *Applied Intelligence*, 52(7):7480–7497.

Hatem Mousselly Sergieh, Teresa Botschen, Iryna Gurevych, and Stefan Roth. 2018. A multimodal translation-based approach for knowledge graph representation learning. In *Proceedings of the 7th Joint Conference on Lexical and Computational Semantics*, pages 225–234.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. RotatE: Knowledge graph embedding

by relational rotation in complex space. In *Proceedings of the 7th International Conference on Learning Representations*.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. In *Proceedings of the 33rd International Conference on Machine Learning*, page 2071–2080.

Meng Wang, Sen Wang, Han Yang, Zheng Zhang, Xi Chen, and Guilin Qi. 2021. Is visual context really helpful for knowledge graph? A representation learning perspective. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 2735–2743.

Shijie Wang, Wenqi Fan, Yue Feng, Shanru Lin, Xinyu Ma, Shuaiqiang Wang, and Dawei Yin. 2025. Knowledge graph retrieval-augmented generation for llm-based recommendation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics*, pages 27152–27168.

Xintao Wang, Qianyu He, Jiaqing Liang, and Yanghua Xiao. 2022. Language models as knowledge embeddings. In *Proceedings of the 31st International Joint Conference on Artificial Intelligence*, pages 2291–2297.

Zhen Wang, Jianwen Zhang, Jianlin Feng, and Zheng Chen. 2014. Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the 28th Conference of the Association for the Advancement of Artificial Intelligence*, pages 1112–1119.

Zikang Wang, Linjing Li, Qiudan Li, and Daniel Zeng. 2019. Multimodal data enhanced representation learning for knowledge graphs. In *Proceedings of the International Joint Conference on Neural Networks*, pages 1–8.

Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. 2016. Representation learning of knowledge graphs with entity descriptions. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, page 2659–2665.

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

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

Ning Xu, Yifei Gao, An-An Liu, Hongshuo Tian, and Yongdong Zhang. 2024. Multi-modal validation and domain interaction learning for knowledge-based visual question answering. *IEEE Transactions on Knowledge and Data Engineering*, 36(11):6628–6640.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding entities and relations for learning and inference in knowledge bases. In *Proceedings of the 3rd International Conference on Learning Representations*.

Shuai Zhang, Yi Tay, Lina Yao, and Qi Liu. 2019. Quaternion knowledge graph embeddings. In *Proceedings of the 33rd Annual Conference on Neural Information Processing Systems*, pages 2731–2741.

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

Yichi Zhang, Zhuo Chen, Lingbing Guo, Yajing Xu, Binbin Hu, Ziqi Liu, Wen Zhang, and Huajun Chen. 2025a. Multiple heads are better than one: Mixture of modality knowledge experts for entity representation learning. In *Proceedings of the 13th International Conference on Learning Representations*.

Yichi Zhang, Zhuo Chen, Lingbing Guo, Yajing Xu, Binbin Hu, Ziqi Liu, Wen Zhang, and Huajun Chen. 2025b. Tokenization, fusion, and augmentation: Towards fine-grained multi-modal entity representation. In *Proceedings of the 39th Conference of the Association for the Advancement of Artificial Intelligence*, pages 13322–13330.

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

Yu Zhao, Xiangrui Cai, Yike Wu, Haiwei Zhang, Ying Zhang, Guoqing Zhao, and Ning Jiang. 2022. MoSE: Modality split and ensemble for multimodal knowledge graph completion. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 10527–10536.

# A Related Work

In this section, we respectively introduce embedding methods for static and multimodal knowledge graph.

## A.1 Static Knowledge Graph Methods

TransE (Bordes et al., 2013) modeled each relation as a translation vector that mapped the head entity embedding close to the tail entity embedding, i.e., $h + r \approx t$. To alleviate the limitations of representing all relations in a single space, TransH (Wang et al., 2014) projected entity embeddings onto relation-specific hyperplanes. DistMult (Yang

et al., 2015) adopted a simple bilinear scoring function, enabling efficient modeling of relation semantics through multiplicative interactions. ComplEx (Trouillon et al., 2016) further extended this formulation into the complex-valued space, allowing the model to naturally capture both symmetric and antisymmetric relational patterns. RotatE (Sun et al., 2019) represented relations as rotations in the complex plane, effectively generalizing translational assumptions. QuatE (Zhang et al., 2019) further generalized this idea by embedding entities and relations in quaternion space. ConvE (Dettmers et al., 2018) introduced convolutional neural networks to model interactions between entity and relation embeddings.

### A.2 Multimodal Knowledge Graph Methods

KBLRN (García-Durán and Niepert) represented one of the earliest attempts by jointly modeling relational, latent, and numerical information. DKRL (Xie et al., 2016) leveraged textual descriptions alongside structural information, while IKRL (Xie et al., 2017) incorporated visual features into entity representations. TransAE (Wang et al., 2019) extended TransE with a multimodal autoencoder to learn joint representations across modalities. MoSE (Zhao et al., 2022) highlighted the issue of modality conflicts and addressed it by maintaining modality-specific representations followed by ensemble-based prediction. RSME (Wang et al., 2021) introduced a learnable gating mechanism to regulate the influence of visual contextual information. IMF (Li et al., 2023) proposed a two-stage fusion strategy that exploited cross-modal complementarity while preserving modality-specific knowledge. VISTA (Lee et al., 2023) enriched multimodal KGs by incorporating relational-level visual and textual information into a unified heterogeneous encoding framework. NativE (Zhang et al., 2024a) proposed a framework for MKGC that introduced a relation-guided dual adaptive fusion module for flexible multi-modal fusion and a collaborative modality adversarial training strategy to augment missing or underrepresented modalities. MyGO (Zhang et al., 2025b) proposed a framework that tokenized fine-grained multi-modal entity information and encoded it with a cross-modal entity encoder, further enhancing representations via fine-grained contrastive learning to capture detailed semantic interactions. MoMoK (Zhang et al., 2025a) leveraged relation-guided modality knowledge experts to learn adaptive, relation-aware multi-modal entity representations, integrating predictions from multiple modalities.

## B Evaluation Protocol

In this experiment, we evaluate our method using four widely adopted evaluation metrics: Mean Reciprocal Rank (MRR), Hits@1, Hits@3, and Hits@10. The MRR metric measures the average reciprocal rank of the correct entity during link prediction, with a higher value indicating better performance. Meanwhile, Hits@n quantifies the proportion of correct entities present within the top-$n$ ranked predictions. MRR and Hits@n can be calculated as:

$$\textbf{MRR} = \frac{1}{|\mathcal{T}_{test}|} \sum_{i=1}^{|\mathcal{T}_{test}|} \left(\frac{1}{r_{h,i}} + \frac{1}{r_{t,i}}\right) \tag{9}$$

$$\textbf{Hits@n} = \frac{1}{|\mathcal{T}_{test}|} \sum_{i=1}^{|\mathcal{T}_{test}|} \left(\mathbf{1}(r_{h,i} \leq n) + \mathbf{1}(r_{t,i} \leq n)\right) \tag{10}$$

where $r_{h,i}$ and $r_{t,i}$ are the results of head prediction and tail prediction respectively, $\mathcal{T}_{test}$ is the test triple set.

## C Supplementary Discussion

Due to space limitations in the main paper, we provide a more detailed discussion of additional analyses in this section. Specifically, we organize our investigation around 9 research questions (RQs):

**RQ1:** How do entity and relation thresholds affect cross-graph deduplication? (Threshold Sensitivity of Cross-Graph Deduplication)

**RQ2:** How does LLM-SE compare with finetuning-based MKGC methods? (Analysis of Finetuned Pretrained-Model Approaches)

**RQ3:** How does multimodal triplet generation benefit entities with different degrees in MKGC? (Case Study on Entity Degree Effects)

**RQ4:** Why are degree-one deep entities selectively pruned during graph pruning? (Rational for Degree-one Graph Pruning)

**RQ5:** How does graph pruning reduce the scale of deep knowledge graphs? (Impact of Graph Pruning on Graph Scale)

**RQ6:** How does the choice of LLM backbone affect multimodal triplet generation? (Impact of Different LLMs)

**RQ7:** Can the proposed dual-view mechanism theoretically model different logical view patterns? (Theoretical Properties)

**RQ8:** How robust is the dual-view complex module to noise introduced by multimodal triplets? (Noise Robustness)

**RQ9:** How robust is LLM-SE to missing multimodal information? (Missing Robustness)

### C.1 Threshold Sensitivity of Cross-Graph Deduplication

In Section 3.1.3, we introduced the cross-graph deduplication, where thresholds $\lambda_e, \lambda_r$ are used to merge semantically identical entities and relations. As illustrated in Figures 7 and 8, we examine how the thresholds influence the number of deep entities and relations, model performance, and the number of generated triplets on the MKG-Y dataset.

We first analyze the changes in the number of entities and relations. As expected, both increase as the threshold grows. We then evaluate the effect of the threshold on MRR and the number of deep triplets. The $\lambda_r$ has minimal impact on both metrics. When it exceeds 0.5, adjusting its value alters MRR by no more than 3%, and its influence on the number of deep triplets remains limited. This is likely because relations are already aligned before the graph deduplication stage. Next, we focus on $\lambda_e$. By jointly analyzing MRR and the quantity of deep triplets, we observe that model performance peaks when $\lambda_e$ is set to 0.8, which aligns with intuition. A threshold that is too low merges many semantically dissimilar entities, severely damaging structural information and reducing performance. Conversely, an excessively high threshold produces many sparse entities that are subsequently removed during graph pruning, decreasing the number of deep triplets and ultimately harming performance. In addition, thresholds associated with higher MRR generally correspond to a larger number of generated triplets, revealing a strong positive relationship between the two.

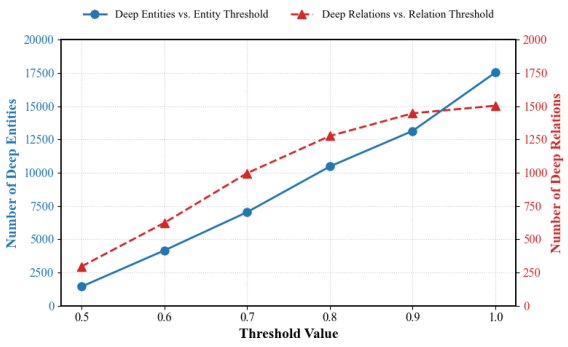
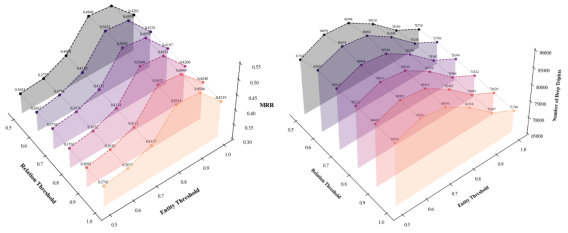
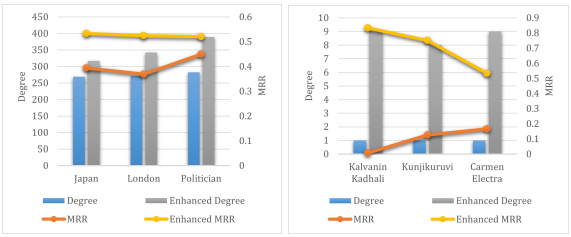



Figure 7: The number of deep entities and relations vary with threshold changes.



(a) MRR performance (b) Number of deep triplets

Figure 8: MRR and the number of deep triplets vary with threshold changes.




(a) Entities with high degree (b) Entities with low degree

Figure 9: Case study on entities with different degrees. The left y-axis denotes the entity degree, and the right y-axis denotes the MRR.

### C.2 Analysis of Finetuned Pretrained-Model Approaches

A distinct line of research in MKGC involves fine-tuning pretrained models to perform link prediction. Representative methods such as LMKE (Wang et al., 2022) and MKGformer (Chen et al., 2022) adopt pretrained language models, yet we do not include them in our main comparison because they are not evaluated on the three standard MKG datasets. LMKE leverages only textual modality and employs contrastive learning with pretrained language models, whereas MKGformer incorporates both textual and visual modalities by injecting triplet-level structural information into two pretrained models. However, MKGformer still yields suboptimal performance. In contrast, our LLM-SE framework uses a frozen large language model and does not require updating pretrained parameters. This distinction is critical: both LMKE and MKGformer must fine-tune pretrained models, which is computationally demanding and time-consuming. LLM-SE provides a more computationally efficient alternative to MKGC methods.

### C.3 Case Study on Entity Degree Effects

To further investigate how LLMs generate multimodal triplets to enhance MKGC, we conduct case studies on entities with different degrees. Using the MKG-W dataset, we sample three high-degree and three low-degree entities, as shown in Figure 9(a) and 9(b), respectively. For a fair analysis of the multimodal triplet generation (MTG) module, we adopt the standard ComplEx instead of DVC in all experiments.

In Figure 9(a), even for entities that originally have a large number of connections, MTG further increases their degrees and improves prediction accuracy. In addition, Figure 9(b) shows that entities with very low degrees (such as *"Kunjikuruvi"*, whose original degree is only 1) typically perform poorly because their sparse connectivity prevents sufficient training. MTG enriches their neighborhood structure, making them more closely integrated into the knowledge graph and yielding substantial performance improvement.

### C.4 Rational for Degree-one Graph Pruning

During graph pruning, we remove deep entities with a degree of one. This design choice raises the question of why only these entities generated by the LLM are pruned, rather than entities with degrees of two, three, or higher.

An entity with a degree of one is connected to the graph through a single edge. During backpropagation, its embedding update depends almost entirely on a single neighboring entity and relation. As a result, the model can easily fit such an entity by adjusting the parameters of this single connection. So, the embedding learning of degree one entities degenerates into a local fitting problem.

In contrast, entities with a degree greater than or equal to two participate in at least two triplets and can transmit information from multiple neighbors. Therefore, retaining these entities helps preserve the connectivity and semantic transmission paths of the graph.

### C.5 Impact of Graph Pruning on Graph Scale

In this work, we employ a graph pruning module to reduce parameter usage, which is closely related to the numbers of deep relations, entities, and triplets in the knowledge graph.

Accordingly, Figure 10 analyzes the effects of graph pruning on three datasets. The graph pruning reduces the numbers of deep relations, entities, and triplets across all datasets. Because entity embeddings dominate the model's storage cost, reducing the number of deep entities directly leads to a decrease in the total number of parameters.

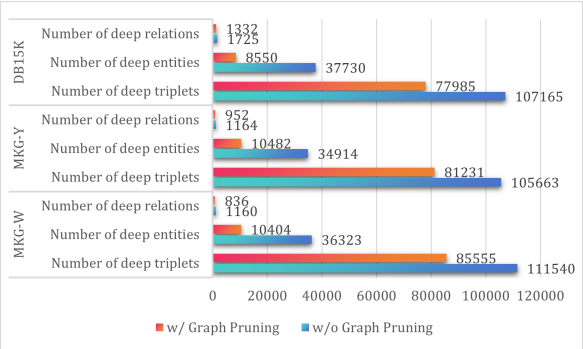



Figure 10: Diagram illustrating the impact of graph pruning. Comparison of the numbers of deep relations, entities, and triplets with and without graph pruning across three datasets.

Table 7: Performance comparison of four backbones for LLM on the MKG-Y dataset.

| Backbone | MRR | Hits@1 | Hits@3 | Hits@10 |
|---|---|---|---|---|
| Qwen (ours) | 0.506 | 0.446 | 0.542 | 0.613 |
| GPT-5 | 0.499 | 0.438 | 0.542 | 0.610 |
| Gemini 3 | 0.492 | 0.440 | 0.520 | 0.601 |
| ERNIE-4.5 | 0.470 | 0.428 | 0.524 | 0.586 |

In terms of time efficiency, graph pruning also reduces the number of deep triplets by nearly 20%, thereby reducing the training time for each epoch. Moreover, our ablation study shows that strong performance is maintained even after pruning.

### C.6 Impact of Different LLMs

In the main text, we employ a LLM for multimodal triplet generation, with Qwen serving as the default backbone. To further analyze the impact of different LLMs and assess whether alternative models could yield better performance, we conduct a comparative study using several representative LLMs. As reported in Table 7, we evaluate the MKG-Y dataset using three models, namely GPT-5, Gemini 3, and ERNIE-4.5-Turbo. Replacing the baseline LLM does not result in substantial performance variations, which demonstrates the robustness and general effectiveness of our proposed framework. We attribute this stability to the strict constraints imposed during the generation process. Specifically, we require the LLM to generate triplets solely based on the provided textual and visual information, with little introduction of additional

external knowledge or creative reasoning. As a result, although different LLMs are trained on diverse corpora, the generated outputs remain largely consistent, resulting in small performance differences between models.

### C.7 Theoretical Properties

**Definition 1.** *A relation between an entity pair* $(h, t)$ *is view-dependent iff* $\forall h, t : \exists r_1, r_2, v_1, v_2 : (h, r_1, t, v_1) \wedge (h, r_2, t, v_2) \wedge r_1 \neq r_2$.

**Definition 2.** *A triple* $(h, r, t)$ *is viewpoint-invariant iff* $\forall h, r, t, v_1 : (h, r, t, v_1) \rightarrow \exists v_2 : (h, r, t, v_2)$.

**Definition 3.** *The same triple may become invalid under a change of viewpoint.* $\forall h, r, t, v_1 : (h, r, t, v_1) \rightarrow \exists v_2 : \neg(h, r, t, v_2)$.

*Proof 1.* Given that $r_1$ evolves to $r_2$, and also given the two views $v_1$ and $v_2$, to model the pattern, we need to have $\phi(h, r_1, t, v_1) = \phi(h, r_2, t, v_2)$. Without loss of generality, we assume that we have only a one dimensional complex vector. In the following text, we use subscripts $a$ and $b$ to represent the real and imaginary parts of complex numbers. Then, we must fulfill the following equality:

$$\begin{aligned} &h'_a r'_{1v_1a} t'_a - h'_b r'_{1v_1b} t'_a + h'_b r'_{1v_1a} t'_b + h'_a r'_{1v_1b} t'_b \\ =&h'_a r'_{2v_2a} t'_a - h'_b r'_{2v_2b} t'_a + h'_b r'_{2v_2a} t'_b + h'_a r'_{2v_2b} t'_b \end{aligned} \tag{11}$$

To ensure that the above equation holds, we require $r'_{1v_1a} = r'_{2v_2a}, r'_{1v_1b} = r'_{2v_2b}$. It is important to note that $r'$ is obtained through complex transformations that combine static relation embeddings with view embeddings in Equation 6. Consequently, even when the above equalities are satisfied, the underlying representations are not necessarily identical. This condition is therefore easy to satisfy.

*Proof 2.* Assuming that $v_1$ evolves into $v_2$, the triplet $(h, r, t)$ remains invariant with respect to the evolution of views. To model this pattern, we require $\phi(h, r, t, v_1) = \phi(h, r, t, v_2)$. Therefore, the following equation must hold:

$$\begin{aligned} &h'_a r'_{v_1a} t'_a - h'_b r'_{v_1b} t'_a + h'_b r'_{v_1a} t'_b + h'_a r'_{v_1b} t'_b \\ =\ &h'_a r'_{v_2a} t'_a - h'_b r'_{v_2b} t'_a + h'_b r'_{v_2a} t'_b + h'_a r'_{v_2b} t'_b. \end{aligned} \tag{12}$$

After simplifying the above equation, two cases can satisfy the equality: 1) $r'_{v_1a} = r'_{v_2a}$ and $r'_{v_1b} = r'_{v_2b}$; 2) $h'_a t'_a = -h'_b t'_b$ and $h'_b t'_a = h'_a t'_b$.

For Case (1), it follows from Equation 6 that the condition can only be satisfied when $v_1$ and $v_2$ are identical, which contradicts the assumption that the views evolve and differ from each other. Therefore, this case is generally infeasible. For Case (2), we explicitly introduce the view embeddings $\mathbf{v}_h$ and $\mathbf{v}_t$ in Equation 5, which are added to the entity embeddings to form $h'$ and $t'$. This design endows the model with sufficient flexibility to naturally satisfy the above conditions.

*Proof 3.* Assuming that $v_1$ evolves into $v_2$, the possibility of triplet $(h, r, t)$ changes with the evolution of the view. To model this pattern, we require $\phi(h, r, t, v_1) \neq \phi(h, r, t, v_2)$. Accordingly, the following inequality must hold:

$$\begin{aligned} &h'_a r'_{v_1a} t'_a - h'_b r'_{v_1b} t'_a + h'_b r'_{v_1a} t'_b + h'_a r'_{v_1b} t'_b \\ \neq\ &h'_a r'_{v_2a} t'_a - h'_b r'_{v_2b} t'_a + h'_b r'_{v_2a} t'_b + h'_a r'_{v_2b} t'_b. \end{aligned} \tag{13}$$

After simplification, the above inequality holds under either of the following conditions: 1) $r'_{v_1a} \neq r'_{v_2a}$ and $r'_{v_1b} \neq r'_{v_2b}$; 2) $h'_a t'_a \neq -h'_b t'_b$ and $h'_b t'_a \neq h'_a t'_b$.

Compared with *Proof 2*, the inequalities in these two cases are easy to satisfy. In particular, for case (1), Equation 6 implies that as long as $v_1$ and $v_2$ are not identical, the inequality is satisfied.

### C.8 Noise Robustness

Due to the potential hallucinations produced by LLMs, our dual-view complex module, must be robust to noise introduced during triplet generation. Therefore, we perturb the generated triplets in Table 8 by randomly replacing the head or tail entities with a certain probability. As shown by the results on all three datasets, LLM-SE maintains strong performance even under 100% noise conditions, and still outperforms the uni-modal baseline ComplEx. This robustness can be attributed to the additional modeling capacity introduced by the view specific parameters $\mathbf{v}_h$, $\mathbf{v}_r$, and $\mathbf{v}_t$, as well as the use of N3 regularization. Furthermore, as the noise level is gradually reduced, the performance of LLM-SE consistently improves. These results provide empirical evidence that the proposed dual-view complex mechanism effectively mitigates the adverse impact of noisy triplets.

### C.9 Missing Robustness

Due to the reliance on multimodal features, it is necessary to evaluate the robustness of LLM-SE under modal missing conditions, as reported in Table 9. We assess model performance when textual and visual modalities are proportionally removed, where, for example, an 80% image setting indicates that 80% of image features are missing. Across

Table 8: Results of noise robustness experiment. The best results are presented in boldface.

| Noise | MKG-W | | | | MKG-Y | | | | DB15K | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MRR | Hits@1 | Hits@3 | Hits@10 | MRR | Hits@1 | Hits@3 | Hits@10 | MRR | Hits@1 | Hits@3 | Hits@10 |
| 100% | 0.299 | 0.248 | 0.321 | 0.390 | 0.323 | 0.300 | 0.332 | 0.362 | 0.359 | 0.267 | 0.398 | 0.505 |
| 80% | 0.337 | 0.287 | 0.355 | 0.429 | 0.354 | 0.327 | 0.368 | 0.400 | 0.362 | 0.271 | 0.409 | 0.520 |
| 60% | 0.368 | 0.314 | 0.395 | 0.464 | 0.387 | 0.352 | 0.407 | 0.448 | 0.371 | 0.282 | 0.421 | 0.541 |
| 40% | 0.399 | 0.344 | 0.424 | 0.497 | 0.433 | 0.386 | 0.458 | 0.513 | 0.395 | 0.309 | 0.441 | 0.560 |
| 20% | 0.442 | 0.383 | 0.469 | 0.549 | 0.462 | 0.407 | 0.497 | 0.559 | 0.428 | 0.321 | 0.469 | 0.591 |
| 0% | **0.477** | **0.405** | **0.519** | **0.607** | **0.506** | **0.446** | **0.542** | **0.613** | **0.441** | **0.351** | **0.488** | **0.604** |

Table 9: Results of missing robustness experiment. The best results are presented in boldface.

| Missing | MKG-W | | | | MKG-Y | | | | DB15K | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | MRR | Hits@1 | Hits@3 | Hits@10 | MRR | Hits@1 | Hits@3 | Hits@10 | MRR | Hits@1 | Hits@3 | Hits@10 |
| text 100% | 0.378 | 0.319 | 0.402 | 0.489 | 0.393 | 0.356 | 0.414 | 0.459 | 0.408 | 0.326 | 0.448 | 0.560 |
| text 80% | 0.398 | 0.335 | 0.424 | 0.515 | 0.413 | 0.372 | 0.435 | 0.488 | 0.413 | 0.329 | 0.453 | 0.566 |
| text 60% | 0.419 | 0.356 | 0.458 | 0.531 | 0.437 | 0.388 | 0.464 | 0.523 | 0.425 | 0.334 | 0.460 | 0.569 |
| text 40% | 0.439 | 0.371 | 0.475 | 0.559 | 0.459 | 0.405 | 0.491 | 0.554 | 0.429 | 0.340 | 0.465 | 0.572 |
| text 20% | 0.460 | 0.392 | 0.497 | 0.580 | 0.486 | 0.427 | 0.525 | 0.585 | 0.438 | 0.349 | 0.482 | 0.596 |
| image 100% | 0.462 | 0.391 | 0.498 | 0.590 | 0.493 | 0.435 | 0.529 | 0.592 | 0.436 | 0.349 | 0.482 | 0.595 |
| image 80% | 0.464 | 0.392 | 0.503 | 0.592 | 0.494 | 0.435 | 0.531 | 0.597 | 0.436 | 0.349 | 0.483 | 0.595 |
| image 60% | 0.467 | 0.395 | 0.507 | 0.596 | 0.496 | 0.435 | 0.534 | 0.600 | 0.436 | 0.349 | 0.486 | 0.598 |
| image 40% | 0.469 | 0.397 | 0.509 | 0.596 | 0.497 | 0.437 | 0.538 | 0.606 | 0.440 | 0.351 | 0.486 | 0.598 |
| image 20% | 0.473 | 0.400 | 0.512 | 0.604 | 0.501 | 0.440 | 0.541 | 0.606 | 0.441 | 0.350 | 0.486 | 0.602 |
| both 0% | **0.477** | **0.405** | **0.519** | **0.607** | **0.506** | **0.446** | **0.542** | **0.613** | **0.441** | **0.351** | **0.488** | **0.604** |

all three datasets, model performance consistently improves as the proportion of missing modalities decreases. In addition, similar to the observations in the ablation study, text missing leads to a more pronounced performance degradation than image missing. These results demonstrate that LLM-SE exhibits strong robustness to modal missing.

## D Prompt Setup

We have described the multimodal triplet generation process in detail throughout the paper. However, the main text can only offer a high-level explanation. So, we additionally provide the specific prompts and corresponding outputs used in the auxiliary steps of the LLM. Figure 11 illustrates representative keywords, prompts, and outputs for each stage.

## E Example of Deep Triplets

Because our framework leverages LLMs to transform multimodal data into structured representations, it is necessary to illustrate how the deep triplets generated by LLM-SE modify and enrich the original surface knowledge graph. To this end, Figure 12 presents a newly constructed KG obtained by partially fusing surface-level and deep-level knowledge.

### 3.1.1 Triplet Extraction

**Entity Name, Description and Images:**
Tip on a Dead Jockey "Tip on a Dead Jockey is a 1957 film, a drama directed by Richard Thorpe and starring Robert Taylor and Dorothy Malone. It is based on The New Yorker 1954 short story by Irwin Shaw"


**Prompt:**
You are a structured knowledge extraction system capable of processing both text and images.
Your task is to extract meaningful triples in the format: [head], [relation], [tail], [source]
source indicates whether the information is derived from "text", "image".
**Input:**
A textual description of an entity (e.g., film, person, location)
One or more related images
**Output Rules:**
Each triple must be on its own line, separated by commas: [head], [relation], [tail], [source]
Do not generate entities including numerical attributes or time information. Such as 18794 square kilometers, 7000 people, 1973. etc
Do not include any explanations, notes, or additional text outside the triples.
Do not use "Not specified", "Unknown", or similar placeholders as entities.
Distinguish whether the triple comes from an image or text accurately, Do not fabricate this information.
Do not generate repeat triples.
Try to generate around 5-10 quality and representative triples for one entity and Do not generate over 10 triples for one entity.
[Main Entity Prioritization]
The given main entity should preferably appear as the head or tail of the triples.
New entities can be introduced only if they can be reasonably inferred from the text, image, or general world knowledge.
[Joint Multimodal Reasoning]
Text is the primary source for extracting triples.
Images are used to verify, supplement, or strengthen information inferred from the text.
The extraction should be a joint reasoning process combining both modalities.
[Entity Formatting Constraints]
Both head and tail entities must be clearly identifiable, well-named entities such as people, places, institutions, rivers, or established conceptual entities
Do not use Boolean or vague phrases, or numerical descriptions as entities
If a relation cannot be paired with a clearly identifiable, named entity from the provided text or image, DO NOT generate that triplet. Omitting a triple is preferred over fabricating or approximating an entity.
[Textual and Visual Verifiability]
Do not fabricate entities or relations unsupported by the text or image.
Every extracted triple must be directly or indirectly supported by the text, the image, or logical inference grounded in both.
Example: If the image shows a building with a visible U.S. flag, this supports "has flag, American flag".
[Relation Coherence]
Relations must be semantically clear and expressed as short words or phrases, not full sentences. Refer to the relations in the original dataset given below.
Relations must be semantically clear and expressed as short words or phrases, not full sentences. Refer to the relations in the original dataset given below.
When possible, reuse relations from the original dataset if they capture the intended meaning (for example, use "shares border with" instead of "borders").
However, if an appropriate relation does not exist, you may create a new one that fits the meaning accurately.
[Category-Aware Image Reasoning]
When extracting triples based on images, follow the corresponding extraction strategy.(Only for image-based triples)
Category: LOCATION
Focus on landmarks, architecture, and natural features visible in the image.
Typical relations: located in, contains landmark, is part of, has flag, capital of, architectural style, shares border with.
New entities (such as United States Capitol) may be introduced if visually or textually supported.
Category: MOVIE POSTER
Focus on textual cues from the poster (title, director, actors, producer, distributor, etc.).
Typical relations: director, screenwriter, producer, distributor, cast member, composer
New entities such as actor names or company names are allowed.
Category: PERSON (PORTRAIT)
Extract the person's representative features visible in the image.
Typical relations: wears, facial hair.
Example: J. D. Chakravarthy, wears, black suit, image.
The examples and typical relations provided are only partial. You are encouraged to extract diverse relations but they need to adhere to the requirement of "[Clear and Coherent Relations]". For other relations not falling into the above categories, please use common sense and refer to the above categories to arrange the extraction strategy.
Relations in the original dataset: (Truncated)
# Now process the following entity and its description:
# Entity: {entity}
# Description: {description}
**Output:**
Tip on a Dead Jockey, director, Richard Thorpe, text
Tip on a Dead Jockey, actor, Robert Taylor, text
Tip on a Dead Jockey, country, U.S.A., image
Tip on a Dead Jockey, cinematography format, CinemaScope, image
...

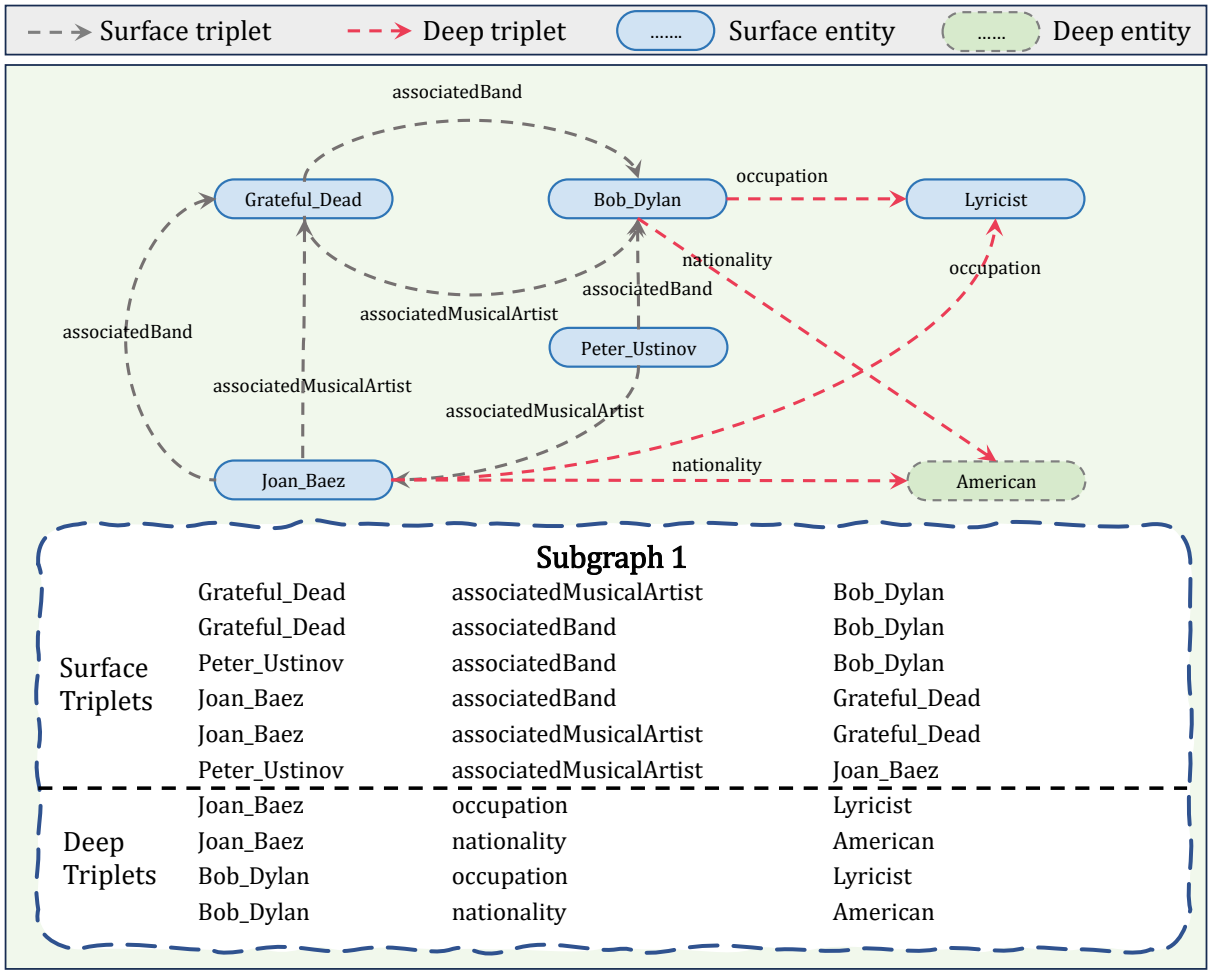
### 3.1.2 Relation Alignment

**Prompt:**
In the previous step, there were extracted triplets from the Wikidata knowledge graph. Each triplet contains two entities (subject and object) and one relation that connects these subject and object. However, some of the relations extracted in the previous step may have not an exact name from Wikidata.
We linked each relation name with top similar exact names from the Wikidata by semantic similarity. Your task is to choose appropriate names for relations that correspond to the text's context and triplet they were taken from.
Text: {description}
Triplets and corresponding relation mappings: {triplets_text}
For each relation from the extracted triplets, you must choose the most appropriate name only from the corresponding list of 5 exact ones that better match each triplet and the context of the previously demonstrated text.
You must ONLY output the final chosen relation for each triple, one per line, in this format:
Triple 1: [chosen_relation]
Triple 2: [chosen_relation]
...
**Output:**
director:['director', 'assistant director', 'LUMIERE director ID', 'executive producer', 'director of photography'
Triple1: [director]
actor:['film crew member', 'cast member', 'character role', 'Television Academy Foundation interviewee ID', 'Kinofilms.ua actor ID']
Triple2: [cast member]
country:['country', 'INAO product ID', 'country of citizenship', 'OpenCorporates register jurisdiction', 'TASS Encyclopedia country ID']
Triple3: [country]
cinematography format:['camera setup', 'director of photography', 'video designer', 'original film format', 'film editor']
Triple4: [original film format]

### 3.1.3 Graph Deduplication

Entity Description Acquisition
**New Entities**
'Richard Thorpe', 'Robert Taylor', 'U.S.A', 'CinemaScope', 'United States of America', ...
**New Relations**
'director', 'cast member', 'country', 'original film format', ...
**Prompt:**
Please provide a concise English description (under 30 words) for each entity '{entity_name}' and relation '{relation_name}' .
**Output:**
U.S.A(new) desc:"United States of America: a federal republic in North America."
United States of America(new) desc:"A federal republic in North America comprising 50 states."
country(new) desc:"The nation associated with an individual, group, or subject in a general context."
country of origin(original) desc:"The nation specifically linked to the birthplace, creation, or initial development of something."
Cross-Graph Deduplication
**### Step1: Deep-Deep pairs**
Similarity(U.S.A, United States of America) = 0.96
U.S.A(new) ⇒ United States of America(new)
**### Step2: Deep-Surface pairs**
Similarity(country, country of origin) = 0.91
country(new) ⇒ country of origin(original)
**Final Output:**
Tip on a Dead Jockey, director, Richard Thorpe
Tip on a Dead Jockey, cast member, Robert Taylor
Tip on a Dead Jockey, country of origin, United States of America
Tip on a Dead Jockey, original film format, CinemaScope

Figure 11: Specific inputs of LLM in the process of multimodal triplet generation.



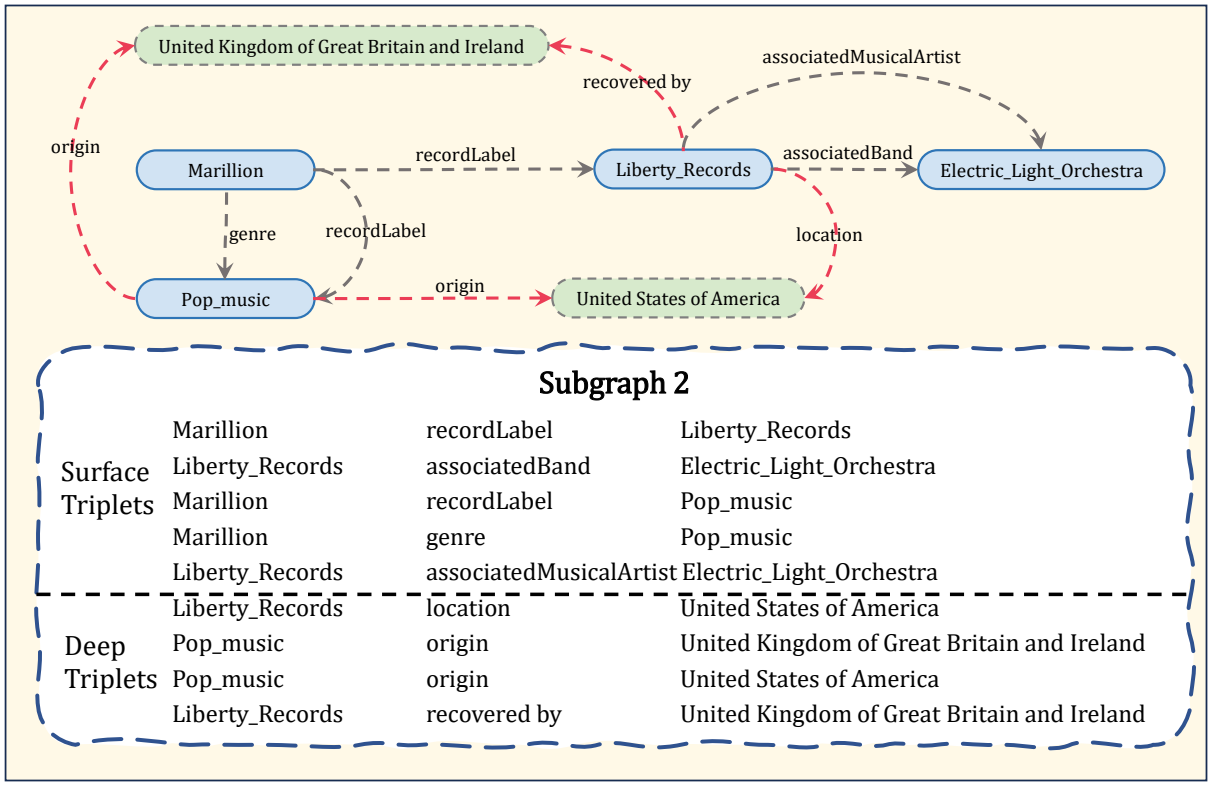




Figure 12: Examples of surface triplets and deep triplets.