# MAKI: Multi-layer Aligned Knowledge Injection for Structure-aware Knowledge Graph Completion with Large Language Models

**Zhiwen Xie**[1,2], **Xin Wang**[3], **Guangyou Zhou**[2*], **Derek F. Wong**[1*]
[1] NLP[2]CT Lab, Department of Computer and Information Science, University of Macau
[2]School of Computer Science, Central China Normal University
[3]Huawei Technologies Co., Ltd.
zwxie@ccnu.edu.cn, wangxin616@huawei.com,
gyzhou@mail.ccnu.edu.cn, derekfw@um.edu.mo

## Abstract

Recent advances in large language models (LLMs) have shown strong potential for knowledge graph completion (KGC). However, existing LLM-based approaches often struggle to effectively capture the structural information in knowledge graphs (KGs), leading to suboptimal reasoning performance. To address this challenge, we propose a Multi-layer Aligned Knowledge Injection (MAKI) model, a novel method that tightly integrates structured KG information into LLMs through multi-layer alignment. Specifically, we first leverage LLMs to encode the textual information of entities and relations, obtaining their semantic representations across multiple hidden layers. We then introduce a multi-layer aligned structure learning module, which uses graph neural networks (GNNs) to learn relational structures while aligning with the corresponding LLM layers to bridge the gap between structural and semantic spaces. Finally, a gated fusion mechanism is used to inject the structured knowledge into the LLM for reasoning over candidate triples. Experimental results on various benchmark datasets demonstrate that the proposed MAKI outperforms existing state-of-the-art methods.

## 1 Introduction

Knowledge Graphs (KGs) are structured representations of facts, where entities are connected through relationships, typically expressed in the form of triples (head entity, relation, tail entity). These graphs have gained prominence in a wide range of downstream applications, including intelligent search (Zhu et al., 2024), recommendation system (Meng et al., 2025; Zhang et al., 2025), and question answering (Long et al., 2025). Despite their broad utility, KGs are inherently incomplete which severely hinders their effectiveness, especially in downstream tasks that rely on comprehensive and accurate knowledge. To address this issue, Knowledge Graph Completion (KGC) has emerged as a crucial research problem. The goal of KGC is to infer missing triples in a KG by predicting unknown relationships between entities based on observed triples.

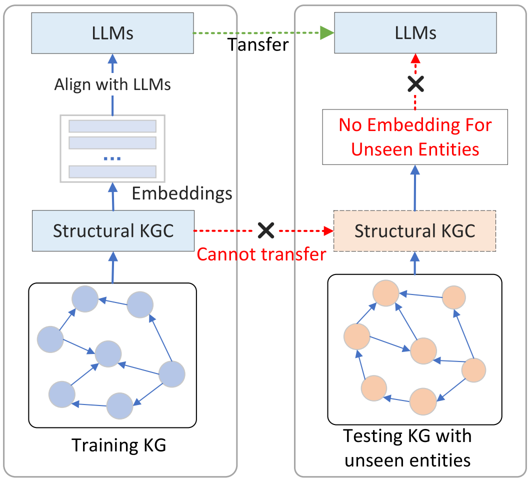


Figure 1: The limitation of existing KG structure enhanced LLM methods.

Existing approaches for KGC can be broadly categorized into structure-based methods and text-based methods. Structure-based approaches primarily focus on the inherent topology of KGs, learning representations of entities and relations through observed structural patterns, such as TransE (Bordes et al., 2013) and RotatE (Sun et al., 2019). While effective for well-connected entities, these models rely heavily on the availability of rich structural information and often underperform on long-tail entities that have limited relational context in KGs. To address this, text-based methods leverage pre-trained language models (PLMs) to encode the textual descriptions of entities and relations, offering a natural way to infer representations even when structural information is sparse. Methods such as KG-BERT (Yao et al., 2019), SimKGC

* Corresponding author.



*Findings of the Association for Computational Linguistics: ACL 2026*, pages 28527–28539

(Wang et al., 2022), and PKGC (Lv et al., 2022) fall into this category. With the recent success of Large Language Models (LLMs) such as LLAMA (Touvron et al., 2023a) and GPT-4 (OpenAI, 2023), researchers have begun exploring their potential for KGC. Yao et al. (2023) were among the first to investigate this direction, employing instruction tuning to guide LLMs to perform triple-level reasoning. While promising, these methods typically rely only on textual information of individual triples, ignoring the underlying graph structure that encodes additional relational patterns.

To incorporate structural context, some recent methods, such as KoPA (Zhang et al., 2024) and SAT (Liu et al., 2025), use structure-based methods (e.g., RotatE) to learn knowledge embeddings and align them with LLMs. However, such integration methods face significant limitations. First, the vector spaces of traditional structure-based KGC models (e.g., RotatE) and LLMs are fundamentally different, which poses a critical challenge. Structure-based KGC methods encode relational patterns through geometric transformations, while LLM embeddings are grounded in contextualized natural language semantics. Simply projecting one space into another does not guarantee meaningful alignment, which may lead to ineffective or even misleading fusion of information. Second, most structure-based methods, including RotatE and TransE, are inherently transductive, which are unable to generate representations for unseen entities that were not present during training. This restricts their utility in open-domain or evolving knowledge settings, where new entities frequently emerge. Consequently, models like KoPA (Zhang et al., 2024) and SAT (Liu et al., 2025), which uses embeddings from transductive structural KGC methods as prompt of LLMs, may limit the generalization capabilities of LLMs, as shown in Figure 1. These limitations underscore the need for a more principled and unified framework that can seamlessly integrate structural and textual information into a coherent semantic space, enhancing the reasoning capacity of LLMs while preserving their flexibility and generalization potential.

To address the aforementioned limitations, we propose a Multi-layer Aligned Knowledge Injection (MAKI) framework, which aims to enhance the reasoning capability of Large LLMs on KGC tasks by injecting structured knowledge from KGs into the LLM in a layer-wise, semantically aligned manner. Specifically, for each entity and relation in a KG, we first leverage the LLM to encode their textual information, and obtain hidden representations from multiple layers of the LLM. Then, these hidden representations are fed into a multi-layer graph neural network (GNN) to learn structure-aware embeddings. Each layer of the GNN is explicitly aligned with a corresponding layer in the LLM, forming a layer-wise bridge between the structural and semantic spaces. Finally, the structure-encoded representations produced by each GNN layer are injected into the corresponding LLM layer via a gated fusion mechanism, enabling the LLM to reason not only from textual semantics but also from the structure information in the KG. By combining textual and structural signals in a layer-aware aligned manner, MAKI provides a principled solution to overcoming the semantic misalignment and generalization limitations faced by prior methods. The main contributions of this paper are summarized as follows:

- We propose a novel Multi-layer Aligned Knowledge Injection (MAKI) framework, which effectively integrates structural knowledge from KGs into LLMs.
- We design a multi-layer aligned GNN module that captures the relational structure of KGs and aligns it with multiple layers of the LLM, bridging the gap between structure and semantic representations.
- We conduct extensive experiments on three transductive datasets and eight versions of inductive datasets, demonstrating that MAKI consistently outperforms existing state-of-the-art methods.

## 2 Related Work

### 2.1 Structure-based KGC

Structure-based approaches focus on modeling the topological structure of KGs by learning embeddings for entities and relations. These methods project entities and relations into continuous vector spaces and learn scoring functions to predict missing links, including but not limited to TransE (Bordes et al., 2013), RotatE (Sun et al., 2019), DistMult (Yang et al., 2015), PairRE (Chao et al., 2021), DaBR (Wang et al., 2025), RSCF (Kim et al., 2025). Despite their success, structure-based KGC methods heavily rely on sufficient connectivity. For sparsely connected entities, especially those

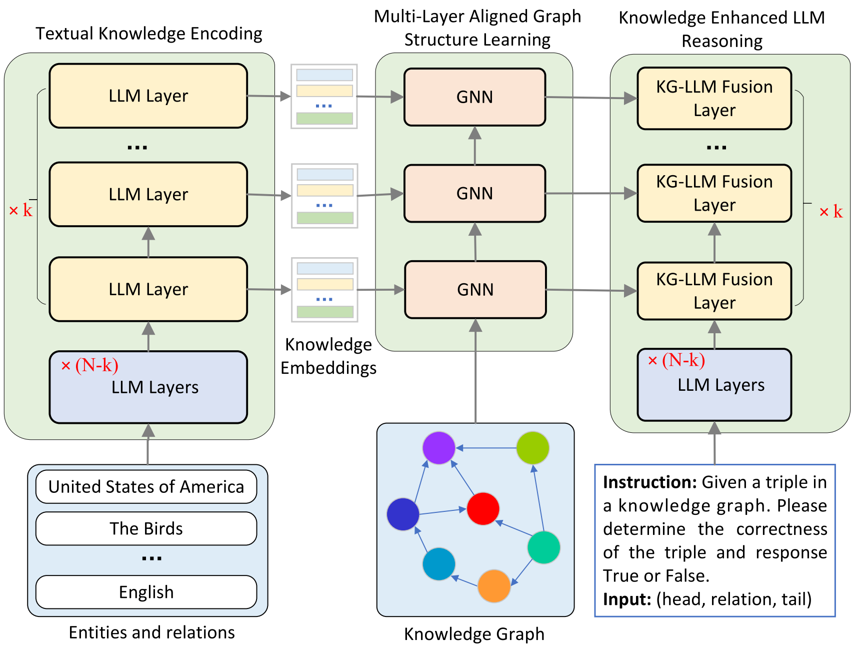


Figure 2: The structure of the proposed MAKI model.

in the long-tail distribution, the performance often degrades significantly due to insufficient context for representation learning. In addition, structure-based methods typically fail to predict links for entities not observed during training, as they lack auxiliary information such as textual descriptions.

### 2.2 Text-based KGC

Text-based KGC methods aim to enhance entity and relation representations by leveraging unstructured textual information, such as entity names, entity descriptions, and relation names. These methods are particularly effective for addressing sparsity issues in structure-based KGC, especially for long-tail or emerging entities with limited connectivity in the graph. A dominant trend in text-based KGC is to use pretrained language models (PLMs) to encode textual descriptions into dense vector representations. These PLMs capture rich semantic and syntactic information from large corpora, enabling effective generalization and reasoning over text. KG-BERT (Yao et al., 2019) reformulates triple classification as a natural language inference task by feeding textual triple sequences into BERT (Devlin et al., 2019). PKGC (Lv et al., 2022) and CSPromp-KG (Chen et al., 2023) uses prompts to integrate both structured knowledge and textual context. Despite their success, these PLM-based methods typically rely on relatively small-scale models (e.g., BERT (Devlin et al., 2019)), which limits their reasoning capabilities and contextual understanding. With the advent of LLMs, such as LLAMA (Touvron et al., 2023a) and GPT-4 (OpenAI, 2023), a new line of research has begun exploring LLMs for KGC. KG-LLM (Yao et al., 2023) explores the use of instruction-tuned LLMs for KGC. MKGL (Guo et al., 2024) integrate LLMs with KGs using a specialized KG language. KoPA (Zhang et al., 2024) and SAT (Liu et al., 2025) pre-train structure-based embeddings for entities and relations, and then uses a prefix adapter to incorporate the structure knowledge into LLMs. In addition, DiffCLR (Liu et al., 2024) and GLM (Plenz and Frank, 2024) also integrate KGs with LLMs for better reasoning. However, due to the semantic gap between KG embeddings and LLM representations, these approaches face challenges in achieving effective integration and generalization to unseen entities.

## 3 Our Approach

### 3.1 Overview

The overall architecture of MAKI is illustrated in Figure 2. Specifically, the proposed MAKI model consists of three components: Textual Knowledge Encoding, Multi-Layer Aligned Graph Structure Learning and Knowledge Enhanced LLM Reasoning. These modules are designed to work in a unified framework to effectively integrate textual and structural knowledge for KGC. In the textual knowledge encoding module, we leverage a pretrained LLM to encode the textual names of entities and relations in the knowledge graph. This process generates rich semantic representations that capture contextual and linguistic information, serving as the initial features for subsequent processing. The multi-layer aligned structure learning component is responsible for modeling the local graph structure of the KG using Graph Neural Networks (GNNs). Each GNN layer is aligned with a specific layer in the LLM to facilitate layer-wise semantic correspondence and gradual knowledge integration. In the final stage, we inject the structural knowledge learned by the GNNs into the corresponding layers of the LLM. This fusion is performed in a layer-wise aligned manner, enabling the LLM to reason jointly over both textual semantics and graph structure.

### 3.2 Textual Knowledge Encoding

To bridge the semantic gap between the structured KGs and LLMs, we adopt a text-driven representation approach, which aligns the representations of entities and relations in the semantic space of LLMs. Specifically, we utilize the surface name of each entity and relation in the KG and encode them using a pretrained LLM.

Formally, for each entity $e$ or relation $r$, we input the corresponding text into the LLM and extract its contextual hidden states. We use the hidden states of the last token to represent the semantic embedding of the input, which is commonly adopted in LLM-based representation learning (Geva et al., 2023). This yields the initial embedding for each entity and relation at each layer:

$$\begin{aligned} \mathbf{h}_e^l &= \mathrm{LLM}^l(e_{text}) \\ \mathbf{h}_r^l &= \mathrm{LLM}^l(r_{text}) \end{aligned} \quad (1)$$

where $\mathbf{h}_e^l$ and $\mathbf{h}_r^l$ denote the hidden representation of the last token of $e_{text}$ and $r_{text}$ at the $l$-the layer of LLM, respectively.

Instead of relying solely on the final layer's output, we preserve multiple intermediate representations for each entity and relation from top-$k$ layers of the LLM, namely $\{\mathbf{h}_e^{N-k+1}, \mathbf{h}_e^{N-k+2}, \cdots, \mathbf{h}_e^N\}$ for each entity and $\{\mathbf{h}_r^{N-k+1}, \mathbf{h}_r^{N-k+2}, \cdots, \mathbf{h}_r^N\}$ for each relation. This design enables the model to perform fine-grained layer-wise integration of graph-based knowledge into the LLM.

### 3.3 Multi-Layer Aligned Graph Structure Learning

To effectively inject structural knowledge into LLMs, it is crucial to not only capture the topological patterns of the KG but also to ensure that the learned structural representations are semantically aligned with the LLM's latent space. To this end, we design a multi-layer aligned graph structure learning module that combines the benefits of GNNs and LLMs through a carefully designed layer-wise alignment strategy. At each layer of this module, the input consists of two parts: (1) the hidden states from the previous GNN layer, which carry multi-hop structural information of the KG, and (2) the LLM-encoded vector representation from the corresponding LLM layer, which ensures semantic alignment with the LLM's embedding space. These two kinds of features are fused at each layer to enable the GNN to learn structure-aware representations that remain consistent with the LLM's semantic space.

Specifically, we align the GNN layers with the top-$k$ layers of the LLMs, which are known to encode higher-level semantics. Each layer of the GNN is aligned with one of these top-$k$ layers of the LLM allowing for deep semantic fusion. At the $l$-th layer of the GNN, we take two kinds of features as the input for each entity $e$, including the output hidden state $\mathbf{s}_e^{l-1}$ of previous GNN layer and the corresponding hidden state $\mathbf{h}_e^{N-k+l}$ from LLM. These two kinds of features are integrated to obtain an intermediate feature that carries both semantic and structural information, which is formally denoted as:

$$\mathbf{m}_e^l = \mathbf{s}_e^{l-1}\mathbf{W}_1 + \mathbf{h}_e^{N-k+l}\mathbf{W}_2 \quad (2)$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ are trainable parameters.

Then, a message passing operation is applied to capture the neighboring information of each entity, which is computed as:

$$\bar{\mathbf{s}}_e^l = \frac{1}{|\mathcal{N}_e|} \sum_{(e,r,p)\in\mathcal{N}_e} [\mathbf{m}_p^l \odot \mathbf{h}_r^{N-k+l}\mathbf{W}_3] \quad (3)$$

$$\mathbf{s}_e^l = \bar{\mathbf{s}}_e^l \mathbf{W}_4 \quad (4)$$

where $\mathcal{N}_e$ is a set of neighboring triples of entity $e$, $\mathbf{W}_3$ and $\mathbf{W}_4$ are trainable linear transformation matrices. This message-passing process enables the entity representations to capture multi-hop relational patterns, while maintaining layer-wise alignment with the LLM.

### 3.4 Knowledge Enhanced LLM Reasoning

In the final stage, we integrate the structural knowledge learned by the multi-layer aligned graph structure learning module into the LLM to enhance its reasoning capabilities for KGC tasks. By fusing the graph-based representations with the LLM's latent semantic space, we enable the model to leverage both structure knowledge and contextual language modeling, improving its ability to make predictions about missing triples in KGs.

Following previous work (Zhang et al., 2024), we reformulate the KGC task as a triple classification problem, where the model is asked to judge whether a given candidate triple is plausible. Specifically, given a candidate triple $(e_i, r, e_j)$, we construct a prompt $P$ that represents the triple in a textual form and instructs the LLM to determine whether it is semantically valid. The prompt is composed of two parts:

$$P = I \oplus X(e_i, r, e_j) \tag{5}$$

where $I$ is the instruction that guides the LLM toward the desired reasoning task (e.g., "*Please determine the correctness of the triple and response True or False*"), and $X(e_i, r, e_j) = D(e_i) \oplus D(r) \oplus D(e_j)$ is the verbalized form of the triple, where $D(\cdot)$ denotes the textual information (e.g., surface name or definition) of each entity and relation.

The prompt $P$ is fed into the LLM, and the model directly generates one of two target tokens: "True" if the triple is semantically valid, or "False" otherwise:

$$o_{(e_i, r, e_j)} = \mathrm{LLM}(P) \in \{\mathrm{True}, \mathrm{False}\} \tag{6}$$

To enhance the model's prediction with structural context, we introduce the KG-LLM Fusion Layer, which injects structured knowledge learned from the KG into the internal representations of the LLM. Specifically, for each of the top-$k$ layers of the LLM, we align the GNN's output with the corresponding hidden representations of LLM. Formally, we inject the $l$-th layer of structural feature vector $s_e^l$ into the $(N-k+l)$-th layer of LLM as follows:

$$\bar{\mathbf{h}}_e^{N-k+l} = \theta \times \mathbf{h}_e^{N-k+l} + (1-\theta) \times \mathbf{s}_e^l \tag{7}$$

where $h_e^{N-k+l}$ is the hidden state of the last token for entity $e$ at the $(N-k+l)$-th layer of LLM, and $\theta$ is a gate vector to control the contribution of $h_e^{N-k+l}$ and $s_e^l$, which is computed as:

$$\theta = \sigma([\mathbf{h}_e^{N-k+l} \oplus \mathbf{s}_e^l]\mathbf{W}_5) \tag{8}$$

where $\sigma$ is the sigmoid function. This gated mechanism allows the model to adaptively balance semantic and structural information, preserving the contextual strength of the LLM while enriching it with structure knowledge from the KG.

In each KG-LLM Fusion layer, we inject the structure features of $e_i$, $r$ and $e_j$ into their corresponding hidden states in LLM layers, resulting in a knowledge-enhanced LLM. Then, the model is fine-tuned by supervising the LLM to generate the correct label.

## 4 Experiments

### 4.1 Datasets and Evaluation Metrics

To evaluate the effectiveness of the proposed method, we conduct experiments on two typical KGC tasks, e.g., triple classification and link prediction. For triple classification task, we evaluate our model on three widely used triple classification benchmarks: UMLS (Kok and Domingos, 2007; Yao et al., 2019), CoDeX-S (Safavi and Koutra, 2020), and FB15K-237N (Lv et al., 2022). The performance of the triple classification task is measured using accuracy, precision, recall, and F1 scores. For the link prediction task, we conduct experiments on the inductive subsets of FB15K-237 and WN18RR from GraIL (Teru et al., 2020), as the testing KGs in these datasets contain unseen entities in training KGs, allowing a better evaluation of the model's reasoning ability on unknown entities. Following previous work (Teru et al., 2020; Mai et al., 2021; Chen et al., 2022), we rank each query triple against other 50 negative triples, and ranked all candidates based on predicted probabilities of labels. We report the Mean Reciprocal Rank (MRR) metric for the link prediction task. More detailed introduction and statistics of the datasets are summarized in Appendix A.

### 4.2 Experimental Setup

All experiments are conducted on a single NVIDIA A6000 GPU with 48GB of memory. Our implementation is based on PyTorch. For the graph neural network (GNN) component in multi-layer aligned graph structure learning, we set the hidden dimension to 512 and stack 6 GNN layers to enable structural message passing across multi-hop neighborhoods and deep information fusion with LLM. We use LLama3.1-8B [1] as the LLM backbone and apply LoRA (Hu et al., 2022) with a rank of 16 to

[1]https://huggingface.co/meta-llama/Llama-3.1-8B

| Models | UMLS | | | | CoDeX-S | | | | FB15K-237N | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | P | R | F1 | Acc | P | R | F1 | Acc | P | R | F1 | Acc |
| TransE | 86.53 | 81.69 | 84.04 | 84.49 | 71.91 | 72.42 | 72.17 | 72.07 | 70.80 | 67.11 | 68.91 | 69.71 |
| DistMult | 87.06 | 86.53 | 86.79 | 86.38 | 69.67 | 59.46 | 64.16 | 66.79 | 58.98 | 56.84 | 57.9 | 58.66 |
| ComplEx | 89.92 | 91.83 | 90.87 | 90.77 | 67.84 | 67.06 | 67.45 | 67.64 | 66.46 | 63.38 | 64.88 | 65.70 |
| RotatE | 90.17 | 94.41 | 92.23 | 92.05 | 75.66 | 75.71 | 75.69 | 75.68 | 69.24 | 66.41 | 67.8 | 68.46 |
| KG-BERT | 70.96 | 92.43 | 80.28 | 77.3 | 70.96 | 92.43 | 80.28 | 77.3 | 53.47 | 97.62 | 67.84 | 56.02 |
| PKGC | 79.87 | 93.29 | 86.05 | 83.96 | 78.33 | 91.57 | 84.43 | 81.68 | 67.37 | 96.18 | 79.23 | 79.16 |
| Zero-shot(GPT-3.5) | 88.04 | 40.71 | 55.67 | 67.58 | 69.13 | 16.94 | 27.21 | 54.68 | 86.62 | 24.01 | 37.59 | 60.15 |
| Zero-shot(Alpaca) | 51.55 | 87.69 | 64.91 | 52.64 | 50.31 | 99.83 | 66.91 | 50.62 | 53.32 | 97.37 | 68.91 | 56.06 |
| KG-LLaMA | 87.84 | 83.05 | 85.38 | 85.77 | 78.67 | 80.74 | 79.69 | 79.43 | 67.37 | 96.23 | 79.25 | 74.81 |
| KG-Alpaca | 94.91 | 76.1 | 84.46 | 86.01 | 79.38 | 81.73 | 80.54 | 80.25 | 62.71 | 98.28 | 76.56 | 69.91 |
| Vanilla IT | 95.18 | 77.76 | 85.59 | 86.91 | 77.01 | 88.89 | 82.52 | 81.18 | 65.87 | 97.53 | 78.63 | 73.50 |
| Structure-aware IT | 93.27 | 86.08 | 89.54 | 89.93 | 77.14 | 88.4 | 82.58 | 81.27 | 69.56 | 93.95 | 79.94 | 76.42 |
| KoPA | 90.85 | 94.70 | 92.70 | 92.58 | 77.91 | 91.41 | 84.11 | 82.74 | 70.81 | 94.09 | 80.81 | 77.65 |
| SAT | - | - | - | - | 83.40 | 89.30 | 86.50 | 85.6 | 82.30 | 85.20 | 83.10 | 82.70 |
| **MAKI** | 97.43 | 98.33 | **97.88** | **97.54** | 85.70 | 88.84 | **87.24** | **87.01** | 81.22 | 89.35 | **85.09** | **84.35** |

Table 1: Triple classification results on UMLS, CoDeX-S and FB15K-237N datasets. The best results are in bold.

| Models | FB15K-237 | | | | WN18RR | | | |
|---|---|---|---|---|---|---|---|---|
| | v1 | v2 | v3 | v4 | v1 | v2 | v3 | v4 |
| GraIL | 46.25 | 39.14 | 53.64 | 49.53 | 75.12 | 58.84 | 45.28 | 73.53 |
| ComPILE | 51.35 | 64.84 | 64.24 | 63.36 | 74.35 | 58.76 | 46.87 | 73.21 |
| MorsE | 58.53 | 74.72 | 73.95 | 73.32 | 69.59 | 69.77 | 52.06 | 66.37 |
| KG-BERT | 45.12 | 56.18 | 54.15 | 53.49 | 77.29 | 83.44 | 86.18 | 86.54 |
| PKGC | 47.98 | 58.76 | 55.26 | 53.77 | 79.94 | 86.01 | 87.06 | 89.25 |
| KG-LLaMA | 79.57 | 80.66 | 79.73 | 79.98 | 84.52 | 90.97 | 91.42 | 91.31 |
| KG-Alpaca | 81.17 | 80.71 | 80.42 | 81.12 | 85.15 | 90.06 | 92.87 | 92.57 |
| Vanilla IT | 81.86 | 82.02 | 82.35 | 81.89 | 86.85 | 91.65 | 93.14 | 92.96 |
| Structure-aware IT | 82.67 | 82.14 | 81.75 | 82.59 | 87.39 | 93.15 | 93.38 | 93.24 |
| **MAKI** | **86.45** | **86.35** | **83.33** | **85.91** | **93.69** | **98.63** | **96.83** | **97.43** |

Table 2: The MRR results for link prediction task on inductive versions of the FB15K-237 and WN18RR datasets. The best results are in bold.

fine-tune a small number of parameters while keeping the majority of the LLM frozen. We train the model using the Adam (Kingma and Ba, 2015) optimizer with a learning rate of 5e-5. And we train the model with a batch size of 32 for 5 epochs.

### 4.3 Baselines

We compare our proposed MAKI method with several strong baselines from three categories: (1) structure-based models, including TransE (Bordes et al., 2013), DistMult (Yang et al., 2015), ComplEx (Trouillon et al., 2016) and RotatE (Sun et al., 2019) for triple classification task; GraIL (Teru et al., 2020), ComPILE (Mai et al., 2021), MorsE (Chen et al., 2022) for inductive link prediction task; (2) PLM-based models, including KG-BERT (Yao et al., 2019) and PKGC (Lv et al., 2022), which use small size pre-trained language models (e.g., BERT (Devlin et al., 2019)) to encode textual descriptions of triples; and (3) LLM-based methods which use LLMs to predict the answer, including training-free methods such as Zero-shot prediction using GPT-3.5-turbo (175B parameters) and Alpaca, and fine-tuning methods that fine-tune LLMs on KGC task, including KG-LLaMA (Yao et al., 2023), KG-Alpaca (Yao et al., 2023), Vanila IT (Zhang et al., 2024), Structure-aware IT (Zhang et al., 2024), KoPA (Zhang et al., 2024), and SAT (Liu et al., 2025).

### 4.4 Main Results

**Triple Classification Results**. The triple classification results on UMLS, CoDeX-S and FB15K-237N are summarized in Table 1. We can see that our MAKI method consistently achieves state-of-the-art (SOTA) performance across all three benchmarks on both accuracy and F1 scores. Specifically, on the UMLS dataset, our MAKI model achieves 97.88% F1 score and 97.54% accuracy, outperforming previous SOTA methods RotatE and KoPA by

| Models | UMLS | | | | CoDeX-S | | | | FB15K-237N | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | P | R | F1 | Acc | P | R | F1 | Acc | P | R | F1 | Acc |
| MAKI | 97.43 | 98.33 | **97.88** | **97.54** | 85.70 | 88.84 | **87.24** | **87.01** | 81.22 | 89.35 | **85.09** | **84.35** |
| w/o TextInit | 97.47 | 87.44 | 92.19 | 92.54 | 80.39 | 88.13 | 84.08 | 83.32 | 74.65 | 93.45 | 83.01 | 80.86 |
| w/o GNN | 98.12 | 79.22 | 87.61 | 88.73 | 84.11 | 77.57 | 80.71 | 81.45 | 72.28 | 88.28 | 79.48 | 77.21 |
| w/o Gate | 98.38 | 92.13 | 95.16 | 95.28 | 83.04 | 89.75 | 86.25 | 85.69 | 78.31 | 89.69 | 83.61 | 82.42 |

Table 3: Ablation study on UMLS, CoDeX-S and FB15K-237N datasets.

| Models | FB15K-237 | | | | WN18RR | | | |
|---|---|---|---|---|---|---|---|---|
| | v1 | v2 | v3 | v4 | v1 | v2 | v3 | v4 |
| **MAKI** | **86.45** | **86.35** | **83.33** | **85.91** | **93.69** | **98.63** | **96.83** | **97.43** |
| w/o TextInit | 85.39 | 84.97 | 84.76 | 84.87 | 92.85 | 96.47 | 95.46 | 97.01 |
| w/o GNN | 83.44 | 82.76 | 82.54 | 82.72 | 89.73 | 93.92 | 94.18 | 94.38 |
| w/o Gate | 86.01 | 85.45 | 85.16 | 85.33 | 93.08 | 97.65 | 95.94 | 97.22 |

Table 4: Ablation study on inductive versions of the FB15K-237 and WN18RR datasets.

large margins. On the CoDeX-S dataset, our MAKI method reaches 87.24% F1 score and 87.01% accuracy, again surpassing previous structure-based models, PLM-based models, and LLM-based models. On the more challenging dataset FB15K-237N, MAKI continues to demonstrate superior performance, obtaining a F1 score of 85.09% and an accuracy of 84.35%, outperforming previous SOTA methods KoPA (Zhang et al., 2024) and SAT (Liu et al., 2025). These results validate the effectiveness of our multi-layer aligned knowledge injection strategy in our MAKI model.

**Link Prediction Results**. Table 2 reports the MRR results of link prediction task on the inductive versions of FB15K-237 and WN18RR, where the entities in testing set are unseen in the training KG. On these inductive datasets, LLM-based approaches demonstrate substantially stronger inductive generalization. Models such as KG-LLaMA, KG-Alpaca, Vanilla IT and Structure-aware IT consistently outperform both structure-based and PLM-based baselines across all versions of inductive datasets. This highlights the advantage of LLMs in capturing transferable reasoning features. Compared to all the baselines, our MAKI model further improves performance by explicitly incorporating structural knowledge, confirming the importance of aligning LLM reasoning with graph topology. These results validate the inductive reasoning ability of MAKI and demonstrate its potential for generalizing to evolving or incomplete KGs where new entities frequently emerge. More results on Hits@10 metric are listed in Appendix B.

In contrast to existing LLM-based approaches, MAKI incorporates structural information from the KG through a multi-layer GNN that is aligned with the internal representations of the LLM. This enables MAKI to preserve both semantic richness and relational structure. Unlike KoPA and SAT, which requires to pre-train a KG embedding model (e.g., RotatE) on a fixed set of entities, our is able to dynamically learn structure-aware representations for unseen entities via textual knowledge encoding and multi-layer aligned GNNs, enabling more effective reasoning under both transductive setting and inductive setting.

### 4.5 Ablation Study

To investigate the contribution of each component in our proposed MAKI framework, we conduct an ablation study by comparing the full model against three variants: (1) "w/o TextInit" removes the LLM-aligned textual initialization of entity and relation embeddings, using random initialization instead. (2) "w/o GNN" removes the multi-layer aligned graph structural learning module, relying solely on textual features. (3) "w/o Gate" replaces the gated fusion mechanism with direct addition of structure and LLM features. The results are shown in Table 3 and Table 4.

We can see that removing the LLM-based textual encoding ("w/o TextInit") causes a significant performance drop across all datasets. For example, the accuracy score drops from 87.01% to 83.32% on CoDeX-S; and the MRR drops from 86.45% to 85.39% on the v1 of FB15k237. These results highlight the importance of LLM-aligned knowledge encoding module, which can effectively bridge the semantic gap between structural knowledge in KGs and textual information of LLMs.

| Models | P | R | F1 | Acc |
|---|---|---|---|---|
| MAKI (LLaMA3.1-8B) | 85.70 | 88.84 | 87.24 | 87.01 |
| MAKI (LLaMA2-7B) | 86.28 | 87.69 | 86.98 | 86.87 |
| MAKI (Qwen2-7B) | 85.68 | 87.43 | 86.55 | 86.64 |
| MAKI (Alpaca-7B) | 83.87 | 89.17 | 86.43 | 86.28 |

Table 5: The performance on CoDeX-S dataset with different LLMs as backbone.

When we remove the GNN module ("w/o GNN"), the performance declines even more drastically. This demonstrates the essential role of graph structure learning, as LLMs alone cannot capture the multi-hop relational dependencies embedded in the graph structure.

Replacing the gated fusion mechanism with simple addition ("w/o Gate") also leads to a consistent drop in performance This suggests that uncontrolled integration of structure into LLM may introduce noise features. The gated mechanism effectively balances the contribution from LLM and GNN representations by adaptively filtering useful information at each layer.

These results confirm that each component of the proposed MAKI model plays a critical role. Their combination enables robust and semantically aligned reasoning, leading to better performance.

### 4.6 Performance with Different LLMs

To evaluate the robustness of MAKI with respect to different backbone LLMs, we conduct experiments to compare the performance with four representative LLMs as backbone, including LLaMA3.1-8B, LLaMA2-7B (Touvron et al., 2023b) , Qwen2-7B (Yang et al., 2024), and Alpaca-7B (Taori et al., 2023), and report the results on the CoDeX-S dataset in Table 5. We can observe that MAKI demonstrates consistently strong performance across all LLM backbones, with F1 scores tightly clustered in a narrow range from 86.43% to 87.24% and accuracy varying from 86.28% to 87.01%. This indicates that the proposed method is not overly sensitive to the specific parameterization or pretraining characteristics of the underlying LLM, and can reliably transfer its reasoning capability across different model families.

### 4.7 Impact of Different Layers

We explore the impact of different numbers of GNN layers on the performance of our model. As shown in Figure 3, when the number of layers $k = 2$, the MAKI model achieves a sub-optimal performance on CoDeX-S. This is because with

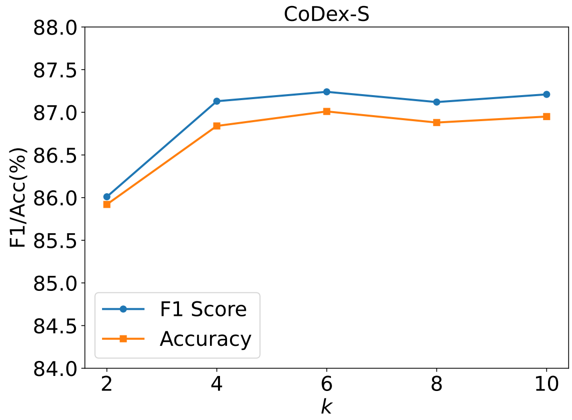


Figure 3: F1 and accuracy performance with different GNN layers (e.g., $k$) on CoDeX-S.

only a few layers, the model fails to inject sufficient structural information into the LLM, limiting its ability to capture complex relational dependencies within KGs. As $k$ increases, performance improves on both F1 and accuracy metrics. Notably, when $k = 6$, the MAKI model reaches its best performance in terms of F1 score and accuracy. However, when we further increase $k$, the performance no longer improves. This is because after a certain number of layers, additional GNN layers start to introduce redundant information or begin to overfit on the graph structure, failing to offer additional benefits for the LLM's reasoning. This suggests that 6 layers strike an optimal balance, effectively capturing the necessary graph structure while avoiding excessive computational overhead. Therefore, we set $k = 6$ in our experiments. More analysis about the dimension size and the case study are provided in Appendix C and Appendix D.

## 5 Conclusion

In this paper, we propose MAKI, a novel method for enhancing KGC using LLMs by injecting structured knowledge from KGs in a semantically aligned manner, bridging the gap between structured and textual representations. Our method first encodes entities and relations using LLMs based on their textual information to ensure alignment in the semantic space. Then, a multi-layer aligned graph structure learning module is applied to learn structural representations. Finally, a gated fusion mechanism is used to inject the structure information into the LLM, allowing the model to benefit from both modalities. Extensive experiments on various benchmark datasets demonstrate that the proposed MAKI outperforms various existing baselines.

## Limitations

Despite the promising results, our approach has several limitations. First, as an LLM-based method, it requires evaluating each candidate triple individually. This leads to low inference efficiency when applied to large-scale link prediction scenarios with a vast number of candidate triples. Second, training our model involves fine-tuning large language models and performing multi-layer alignment with GNNs. This results in considerable computational cost and extended training time, which may hinder deployment in environments with limited resources.

## Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant 62402194, 62377021, the China Postdoctoral Science Foundation under Grant Number 2024M751062, the Natural Science Foundation of Hubei Province for Distinguished Young Scholars (No. 2023AFA096). This work was supported in part by the Macao Young Scholars Program (Grant No. AM2024015), the Science and Technology Development Fund of Macau SAR (Grant Nos. FDCT/0007/2024/AKP, EF2024-00185-FST), the UM and UMDF (Grant Nos. MYRG-GRG2024-00165-FST-UMDF, MYRG-GRG2025-00236-FST), the Tencent AI Lab Rhino-Bird Research Program (Grant No. EF2023-00151-FST), and the Stanley Ho Medical Development Foundation (Grant No. SHMDF-AI/2026/001).

## References

Olivier Bodenreider. 2004. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Res.*, 32(Database-Issue):267–270.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013.*, pages 2787–2795.

Linlin Chao, Jianshan He, Taifeng Wang, and Wei Chu. 2021. PairRE: Knowledge graph embeddings via paired relation vectors. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 4360–4369. Association for Computational Linguistics.

Chen Chen, Yufei Wang, Aixin Sun, Bing Li, and Kwok-Yan Lam. 2023. Dipping plms sauce: Bridging structure and text for effective knowledge graph completion via conditional soft prompting. In *Findings of the Association for Computational Linguistics: ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 11489–11503. Association for Computational Linguistics.

Mingyang Chen, Wen Zhang, Yushan Zhu, Hongting Zhou, Zonggang Yuan, Changliang Xu, and Huajun Chen. 2022. Meta-knowledge transfer for inductive knowledge graph embedding. In *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 927–937. ACM.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2D knowledge graph embeddings. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18)*, pages 1811–1818.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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, NAACL-HLT 2019*, pages 4171–4186. Association for Computational Linguistics.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 12216–12235. Association for Computational Linguistics.

Lingbing Guo, Zhongpu Bo, Zhuo Chen, Yichi Zhang, Jiaoyan Chen, Yarong Lan, Mengshu Sun, Zhiqiang Zhang, Yangyifei Luo, Qian Li, Qiang Zhang, Wen Zhang, and Huajun Chen. 2024. Mkgl: Mastery of a three-word language. In *Advances in Neural Information Processing Systems*, volume 37, pages 140509–140534. Curran Associates, Inc.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Junsik Kim, Jinwook Park, and Kangil Kim. 2025. RSCF: Relation-semantics consistent filter for entity embedding of knowledge graph. In *Proceedings* *of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12320–12336, Vienna, Austria. Association for Computational Linguistics.

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

Stanley Kok and Pedro M. Domingos. 2007. Statistical predicate invention. In *Machine Learning, Proceedings of the Twenty-Fourth International Conference (ICML 2007), Corvallis, Oregon, USA, June 20-24, 2007*, volume 227 of *ACM International Conference Proceeding Series*, pages 433–440. ACM.

Yu Liu, Yanan Cao, Xixun Lin, Yanmin Shang, Shi Wang, and Shirui Pan. 2025. Enhancing large language model for knowledge graph completion via structure-aware alignment-tuning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, Suzhou, China. Association for Computational Linguistics.

Yu Liu, Yanan Cao, Shi Wang, Qingyue Wang, and Guanqun Bi. 2024. Generative models for complex logical reasoning over knowledge graphs. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, WSDM '24, page 492–500, New York, NY, USA. Association for Computing Machinery.

Xiao Long, Liansheng Zhuang, Aodi Li, Minghong Yao, and Shafei Wang. 2025. EPERM: an evidence path enhanced reasoning model for knowledge graph question and answering. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pages 12282–12290. AAAI Press.

Xin Lv, Yankai Lin, Yixin Cao, Lei Hou, Juanzi Li, Zhiyuan Liu, Peng Li, and Jie Zhou. 2022. Do pretrained models benefit knowledge graph completion? A reliable evaluation and a reasonable approach. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 3570–3581. Association for Computational Linguistics.

Sijie Mai, Shuangjia Zheng, Yuedong Yang, and Haifeng Hu. 2021. Communicative message passing for inductive relation reasoning. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021*, pages 4294–4302. AAAI Press.

Fanshen Meng, Zhenhua Meng, Ru Jin, Rongheng Lin, and Budan Wu. 2025. DOGE: llms-enhanced hyperknowledge graph recommender for multimodal recommendation. In *AAAI-25, Sponsored by the Association for the Advancement of Artificial Intelligence, February 25 - March 4, 2025, Philadelphia, PA, USA*, pages 12399–12407. AAAI Press.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

Moritz Plenz and Anette Frank. 2024. Graph language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4477–4494, Bangkok, Thailand. Association for Computational Linguistics.

Tara Safavi and Danai Koutra. 2020. Codex: A comprehensive knowledge graph completion benchmark. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 8328–8350. Association for Computational Linguistics.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. RotatE: Knowledge graph embedding by relational rotation in complex space. In *7th International Conference on Learning Representations*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Komal K. Teru, Etienne G. Denis, and William L. Hamilton. 2020. Inductive relation prediction by subgraph reasoning. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 9448–9457. PMLR.

Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury, and Michael Gamon. 2015. Representing text for joint embedding of text and knowledge bases. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1499–1509.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models. *CoRR*, abs/2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, and 49 others. 2023b. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

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

Liang Wang, Wei Zhao, Zhuoyu Wei, and Jingming Liu. 2022. Simkgc: Simple contrastive knowledge graph completion with pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022*, pages 4281–4294. Association for Computational Linguistics.

Weihua Wang, Qiuyu Liang, Feilong Bao, and Guanglai Gao. 2025. Distance-adaptive quaternion knowledge graph embedding with bidirectional rotation. In *Proceedings of the 31st International Conference on Computational Linguistics, COLING 2025, Abu Dhabi, UAE, January 19-24, 2025*, pages 4219–4231. Association for Computational Linguistics.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, and 40 others. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

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 *3rd International Conference on Learning Representations*.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. KG-BERT: BERT for knowledge graph completion. *CoRR*, abs/1909.03193.

Liang Yao, Jiazhen Peng, Chengsheng Mao, and Yuan Luo. 2023. Exploring large language models for knowledge graph completion. *CoRR*, abs/2308.13916.

Huanyu Zhang, Xiaoxuan Shen, Baolin Yi, Jianfang Liu, and Yinao Xie. 2025. A plug-in critiquing approach for knowledge graph recommendation systems via representative sampling. In *Proceedings of the ACM on Web Conference 2025, WWW 2025, Sydney, NSW, Australia, 28 April 2025- 2 May 2025*, pages 322–333. ACM.

Yichi Zhang, Zhuo Chen, Lingbing Guo, Yajing Xu, Wen Zhang, and Huajun Chen. 2024. Making large language models perform better in knowledge graph completion. In *Proceedings of the 32nd ACM International Conference on Multimedia, MM 2024, Melbourne, VIC, Australia, 28 October 2024 - 1 November 2024*, pages 233–242. ACM.

Qiannan Zhu, Haobo Zhang, Qing He, and Zhicheng Dou. 2024. Query-aware explainable product search with reinforcement knowledge graph reasoning. *IEEE Trans. Knowl. Data Eng.*, 36(3):1260–1273.

# A Dataset Details

For the triple classification task, we use UMLS (Kok and Domingos, 2007; Yao et al., 2019), CoDeX-S (Safavi and Koutra, 2020), and FB15K-237N (Lv et al., 2022) datasets. The UMLS dataset is a biomedical KG that contains concepts and relations derived from the Unified Medical Language System (Bodenreider, 2004). CoDex-S is a subset of the CoDEx (Safavi and Koutra, 2020) benchmark which is designed to assess models that leverage both structural and textual features for reasoning. The FB15K-237N dataset is derived from FB15K-237 (Toutanova et al., 2015) by removing relations that contain mediator (CVT) nodes. The detailed statistics of the datasets UMLS, CoDeX-S and FB15K-237N are listed in Table 6.

For the link prediction task, we use inductive versions of WN18RR and FB15K-237 from GraIL (Teru et al., 2020). These datasets are derived from the original KGs WN18RR (Dettmers et al., 2018) and FB15K-237 (Toutanova et al., 2015), respectively. Each dataset provides four variants with different scales to enable robust and comprehensive evaluation. For each variant, the data are partitioned into two disjoint subgraphs sampled from the original KG, corresponding to the training and test graphs. Table 7 provides the statistics of these datasets.

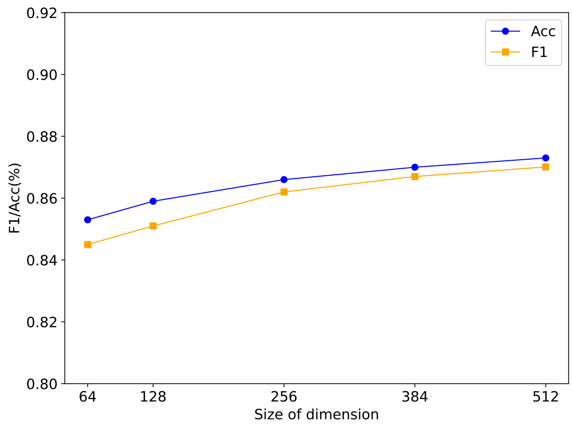


Figure 4: F1 and accuracy performance with different sizes of dimension on CoDeX-S.

# B Hits@10 Results for Link Prediction

Table 8 reports the hits@10 results on inductive versions of FB15K-237 and WN18RR. Structure-based methods perform moderately, especially on FB15K-237 (up to 95.92% for MorsE on v3 of

| Dataset | #Ent | #Rel | #Train | #Valid (+/-) | #Test (+/-) |
|---|---|---|---|---|---|
| UMLS | 135 | 46 | 5216 | 652/652 | 661/661 |
| CoDeX-S | 2023 | 42 | 32888 | 1827/1827 | 1828/1828 |
| FB15K-237N | 13104 | 93 | 87282 | 7041/7041 | 8226/8226 |

Table 6: Statistics of the datasets UMLS, CoDeX-S and FB15K-237N. #Ent and #Rel denote the numbers of entities and relations, respectively. The ratios of positive (+) and negative (-) samples are 1:1 in the valid and test datasets.

| | | WN18RR | | | FB15K-237 | | |
|---|---|---|---|---|---|---|---|
| | | #Rel | #Ent | #Triple | #Rel | #Ent | #Triple |
| v1 | train | 9 | 2746 | 6678 | 183 | 2000 | 5226 |
| | test | 9 | 922 | 1991 | 146 | 1500 | 2404 |
| v2 | train | 10 | 6954 | 18968 | 203 | 3000 | 12085 |
| | test | 10 | 2923 | 4863 | 176 | 2000 | 5092 |
| v3 | train | 11 | 12078 | 32150 | 218 | 4000 | 22394 |
| | test | 11 | 5084 | 7470 | 187 | 3000 | 9137 |
| v4 | train | 9 | 3861 | 9842 | 222 | 5000 | 33916 |
| | test | 9 | 7208 | 15157 | 204 | 3500 | 14554 |

Table 7: Statistics of the inductive versions of WN18RR and FB15K-237 datasets."#Rel", "#Ent" and "#Triple" are the numbers of relations, entities and triples, respectively.

| Models | **FB15K-237** | | | | **WN18RR** | | | |
|---|---|---|---|---|---|---|---|---|
| | v1 | v2 | v3 | v4 | v1 | v2 | v3 | v4 |
| GraIL | 63.49 | 79.82 | 78.19 | 82.72 | 83.51 | 80.61 | 58.67 | 76.34 |
| ComPILE | 67.32 | 80.54 | 83.35 | 84.41 | 81.56 | 80.89 | 59.87 | 75.56 |
| MorsE | 80.73 | 95.11 | 95.92 | 92.72 | 79.57 | 79.77 | 69.93 | 77.07 |
| KG-BERT | 74.39 | 82.74 | 81.04 | 76.76 | 92.83 | 93.65 | 92.76 | 93.18 |
| PKGC | 78.42 | 83.92 | 82.17 | 77.24 | 93.48 | 94.51 | 93.12 | 95.07 |
| KG-LLaMA | 98.99 | 97.59 | 98.78 | 98.21 | 98.41 | 98.09 | 98.45 | 99.65 |
| KG-Alpaca | 98.78 | 97.91 | 98.78 | 98.42 | 99.16 | 98.87 | 98.65 | 99.12 |
| Vanilla IT | 99.26 | 98.64 | 98.95 | 98.84 | 99.26 | 99.12 | 98.85 | 99.28 |
| Structure-aware IT | 98.78 | 98.43 | 98.96 | 98.87 | 99.32 | 99.16 | 99.17 | 99.54 |
| **MAKI** | **99.51** | **99.79** | **99.54** | **99.36** | **99.47** | **99.86** | **99.17** | **99.69** |

Table 8: Hits@10 results on inductive version of FB15K-237 and WN18RR datasets. The best results are in bold.

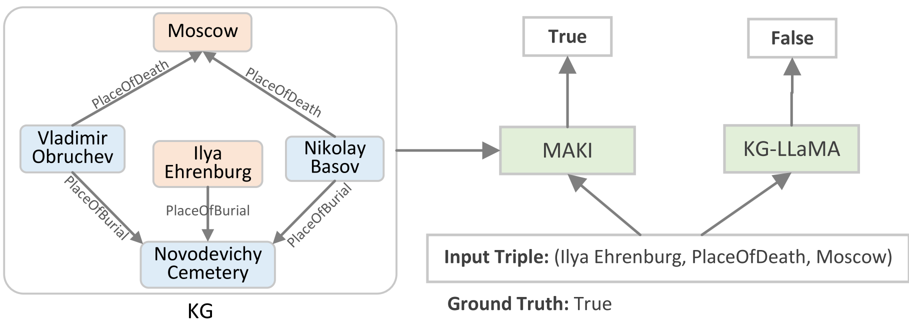


Figure 5: An example of the prediction of MAKI and KG-LLaMA model on CoDeX-S dataset.

FB15K-237), but lag on WN18RR (mostly below 85%). PLM-based methods show stronger semantic reasoning, particularly on WN18RR (up to 95.07% for PKGC on v4 of WN18RR), yet remain below LLM-based approaches. LLM-based models consistently achieve superior results across all datasets, with nearly all scores exceeding 97%. Our proposed MAKI obtains the best performances across all these datasets, demonstrating its effectiveness in integrating structural and semantic knowledge for robust reasoning.

## C Analysis of Dimension Size

Figure 4 illustrates the impact of hidden dimension size of GNNs on model performance on the CoDeX-S dataset, measured by Accuracy and F1 score. As the size of dimension increases from 64 to 512, both metrics exhibit a consistent and monotonic improvement, indicating that higher-dimensional representations enable the model to capture richer semantic and structural information. Notably, the performance gains are more pronounced when increasing the dimension from 64 to 256, suggesting that low-dimensional embeddings may be insufficient to fully model the relational complexity of the KG. Beyond 384 dimensions, the improvements become marginal. The highest Accuracy and F1 scores are achieved with a dimension size of 512. Based on this observation, we adopt 512 as the default dimension in our experiments to ensure optimal performance.

## D Case Study

Figure 5 presents a qualitative comparison between MAKI and KG-LLaMA on the CoDeX-S dataset. The input query is the triple (*Ilya Ehrenburg*, *PlaceOfDeath*, *Moscow*), whose ground truth label is *True*. As shown on the left, the underlying knowledge graph contains multiple semantically related entities and relations, including *Vladimir Obruchev* and *Nikolay Basov*, both of whom share the same *PlaceOfDeath* relation with *Moscow*, as well as *Novodevichy Cemetery*, which is connected via the *PlaceOfBurial* relation. By effectively aggregating and reasoning over these relational patterns and contextual entity connections, MAKI successfully infers the correct label for the query. In contrast, KG-LLaMA predicts *False*, indicating it is difficult determine the correctness of this give triple without leveraging structured graph evidence and relational dependencies. This example highlights MAKI's stronger capability in exploiting local graph structures and relational semantics for KGC.