



Latest updates: <https://dl.acm.org/doi/10.1145/3711896.3737011>

RESEARCH-ARTICLE

K-Paths: Reasoning over Graph Paths for Drug Repurposing and Drug Interaction Prediction

TASSALLAH ABDULLAHI, Brown University, Providence, RI, United States

IOANNA GEMOU, Brown University, Providence, RI, United States

NIHAL V NAYAK, Brown University, Providence, RI, United States

GHULAM MURTAZA, Brown University, Providence, RI, United States

STEPHEN H BACH, Brown University, Providence, RI, United States

CARSTEN EICKHOFF, University of Tübingen, Tubingen, Baden-Wurttemberg, Germany

[View all](#)

Open Access Support provided by:

Brown University

University of Tübingen



PDF Download
3711896.3737011.pdf
15 January 2026
Total Citations: 1
Total Downloads: 1482

Published: 03 August 2025

[Citation in BibTeX format](#)

KDD '25: The 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining

August 3 - 7, 2025
Toronto ON, Canada

Conference Sponsors:

SIGMOD
SIGKDD

K-Paths: Reasoning over Graph Paths for Drug Repurposing and Drug Interaction Prediction

Tassallah Abdullahi

tassallah_abdullahi@brown.edu

Brown University

Providence, RI, USA

Ghulam Murtaza

ghulam_murtaza@brown.edu

Brown University

Providence, RI, USA

Ioanna Gemou[†]

ioanna_gemou@brown.edu

Brown University

Providence, RI, USA

Stephen H. Bach

stephen_bach@brown.edu

Brown University

Providence, RI, USA

Nihal V. Nayak

nihal_vivekanand_nayak@brown.edu

Brown University

Providence, RI, USA

Carsten Eickhoff^{*}

carsten.eickhoff@uni-tuebingen.de

University of Tübingen

Tübingen, Germany

Ritambhara Singh^{*}

ritambhara@brown.edu

Brown University

Providence, RI, USA

enabling scalable and explainable LLM-augmented scientific discovery. We release our code and the retrieved paths as a benchmark for inductive reasoning.¹

CCS Concepts

- Computing methodologies → Knowledge representation and reasoning.

Keywords

Knowledge graph reasoning, Drug discovery, LLMs, GNNs, Explainability, Inductive reasoning

ACM Reference Format:

Tassallah Abdullahi, Ioanna Gemou[†], Nihal V. Nayak, Ghulam Murtaza, Stephen H. Bach, Carsten Eickhoff, and Ritambhara Singh. 2025. K-Paths: Reasoning over Graph Paths for Drug Repurposing and Drug Interaction Prediction. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '25), August 3–7, 2025, Toronto, ON, Canada*. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/3711896.3737011>

1 Introduction

Drug development and safety assessment have traditionally been time-intensive and costly, often spanning years and requiring billions of dollars [11, 31]. Recent advances in computational power and the availability of clinical and biological data are transforming this landscape, enabling faster, more cost-effective discovery and validation of safer drugs [21, 36]. For instance, deep learning models trained on genomic and chemical datasets now predict drug efficacy and toxicity with high accuracy, reducing the need for extensive lab experiments [8]. Yet, the sheer volume and heterogeneity of these datasets pose integration challenges, making it difficult to extract meaningful insights [9, 17].

Knowledge Graphs (KGs) offer a structured solution by integrating complex biological relationships, linking diseases, drugs, and proteins into an interconnected framework [20, 24]. Despite their potential, the scale and complexity of KGs make it difficult

Abstract

Biomedical knowledge graphs (KGs) encode rich, structured information critical for drug discovery tasks, but extracting meaningful insights from large-scale KGs remains challenging due to their complex structure. Existing biomedical subgraph retrieval methods are tailored for graph neural networks (GNNs), limiting compatibility with other paradigms, including large language models (LLMs). We introduce K-Paths, a model-agnostic retrieval framework that extracts structured, diverse, and biologically meaningful multi-hop paths from dense biomedical KGs. These paths enable prediction of unobserved drug–drug and drug–disease interactions, including those involving entities not seen during training, thus supporting inductive reasoning. K-Paths is training-free and employs a diversity-aware adaptation of Yen’s algorithm to extract the K shortest loopless paths between entities in a query, prioritizing biologically relevant and relationally diverse connections. These paths serve as concise, interpretable reasoning chains that can be directly integrated with LLMs or GNNs to improve generalization, accuracy, and enable explainable inference. Experiments on benchmark datasets show that K-Paths improves zero-shot reasoning across state-of-the-art LLMs. For instance, Tx-Gemma 27B improves by 19.8 and 4.0 F1 points on interaction severity prediction and drug repurposing tasks, respectively. Llama 70B achieves gains of 8.5 and 6.2 points on the same tasks. K-Paths also boosts the training efficiency of EmerGNN, a state-of-the-art GNN, by reducing the KG size by 90% while maintaining predictive performance. Beyond efficiency, K-Paths bridges the gap between KGs and LLMs,

^{*}co-corresponding authors.

[†]Also affiliated with the Technical University of Denmark.



This work is licensed under a Creative Commons Attribution 4.0 International License.
KDD '25, Toronto, ON, Canada

© 2025 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-1454-2/2025/08

<https://doi.org/10.1145/3711896.3737011>

¹<https://github.com/rsinghlab/K-Paths>

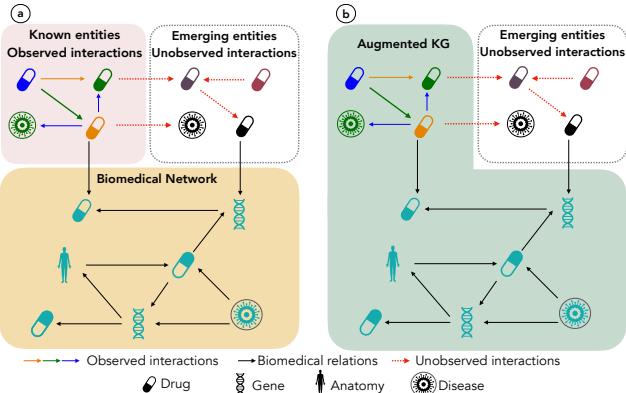


Figure 1: Schematic representation of the predictive framework for unobserved interactions. (a) Problem formulation: Given a network of observed interactions among known entities (drugs or diseases) and a broader biomedical network containing additional relationships between various entities (e.g., drugs, diseases, genes, etc.). The task is to predict unobserved interactions between a known entity and an emerging entity or two emerging entities. **(b) Augmented KG:** The observed interactions and biomedical network are integrated to create a richer representation for the task.

to retrieve task-related information efficiently [32]. While typical KGs contain thousands of nodes and millions of edges, only a small subgraph is relevant for a given task [45], limiting practical utility in domains like drug discovery, where precise, targeted insights are critical. Graph Neural Networks (GNNs) have proven effective in leveraging KGs for drug discovery, excelling in link prediction and capturing biological relationships [21, 34, 47]. However, GNNs incur high computational costs on large-scale KGs [1] and critically fail to generalize to unseen entities [18], a drawback in drug discovery where new drugs and diseases continuously emerge.

Large Language Models (LLMs) offer a promising alternative, excelling in zero-shot reasoning and inductive generalization [3, 4, 23]. Recent works have shown that LLMs can be grounded in KGs to enhance factual accuracy and reduce hallucinations in specialized domains [2, 3, 14, 41]. However, extracting meaningful insights from KGs using LLMs is an ongoing challenge [15, 33]. Current approaches to KG integration often exhibit limitations. General-purpose methods like [29, 30, 35] focus on narrow tasks such as question answering or binary classification, using extraction methods that struggle to scale to dense biomedical KGs. Conversely, biomedical-specific approaches like [46] are tightly coupled to GNN architectures, hindering their direct integration with modern LLMs without substantial modifications. Addressing these limitations is crucial for developing a generalizable and scalable framework for drug discovery.

We introduce “**K-Paths**”, a novel model-agnostic retrieval framework designed to extract highly relevant entities and relationships from large biomedical KGs to aid in predicting unobserved drug–disease and drug–drug interactions, including those involving entities unseen during training. Unlike traditional approaches, K-Paths generates structured, interpretable multi-hop paths directly usable by LLMs, enabling efficient and accurate zero-shot reasoning.

K-Paths is particularly valuable in drug discovery, where identifying such interactions can facilitate drug repurposing and safer treatment opportunities. Figure 1 illustrates the problem K-Paths addresses: we aim to predict unobserved interactions between known and emerging entities (or between two emerging entities), where no interaction of interest has been observed previously. This formulation establishes an inductive reasoning setting, enabling predictions beyond observed edges.

K-Paths is training-free and operates in three steps as shown in fig. 2. First, it augments the biomedical KG by integrating observed interactions from training data, enhancing its representation of interactions. This enriched KG serves as the network for retrieving biologically meaningful connections. Next, it retrieves biologically meaningful connections using our diversity-aware adaptation of Yen’s algorithm [44]. While Yen’s algorithm iteratively finds alternative shortest paths, our adaptation prioritizes relational diversity and biological relevance over redundant shortest-path variations. Finally, the retrieved paths are transformed into natural language representations, enabling LLMs to reason over them and effectively predict interactions. These paths can also be used to construct task-specific subgraphs, allowing GNN models to operate on a more focused graph and substantially reducing computational overhead.

We evaluate K-Paths in both zero-shot generative and supervised learning settings on drug repurposing and drug–drug interaction (DDI) tasks involving emerging entities. Experiments show that K-Paths significantly enhances LLM reasoning in a zero-shot setting: on DDI severity prediction, Tx-Gemma 27B improves by 19.8 F1 points, and on drug repurposing tasks, by 4.0. Llama 70B achieves similar gains of 8.5 and 6.2 points on the same tasks, respectively. In supervised settings, K-Paths reduces KG size by 90% and improves training efficiency for EmerGNN, a state-of-the-art GNN model, [46], without significant loss in performance. More importantly, the retrieved paths provide interpretable rationales for predicted interactions, enhancing explainability and offering valuable biological insights.

2 Related work

2.1 Biomedical KGs

Biomedical KGs have been widely used to model the complex relationships among drugs, diseases, genes, and other biological entities, enabling data-driven advances in drug discovery [21, 38, 45, 47]. These large-scale KGs are often curated for specialized purposes such as drug-disease interactions [20, 40], gene-drug interactions [39], and protein-protein interactions [16]. In this work, we focus on drug-disease interactions and drug-drug interactions as they play a critical role in drug discovery, clinical decision-making, and patient safety. A key limitation of biomedical KGs is that they can be incomplete in several ways. They may lack observed interactions between existing entities (e.g., drugs or diseases) or fail to capture interactions involving emerging entities—those newly introduced or underrepresented in the KG. We focus on predicting unobserved interactions among existing and emerging drugs or diseases, as well as interactions involving emerging drugs or diseases. This setting requires inductive reasoning, where a model must generalize to unseen nodes [46], as opposed to transductive reasoning, which assumes full knowledge of the graph at training time. The inductive

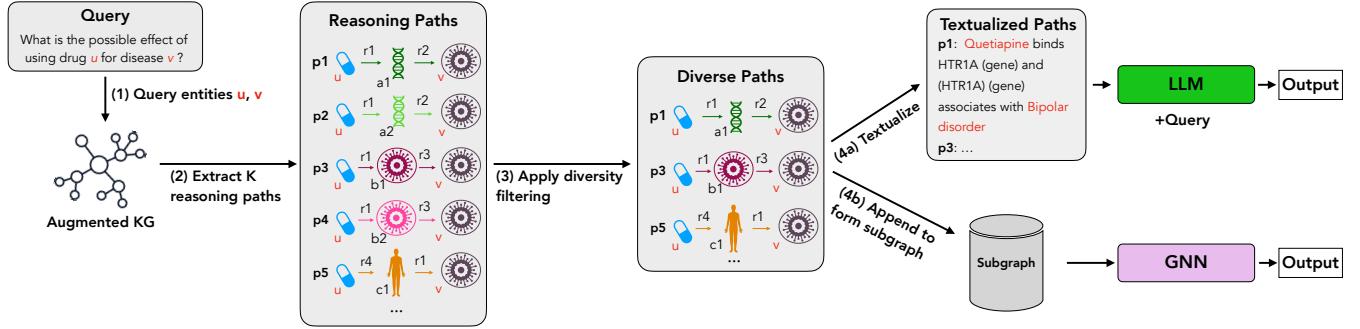


Figure 2: K-Paths Overview. (1) Given a query about the effect of an entity (u) on another entity (v), (2) K-Paths extracts reasoning paths from an augmented KG connecting (u) and (v). (3) These paths are filtered for diversity and (4a) transformed into natural language descriptions for LLM inference. (4b) The retrieved paths can also be used to construct a subgraph, enabling GNNs to leverage more manageable information for training and prediction.

nature of this challenge reflects real-world biomedical discovery, where new drugs and diseases are continuously introduced.

2.2 GNNs for drug discovery

GNNs, including variants of graph convolutional networks, have demonstrated strong performance in modeling the structure of biomedical KGs and predicting unobserved interactions between entities [27, 38, 46]. These models often leverage external KGs such as Hetionet [20] to learn richer representations of biomedical entities. However, training on large-scale KGs is computationally expensive. To address scalability, recent methods like [38, 45] extract fixed-sized subgraphs before passing them to the graph network. While effective, these approaches are typically designed for transductive settings, where both nodes involved in a prediction are observed during training. This limits their ability to generalize to emerging entities, a critical need in drug discovery. In this work, we operate in the inductive setting, aiming to predict interactions involving unseen drugs or diseases.

Our work is closely related to EmerGNN [46], which augments the training network with Hetionet and uses a flow-based model to extract paths and construct subgraphs for GNN processing. However, EmerGNN requires a separately trained model and relies on beam search to generate paths tailored for graph architectures. In contrast, K-Paths is training-free, retrieves diverse and biologically meaningful paths with minimal computational overhead, and supports model-agnostic integration with GNNs and LLMs. This flexibility enables K-Paths to support both supervised and zero-shot learning. Furthermore, in section 5, we show that integrating K-Paths into EmerGNN improves efficiency and performance compared to using the entire KG. K-Paths also significantly improves LLM zero-shot performance without any parameter updates.

2.3 LLMs for drug discovery

LLMs have emerged as powerful tools in biomedical research, with applications ranging from conversational agents for drug discovery [37], and drug repurposing [22], to molecular understanding [25]. A key factor in their success is in-context learning [7],

which enables LLMs to incorporate new information, such as retrieved examples or structured facts, at inference time without fine-tuning. This ability is particularly valuable for multi-hop reasoning, where models synthesize information across multiple steps or sources, such as KG paths, to derive accurate conclusions [30]. Recent studies show that LLMs can reason faithfully over KG paths to answer complex factual queries [29, 30]. However, this potential remains underexplored in the context of biomedical KGs, particularly in inductive settings, where multi-hop reasoning could improve tasks like predicting interactions involving novel drugs or emerging diseases.

Graph-Augmented LLMs: K-Paths builds upon emerging research at the intersection of retrieval-augmented generation (RAG) and graph-based reasoning [14, 29]. Existing approaches construct synthetic KGs from unstructured text using LLMs [14, 41] or rely on general-domain KGs like Freebase [29] for question answering. In contrast, K-Paths leverages curated biomedical KGs to enable precise, domain-specific inference. Moreover, existing systems like RoG [29] and GNN-RAG [30] require trained planning/retrieval modules or exhaustive path enumeration (e.g., KG-LLM's [35] depth-first search), which are computationally expensive and difficult to scale to dense KGs. K-Paths circumvents these limitations by using a training-free, heuristic adaptation of Yen's algorithm, with a filtering step for efficiency and relational diversity.

Design Trade-offs: RoG, GNN-RAG, and KG-LLM represent three distinct paradigms for path-based LLM reasoning, but each faces limitations for full-graph biomedical inference. RoG and GNN-RAG's retrieval modules operate on query-specific subgraphs generated via PageRank [5] and assume answers exist within the local subgraph. This assumption may break down in inductive settings, where test-time entities may not appear during training. Additionally, their reliance on pre-generated subgraphs limits scalability to large biomedical KGs. Similarly, KG-LLM is limited to a binary setup, and its exhaustive path enumeration over the full KG becomes infeasible in high-degree biomedical graphs. In contrast, K-Paths enables scalable, explainable inference by retrieving diverse, biologically meaningful paths, without training or exhaustive search. We therefore analyze these approaches conceptually rather than

as direct baselines, because they differ significantly in reasoning scope, scalability, and modeling assumptions.

3 Approach

3.1 Problem definition

Building on the problem illustrated in fig. 1, we aim to predict unobserved interactions in an inductive reasoning setup, as described in Zhang et al. [46]. These interactions can involve:

- A known entity (e.g., a well-studied drug or disease with some observed interactions) and an emerging entity (e.g., a drug or disease whose interactions of interest have not been observed), or
- Two emerging entities.

The unobserved interactions may include drug-drug interactions or drug-disease interactions.

Formally, we define a knowledge graph $\mathcal{G} = \{(u, r, v) \mid u, v \in \mathcal{E}, r \in \mathcal{R}\}$ where $u, v \in \mathcal{E}$ represent biomedical entities (e.g., drugs, diseases, genes) and $r \in \mathcal{R}$ denotes a relation type. These relation types include known drug-drug and drug-disease interactions (observed interactions) and a broader biomedical network with relationships like drug-gene or gene-gene interactions.

Given two query entities u and v , our goal is to infer their interaction type, which is framed as predicting the presence and type of relation r between u and v . We define a computational model $\phi(\mathcal{G})$ to predict these interactions under both zero-shot generative and supervised settings. Specifically, we leverage LLMs for reasoning-based inference and GNNs for interaction prediction.

3.2 K-Paths framework

We introduce **K-Paths**, a framework for predicting the interaction between entities u and v . K-Paths comprises three key components:

- (1) An *augmented KG*: This module constructs a knowledge graph \mathcal{G} by integrating known drug-drug or drug-disease interactions with a broader biomedical KG. This integration incorporates additional entities such as genes and their known relationships (e.g., drug-gene or gene-gene interactions).
- (2) A *diverse path retrieval module*: This module employs a novel path retrieval algorithm to retrieve a diverse set of relevant reasoning paths connecting the query entities u and v from the augmented KG \mathcal{G} .
- (3) A *path integration module*: This module processes the retrieved query-specific paths for interaction prediction. For LLM-based reasoning and interaction inference, the paths are transformed into natural language and appended to the interaction query prompt. For GNN-based interaction prediction, the paths are reconstructed into query-specific subgraphs.

The overall K-Paths framework is illustrated in fig. 2.

3.3 Augmented KG

Following prior work [38, 45, 46], we define the *augmented KG* as our knowledge graph \mathcal{G} , constructed by integrating:

- (1) Observed interactions (e.g., drug-drug and drug-disease) from the training set.

- (2) Hetionet, a biomedical knowledge graph containing biological entities (e.g., genes, proteins, pathways) and their relationships (e.g., drug-gene, gene-gene, protein-protein interactions) [20].

Since Hetionet is incomplete, incorporating known interactions from the training set enhances coverage and helps build a more comprehensive augmented KG. However, interactions involving emerging entities remain missing in this graph. To address this, we leverage existing relationships within the augmented KG to infer these unobserved interactions. Additionally, following [46], we incorporate inverse relations to account for the directed nature of the augmented KG, ensuring bidirectional information flow.

The augmented KG \mathcal{G} serves as the structured knowledge graph for all subsequent tasks.

Algorithm 1: Filtering algorithm to remove relationally redundant retrieved paths.

```

Input: A set of paths  $P = \{p_1, p_2, \dots, p_K\}$ , where each path
       $p$  is described by
       $R(p) = (r_1, r_2, \dots, r_l)$  (sequence of relations) and
       $\mathcal{E}(p) = (e_1, e_2, \dots, e_m)$  (sequence of entities).
Output: A subset  $P' \subseteq P$  with redundant paths removed.

Initialize  $P' \leftarrow \emptyset$ ;
foreach path  $p \in P$  do
  | Let  $R(p) = (r_1, r_2, \dots, r_l)$  be the relation sequence of  $p$ ;
  | Let  $l = |R(p)|$  (the length of the relation sequence);
  | if  $\nexists q \in P'$  such that  $R(q) = R(p)$  and  $|R(q)| = l$  then
    |    $P' \leftarrow P' \cup \{p\}$ ;
return  $P'$ 

```

3.4 Diverse path retrieval module

The path retrieval module is a key component of our framework. It provides the downstream computational model with highly relevant yet manageable information from our augmented KG \mathcal{G} .

Our diverse path retrieval algorithm retrieves a set of K shortest diverse paths between two entities, $\{u, v\}$, from \mathcal{G} using Yen's algorithm [44]. We prioritize shortest paths for several reasons. Shorter paths capture stronger, more interpretable relationships, while longer paths introduce noise and uncertainty [6, 26]. Finally, empirical studies on biomedical KGs further show that meaningful interactions typically occur within a few hops [19, 47].

Yen's algorithm extends Dijkstra's algorithm [10] by iteratively computing shortest paths while temporarily excluding specific edges, thus generating progressively longer, loop-free alternatives. The output from this process contains relationally redundant paths. Therefore, we introduce a filtering algorithm (algorithm 1) to remove these redundancies. Our filtering algorithm eliminates paths with duplicate relation sequences of the same path length. This filtering step results in a diverse set of paths, which are passed to the next module in our framework.

Edge cases: In cases where only one of $\{u, v\}$ exists in \mathcal{G} , we retrieve its immediate neighbors and their connecting relationships,

applying the same filtering algorithm. This provides valuable context even when a complete path between two entities cannot be established. If neither entity exists in \mathcal{G} , no information is retrieved.

3.5 Path integration module

This module integrates the retrieved diverse paths to predict interactions between the query entities. We explore two distinct approaches: LLM-based reasoning and GNN-based prediction.

LLM reasoning We convert the entities and relations from the diverse paths into natural language using predefined dictionaries. These dictionaries map entities and relations to their respective types and textual representations. To improve clarity, we append entity type descriptors in parentheses after each entity.

For example, as illustrated in fig. 2, consider the path:

$$p_1 : u \rightarrow r_1 \rightarrow a_1 \rightarrow r_2 \rightarrow v$$

representing the relationship between *Quetiapine* and *Bipolar disorder*. This path is transformed into the natural language description:

Quetiapine binds HTR1A (gene) and HTR1A (gene) associates with Bipolar disorder.

Following [46], if the retrieved relation belongs to an inverse relation category, we convert it into passive voice. For instance, the relation:

(Disease) downregulates (Gene)

is converted into:

(Gene) is downregulated by (Disease).

This explicit type mapping helps the LLM understand the semantic roles of each entity, even if it is unfamiliar with domain-specific entities like “HTR1A”. This conversion process is applied to all K retrieved paths. In the zero-shot setting, these textualized paths are appended to the original query, providing contextual information for the LLM to perform inference. In the supervised setting, we can fine-tune the LLM using these textualized paths as training data. Furthermore, representing the retrieved paths in natural language enhances the explainability of the LLM’s predictions.

GNN interaction prediction For GNN-based prediction, instead of directly inputting the large augmented KG into the GNN, we use the diverse paths to construct smaller, query-specific subgraphs. This approach significantly reduces the computational complexity and allows GNN to focus on the most relevant information. During training, the GNN learns entity-specific representation by aggregating information from known drug-drug or drug-disease interactions and their corresponding query-specific subgraphs. These learned representations are then used to predict the type of interaction between entity pairs. During testing, we extend the learned representation space by incorporating new test nodes and their corresponding query-specific subgraphs. This allows us to evaluate the model’s ability to accurately predict unobserved interactions using this extended graph in a supervised inductive setting.

4 Experiments

In this section, we aim to answer the following research questions:

RQ1: Can LLMs accurately predict interaction *types*, *severities*, or *indications* by reasoning over multi-hop knowledge graph paths provided as context in a zero-shot setting?

RQ2: How do path selection strategies (e.g., shortest path, diverse path selection) and the number of reasoning paths (K) influence LLM performance in interaction prediction tasks?

RQ3: How do subgraphs derived from the K-Paths framework impact the performance of LLMs and GNNs in a supervised setting?

Table 1: Datasets Overview. u and v are drugs or disease entities.

Dataset	Entities	Categories	Example
DDInter	1689 drugs	3 severity levels	Severity: Major ($u + v$)
DrugBank	1710 drugs	86 interaction levels	u decreases v ’s excretion rate
PharmacotherapyDB	601 drugs; 97 diseases	3 indications	u treats v

4.1 Datasets

Evaluation datasets: Following previous work, we conducted experiments in an inductive setting to assess the model’s generalization to unseen entities. In this setting, the training set includes interactions only between entities in the training set, the validation set contains interactions where at least one entity appears only in the validation set or interactions exclusively within it, and the test set follows the same rule for test entities. This setup enables evaluation of the model’s ability to predict unobserved interactions involving both known and emerging entities, as well as interactions between emerging entities.

We used three datasets with different task objectives (Table 1): DDInter [42], PharmacotherapyDB (v1.0) [19], and DrugBank [40]. For DrugBank, we applied the inductive split from [12], while for DDInter, we created train, validation, and test sets following the described inductive split. Due to its smaller size, PharmacotherapyDB was divided into training and test sets in a similar inductive setting. Table 1 provides detailed statistics, including the number of specific interaction types.

External knowledge base: Hetionet is a heterogeneous biomedical network curated from 29 databases. It includes various biomedical entities such as compounds, genes, diseases, etc. Following [45], the processed version used in this work includes 33,765 nodes across 11 types and 1,690,693 edges spanning 23 relation types.

4.2 Baselines

We compare the K-Paths framework with the following baselines:

4.2.1 LLM-based baselines: Here, we evaluated K-Paths’ impact on LLM reasoning in zero-shot and supervised settings.

- **Reasoning based on internal knowledge (Base):** The LLM uses only its internalized knowledge to infer interactions and relationships.
- **Reasoning based on textual definitions (Definitions):** The LLM leverages textual definitions of drugs and diseases from external resources, such as DrugBank² and the Disease Ontology³, to infer interactions and relationships independent of the knowledge graph’s structure.

²<https://go.drugbank.com/>

³<https://disease-ontology.org/>

4.2.2 GNN-based baselines: Here, we evaluated K-Paths’ impact on GNNs in a supervised setting only.

- **Complete graph-based prediction (Complete KG):** The Graph Neural Network (GNN) uses the entire augmented KG to predict interactions, considering all entities and relationships.

4.3 Implementation details

We provide the details for K-Paths implementation for all our experiments with different LLMs and GNNs. For further details on prompting strategies, training configurations, and hyperparameters, see appendix A. During path retrieval, we limit the maximum path length to 3 as longer paths tend to be more susceptible to noise and less likely to represent meaningful, interpretable relationships. We assume there is no direct link between the query entities whose interaction we aim to predict, and the model must infer the interaction type by reasoning over existing facts in the KG. To prevent data leakage, we explicitly check and remove direct interaction links between entities during training and verify their absence in the test set. For all experiments across datasets, we set $K = 10$ retrieved paths per query. As K increases, both the number of hops and average path length grow accordingly.

For LLM-based reasoning, we employ a suite of models **LLaMA-3.1-8B-Instruct** and **LLaMA-3.1-70B-Instruct** [13], **Tx-Gemma-9B-Chat** and **Tx-Gemma-27B-Chat** [37], as well as **Qwen2.5-14B-Instruct** and **Qwen2.5-32B-Instruct** [43] as our primary inference models, leveraging their instruction-following capabilities. In zero-shot settings, we utilize direct prompting to evaluate their reasoning ability. In supervised settings, we fine-tuned Llama-3.1-8B-Instruct using QLoRA, a lightweight and efficient fine-tuning approach. In both experiments, we generate the output with greedy decoding. For GNN architectures, we use Relational Graph Convolutional Networks (RGCN) because of their ability to handle relational data, which aligns with our task. Additionally, we adopt EmerGNN’s backbone architecture, as it is the current state-of-the-art inductive graph-based model for drug-drug interaction prediction tasks.

4.4 Evaluation

Following prior work [46], we evaluate all GNN and LLM models on a multi-class classification task using accuracy, macro-averaged F1-score, and Cohen’s Kappa. Since LLMs produce open-ended text responses, we first parse each output using regular expressions and map it to one of the predefined dataset labels. If a response cannot be reliably mapped to a valid label, we default the prediction to the majority class from the training set. For the DrugBank dataset, which contains 86 long, descriptive label phrases, we supplement regex-based evaluation with BERTScore to capture semantic similarity and provide a more robust assessment beyond exact string matching.

5 Results

5.1 K-Paths improves zero-shot reasoning

We evaluate LLMs’ reasoning capabilities for drug-disease interaction prediction in three zero-shot scenarios with varying context:

(1) **Base**—the LLM relies solely on its internal knowledge; (2) **Definitions**—textual definitions of drugs and diseases are provided; and (3) **K-Paths**—the LLM is provided with diverse paths retrieved by our K-Paths framework. Table 2 presents results across multiple LLM sizes and domains on the DDInter, PharmacotherapyDB, and DrugBank datasets.

Consistent improvements with K-Paths: Across all datasets, LLMs significantly benefit from K-Paths, outperforming both Base and Definitions settings. On DDInter, K-Paths boosts the F1-score by 13.42 for Llama 3.1 8B and 6.18 for Llama 3.1 70B, compared to the Definitions setting. Similar gains appear on PharmacotherapyDB with improvements of 12.45 (8B) and 8.46 (70B). The Tx-Gemma models (9B/27B), specifically fine-tuned for therapeutic tasks, show remarkable gains with K-Paths—achieving 29.2 and 19.8 point F1-score improvements on DDInter, respectively. In some cases, Definitions slightly improve performance over Base or even degrade it, but they consistently fall short of the substantial gains achieved with K-Paths. This gap likely stems from the nature of the information provided. K-Paths offers structured, contextualized knowledge, showing how entities are related. Definitions, while informative, provide declarative knowledge that lacks crucial relational context, needed for LLMs to reason about interactions effectively.

The harder the task, the larger the gap: DrugBank involves 86 interaction types. In the Base setting, most models hover around 1% F1 or 19% BERTScore, whereas with K-Paths they achieve 30–40%, emphasizing the importance of structured external knowledge in such complex tasks.

K-Paths eases need for scale & domain-specific models: Larger models generally outperform smaller ones in the Base setting (e.g., 70B vs. 8B Llama), but these gains diminish with external knowledge. Surprisingly, on DDInter, smaller models (8B Llama, Qwen 14B) occasionally outperform their larger counterparts when using K-Paths, suggesting high-quality external knowledge can reduce reliance on model scale. Likewise, Tx-Gemma’s domain-specific pre-training only helps consistently on PharmacotherapyDB. However, their performance varies across datasets, suggesting domain specialization doesn’t guarantee optimal reasoning. These results demonstrate that structured reasoning paths provide benefits for zero-shot interaction prediction, often outweighing the benefits of model scale or domain-specific pretraining.

5.2 K-Paths enables explainable inference

To complement our quantitative evaluation in section 5.1, we conducted a qualitative analysis examining how the different forms of external knowledge—definitions and K-Paths influence the Llama 3.1 70B’s responses. Table 3 presents two case studies from PharmacotherapyDB and DDInter, comparing the model’s predictions under different knowledge augmentation conditions. We make two key observations. (1) The Base model frequently predicts incorrectly, demonstrating insufficient domain knowledge. In the PharmacotherapyDB example, although the Definition of *Vincristine* mentions its use in cancer treatments, the model failed to infer its applicability to *Muscle Cancer*. However, incorporating reasoning paths from K-Paths corrects the LLM’s predictions. This suggests that factual definitions without explicit relational information are insufficient for correct reasoning in interaction prediction.

Table 2: Performance comparison of different models on zero-shot reasoning tasks. Bold indicates the best performance, while underlined denotes the second-best performance. K-Paths improves domain-specific reasoning in a zero-shot setting.

Model	Setting	DDInter			PharmacotherapyDB			DrugBank			
		Accuracy	F1	Kappa	Accuracy	F1	Kappa	Accuracy	F1	Kappa	BERTScore
Llama 3.1 8B Instruct	Base	69.09	33.34	4.37	57.94	<u>51.91</u>	35.18	31.36	0.65	0.87	18.29
	Defintions	<u>70.43</u>	<u>33.46</u>	<u>5.51</u>	57.54	51.70	34.38	<u>31.63</u>	<u>0.68</u>	<u>0.95</u>	12.09
	K-Paths	75.93	46.76	36.99	69.44	64.36	51.34	55.54	40.46	45.58	63.51
Llama 3.1 70B Instruct	Base	70.51	<u>40.01</u>	<u>19.07</u>	60.71	59.00	39.38	<u>31.32</u>	<u>1.35</u>	<u>3.88</u>	18.65
	Definitions	<u>71.61</u>	37.07	12.98	<u>62.30</u>	<u>61.14</u>	<u>42.30</u>	30.48	1.03	1.84	<u>19.69</u>
	K-Paths	78.01	46.19	35.33	71.03	67.46	54.66	57.72	46.91	49.19	66.52
Tx-Gemma-9B-chat	Base	<u>16.73</u>	<u>16.75</u>	<u>0.85</u>	<u>67.86</u>	<u>65.40</u>	<u>49.08</u>	<u>31.43</u>	<u>0.85</u>	<u>0.59</u>	12.42
	Defintions	8.54	8.48	0.56	64.29	61.46	45.10	31.04	0.80	<u>0.59</u>	<u>18.83</u>
	K-Paths	56.73	45.94	26.62	70.63	66.04	53.09	53.38	41.05	44.64	63.14
Tx-Gemma-27B-chat	Base	<u>27.52</u>	<u>27.28</u>	<u>4.83</u>	<u>67.46</u>	<u>67.49</u>	<u>51.39</u>	31.40	<u>0.79</u>	<u>1.73</u>	18.49
	Definitions	20.86	21.72	3.3	59.52	58.98	40.20	<u>31.47</u>	0.77	1.6	<u>19.21</u>
	K-Paths	60.88	47.06	26.56	72.62	71.49	58.19	50.71	43.93	40.33	58.78
Qwen2.5-14B-Instruct	Base	19.76	<u>18.36</u>	<u>0.78</u>	58.33	<u>57.91</u>	<u>39.72</u>	31.53	<u>0.61</u>	<u>0.07</u>	5.00
	Definitions	<u>22.16</u>	18.23	0.43	57.94	57.3	39.13	<u>31.54</u>	<u>0.61</u>	<u>0.07</u>	-1.00
	K-Paths	66.25	49.16	30.03	65.08	58.18	45.75	49.12	41.76	33.84	47.39
Qwen2.5-32B-Instruct	Base	<u>38.23</u>	<u>31.13</u>	<u>2.6</u>	<u>65.08</u>	<u>65.03</u>	<u>47.50</u>	<u>31.95</u>	<u>0.78</u>	<u>1.85</u>	12.98
	Definitions	23.61	20.98	0.81	62.30	62.55	44.14	31.54	0.61	0.04	1.00
	K-Paths	63.69	48.88	29.23	71.83	67.83	55.10	41.23	31.31	20.02	32.69

Table 3: Comparison of LLM responses based on external knowledge type. K-Paths allows for explainable inference.

	PharmacotherapyDB	DDinter
Query	Determine the possible effect of using Vincristine (Drug) for Muscle cancer (Disease).	Determine the severity of interaction when Ritonavir (Drug 1) and Leflunomide (Drug 2) are taken together.
Answer	Disease Modifying	Major
Definitions	Vincristine is an antitumor that treats leukemia, lymphoma, Hodgkin's disease, and other blood disorders. Muscle cancer is a musculoskeletal system cancer located in the muscle.	Ritonavir , an HIV protease inhibitor, boosts other protease inhibitors' effectiveness and is used in some HCV therapies. As a CYP3A inhibitor, it increases drug concentrations. Leflunomide is a pyrimidine synthesis inhibitor belonging to the disease-modifying antirheumatic drugs chemically and pharmacologically very heterogeneous.
K-Paths	Vincristine treats Kidney Cancer (Disease) and Kidney Cancer (Disease) resembles Muscle Cancer . Vincristine downregulates TP53 (Gene) and TP53 (Gene) is associated with Muscle Cancer.	Ritonavir binds CYP2C9 (Gene) and CYP2C9 (Gene) is bound by Leflunomide Ritonavir causes Neutropenia (Side Effect), and Neutropenia (Side Effect) is caused by Leflunomide
LLM Only	Non Indications	Moderate
LLM+Definitions	Non Indications	Moderate
LLM+K-Paths	Disease Modifying	Major

(2) K-Paths enables the LLM to connect relevant entities and their contextual relationships, leading to improved accuracy. For instance, in the DDInter example, K-Paths allows the model to correctly predict a “Major” interaction between **Ritonavir** and **Leflunomide**, highlighting **Neutropenia** as a possible side effect. This is crucial, as **Neutropenia** is a potentially life-threatening condition. A “Moderate” misclassification could have severe safety implications, stressing the importance of high-quality reasoning paths in safety-critical applications. In summary, while Definitions provide some information about query entities, structured, diverse KG paths are essential for effective reasoning and improved zero-shot performance. We further examine edge cases with limited or no KG paths in appendix B, demonstrating K-Paths’s graceful fallback to parametric knowledge when structural information is unavailable.

5.3 Path selection strategies and performance

We study how number of retrieved paths (K) and path selection strategy affect the prediction performance of Llama 3.1 8B on the validation set. We evaluate several path selection strategies: (1) the “Base” setting ($K = 0$), (2) diverse paths from K-Paths ($K = 1, 5, 10, 15, 20$). As K increases, the number of hops and path length increases. (3) shortest paths (without diversity filtering), and (4) local neighborhood edges (5 neighbors per entity). Results in fig. 3 show that adding diverse reasoning paths from K-Paths substantially improve performance. On DDInter, F1-scores increase from 12.66% ($K = 0$) to 46.73% ($K = 10$), and on DrugBank, they increase from 0.59% ($K = 0$) to 43.35% ($K = 10$). However, performance gains diminish beyond $K = 10$, suggesting that excessively

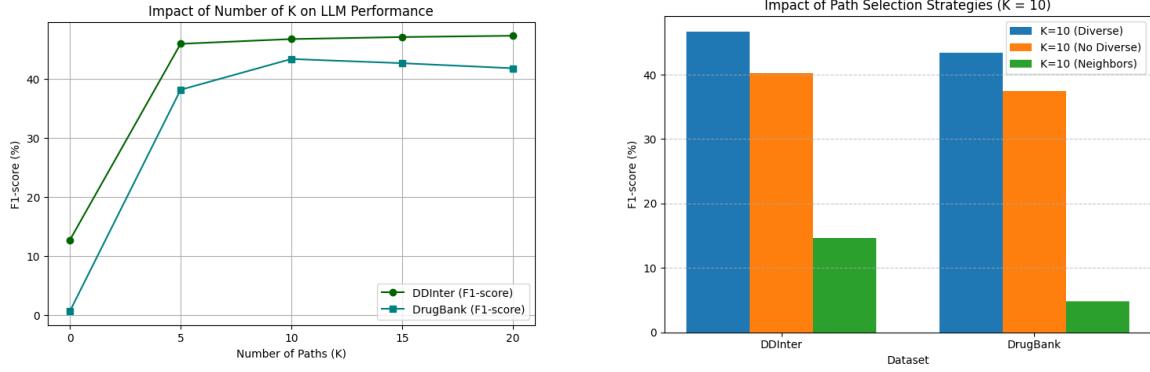


Figure 3: Influence of path selection strategies on Llama 3.1 8B. Diverse paths are essential for performance improvement.

Table 4: Supervised performance of various models across datasets. Bold indicates the best result within each model category; * marks the overall best performance. With supervision, LLMs benefit from both knowledge types, while K-Paths enhances GNN efficiency without significant performance loss.

Model	Setting	DDInter			PharmacotherapyDB			DrugBank		
		Accuracy	F1	Kappa	Accuracy	F1	Kappa	Accuracy	F1	Kappa
LLM-Based Models										
QLoRA-Llama	Definitions	82.36	67.96	53.64	81.75*	79.91*	71.84*	73.22*	68.15*	67.26*
QLoRA-Llama	K-Paths	80.55	68.63	57.85	78.57	76.71	66.68	71.83	65.57	65.63
Graph-Based Models (GNNs)										
EmerGNN	Complete KG	84.26	68.00	58.92*	71.43	68.41	55.38	71.04	59.42	65.14
EmerGNN	K-Paths	84.53*	68.85*	58.91	71.03	68.12	54.10	68.98	59.06	62.54
RGCN	Complete KG	72.01	51.47	31.38	61.11	60.08	41.12	29.98	15.49	20.88
RGCN	K-Paths	73.32	52.12	32.70	66.82	61.74	39.25	31.70	17.51	23.14

long paths introduce redundancy or noise. K-Paths consistently outperforms shortest-path selection (without diversity filtering) and neighborhood selection. Omitting diversity filtering results in degradation of the F1-score by 6.99% on DDInter and 4.32% on DrugBank, highlighting the importance of diverse path retrieval.

5.4 Impact of K-Paths in supervised settings

We evaluate the influence of subgraphs derived from retrieved paths on supervised learning for LLMs and GNNs.

LLM performance: We fine-tuned Llama 3.1 8B Instruct using K-Paths and textual definitions. As shown in table 5, fine-tuned LLMs perform comparably regardless of the training source, indicating their ability to integrate both structured and unstructured knowledge when supervision is provided. However, we further observe that definitions often outperform K-Paths, likely because they provide a more direct and semantically rich signal that is easier to learn; in contrast, in the zero-shot setting (section 5.1), K-Paths provide more useful relational cues.

GNN performance: We compared GNN models trained on the full knowledge graph (Complete KG) against those trained on the subgraphs constructed from the K-Paths framework. Table 4 shows that with K-Paths, we achieve comparable accuracy to using the Complete KG despite being approximately 90% smaller. This reinforces that smaller, task-specific graphs enhance efficiency without significant performance loss. For example, on DDInter, EmerGNN

achieves nearly identical performance using K-Paths (F1: 68.85%) compared to Complete KG (F1: 68.00%), suggesting that a targeted subgraph retains essential knowledge while significantly improving efficiency. RGCN benefits from K-Paths, with F1 increasing from 51.47% (Complete KG) to 52.12% (K-Paths) on DDInter and from 15.49% to 17.51% on DrugBank. This highlights the advantage of a more focused graph structure for such models. This result is consistent in the transductive setting shown in appendix F.2.

Efficiency analysis: Table 5 presents a comparison of graph statistics and training times before and after applying K-Paths for subgraph retrieval, along with the retrieval duration in minutes. The results clearly show that K-Paths substantially reduces the number of nodes, relations, and triplets, leading to significantly faster training and lower memory overhead. For instance, on DrugBank, the number of nodes reduces from 35,103 to 6,335, and triplets from 1,789,976 to 184,273. Correspondingly, EmerGNN’s training time per epoch drops from 1011.86s to 146.98s.. A similar trend is observed in DDInter, where subgraph pruning leads to a speedup from 606.24s to 102.21s per epoch. Note that K-Paths includes reverse edges during retrieval, effectively doubling the triplet count. Despite this, the total retrieval time remains under 15 minutes for all datasets. We see a similar trend in table 8 for the test data and show the important relations retained by K-Paths in appendix F.4. These results highlight that K-Paths improves training efficiency without compromising model performance, reinforcing the scalability of our approach.

Table 5: Augmented KG Statistics: K-Paths Improves Training Efficiency.

Dataset	Before Retrieval (Complete KG)					After Retrieval (K-Paths)					
	#Nodes	#Relations	#Triplets	Sec/Epoch	{EmerGNN, RGCN}	#Nodes	#Relations	#Triplets	Sec/Epoch	{EmerGNN, RGCN}	Time (min)
DDInter	35,107	26	1,763,596	{606.24s, 5.78s}		4,723	18	113,933	{102.21s, 1.18s}		13.57
PharmacotherapyDB	34,412	26	1,691,829	{20.83s, 2.41s}		3,307	23	11,905	{2.95s, 0.12s}		10.17
DrugBank	35,103	109	1,789,976	{1011.86s, 11.41s}		6,335	101	184,273	{146.98s, 3.56s}		13.12

LLM vs. GNN performance: Finally, we observe that supervised fine-tuning enables LLMs to outperform GNNs on most datasets, suggesting that supervised LLMs can effectively leverage multiple structured and unstructured modalities and sometimes even surpass GNNs trained solely on relational graphs.

6 Conclusion

We present *K-Paths*, a model-agnostic retrieval framework that introduces a diversity-aware adaptation of Yen’s algorithm to extract biologically meaningful, multi-hop reasoning paths from large biomedical KGs. These paths serve as structured, relational evidence that supports both prediction and interpretability across model types. Our experiments show that K-Paths benefits both LLMs and GNNs. For LLMs, the retrieved paths serve as useful context that enables zero-shot, inductive reasoning about unobserved drug–disease interactions. For GNNs, K-Paths identifies compact subgraphs, reducing graph size by up to 90% while maintaining performance and improving training efficiency. While our focus is on repurposing and interaction prediction, K-Paths is potentially generalizable to other biomedical tasks such as protein–protein interaction prediction or treatment recommendation. We acknowledge limitations: K-Paths relies on existing interaction types and graph connectivity, which may limit its ability to infer entirely novel interaction types or handle sparsely represented entities. Nevertheless, by anchoring predictions in relationally diverse paths, K-Paths mitigates hallucinations and supports biologically grounded inference. To our knowledge, K-Paths is one of the first frameworks to enable path-based heuristics for reasoning over unseen drug pairs, a critical need for early-stage drug discovery. We believe this work lays a foundation for future exploration of integrating LLMs and KGs in biomedical applications, paving the way for more efficient and interpretable solutions in drug discovery and other related fields.

Acknowledgements

This material is based in part upon work supported by the National Science Foundation under Grant No. RISE-2425380. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. Disclosure: Stephen Bach is an advisor to Snorkel AI, a company that provides software and services for data-centric artificial intelligence.

References

- [1] Hussein Abdallah, Waleed Afandi, Panos Kalnis, and Essam Mansour. 2024. Task-Oriented GNNs Training on Large Knowledge Graphs for Accurate and Efficient Modeling. *arXiv preprint arXiv:2403.05752* (2024).
- [2] Tassallah Abdullahi, Laura Mercurio, Ritambhara Singh, and Carsten Eickhoff. 2024. Retrieval-Based Diagnostic Decision Support: Mixed Methods Study. *JMIR Medical Informatics* 12 (2024), e50209.
- [3] Tassallah Abdullahi, Ritambhara Singh, and Carsten Eickhoff. 2024. Retrieval augmented zero-shot text classification. In *Proceedings of the 2024 ACM SIGIR international conference on theory of information retrieval*. 195–203.
- [4] Tassallah Abdullahi, Ritambhara Singh, Carsten Eickhoff, et al. 2024. Learning to make rare and complex diagnoses with generative AI assistance: qualitative study of popular large language models. *JMIR Medical Education* 10, 1 (2024), e51391.
- [5] Reid Andersen, Fan Chung, and Kevin Lang. 2006. Local graph partitioning using pagerank vectors. In *2006 47th annual IEEE symposium on foundations of computer science (FOCS'06)*. IEEE, 475–486.
- [6] Albert-Laszlo Barabasi and Zoltan N Oltvai. 2004. Network biology: understanding the cell’s functional organization. *Nature reviews genetics* 5, 2 (2004), 101–113.
- [7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.
- [8] Juan Manuel Zambrano Chaves, Eric Wang, Tao Tu, Eeshit Dhaval Vaishnav, Byron Lee, S. Sara Mahdavi, Christopher Semturs, David Fleet, Vivek Natarajan, and Shekoofeh Azizi. 2024. Tx-LLM: A Large Language Model for Therapeutics. *arXiv:2406.06316 [cs.CL]* <https://arxiv.org/abs/2406.06316>
- [9] Susan B. Davidson, Chris Overton, and Peter Buneman. 1995. Challenges in integrating biological data sources. *Journal of Computational Biology* 2, 4 (1995), 557–572.
- [10] Edsger W Dijkstra. 1959. A note on two problems in connexion with graphs. *Numerische mathematik* 1, 1 (1959), 269–271.
- [11] Joseph A DiMasi, Ronald W Hansen, and Henry G Grabowski. 2003. The price of innovation: new estimates of drug development costs. *Journal of health economics* 22, 2 (2003), 151–185.
- [12] Haotong Du, Quanming Yao, Juzheng Zhang, Yang Liu, and Zhen Wang. 2024. Customized Subgraph Selection and Encoding for Drug-drug Interaction Prediction. *arXiv preprint arXiv:2411.01535* (2024).
- [13] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783* (2024).
- [14] Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, and Jonathan Larson. 2024. From Local to Global: A Graph RAG Approach to Query-Focused Summarization. *arXiv:2404.16130 [cs.CL]* <https://arxiv.org/abs/2404.16130>
- [15] Bahare Fatemi, Jonathan Halcrow, and Bryan Perozzi. 2023. Talk like a graph: Encoding graphs for large language models. *arXiv preprint arXiv:2310.04560* (2023).
- [16] Ziqi Gao, Chenran Jiang, Jiawen Zhang, Xiaosen Jiang, Lanqing Li, Peilin Zhao, Huanming Yang, Yong Huang, and Jia Li. 2023. Hierarchical graph learning for protein–protein interaction. *Nature Communications* 14, 1 (2023), 1093.
- [17] Vladimir Gligorjević and Nataša Pržulj. 2015. Methods for biological data integration: perspectives and challenges. *Journal of the Royal Society Interface* 12, 112 (2015), 20150571.
- [18] Will Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive representation learning on large graphs. *Advances in neural information processing systems* 30 (2017).
- [19] Daniel Himmelstein, Pouya Khankhanian, Christine S Hessler, Ari J Green, and Sergio Baranzini. 2016. PharmacotherapyDB 1.0: the open catalog of drug therapies for disease. *Figshare. DOI: https://doi.org/10.6084/m9.figshare.3103054* (2016).
- [20] Daniel Scott Himmelstein, Antoine Lizée, Christine Hessler, Leo Brueggeman, Sabrina L Chen, Dexter Hadley, Ari Green, Pouya Khankhanian, and Sergio E Baranzini. 2017. Systematic integration of biomedical knowledge prioritizes drugs for repurposing. *eLife* 6 (sep 2017), e26726. doi:10.7554/eLife.26726
- [21] Kexin Huang, Payal Chandak, Qianwen Wang, Shreyas Havaldar, Akhil Vaid, Jure Leskovec, Girish N Nadkarni, Benjamin S Glicksberg, Nils Gehlenborg, and Marinka Zitnik. 2024. A foundation model for clinician-centered drug repurposing. *Nature Medicine* (2024), 1–13.
- [22] Yoshitaka Inoue, Tianci Song, and Tianfan Fu. 2024. DrugAgent: Explainable Drug Repurposing Agent with Large Language Model-based Reasoning. *arXiv:2408.13378 [cs.AI]* <https://arxiv.org/abs/2408.13378>

- [23] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems* 35 (2022), 22199–22213.
- [24] Michael Kuhn, Christian von Mering, Monica Campillos, Lars Juhl Jensen, and Peer Bork. 2007. STITCH: interaction networks of chemicals and proteins. *Nucleic acids research* 36, suppl_1 (2007), D684–D688.
- [25] Youwei Liang, Ruiyi Zhang, Li Zhang, and Pengtao Xie. 2023. DrugChat: Towards Enabling ChatGPT-Like Capabilities on Drug Molecule Graphs. arXiv:2309.03907 [q-bio.BM] <https://arxiv.org/abs/2309.03907>
- [26] David Liben-Nowell and Jon Kleinberg. 2003. The link prediction problem for social networks. In *Proceedings of the twelfth international conference on Information and knowledge management*. 556–559.
- [27] Xuan Lin, Zhe Quan, Zhi-Jie Wang, Tengfei Ma, and Xiangxiang Zeng. 2020. KGNN: Knowledge Graph Neural Network for Drug-Drug Interaction Prediction. In *IJCAI*, Vol. 380. 2739–2745.
- [28] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv:1907.11692 [cs.CL] <https://arxiv.org/abs/1907.11692>
- [29] Linhao Luo, Yuan-Fang Li, Gholamreza Haffari, and Shirui Pan. 2024. Reasoning on Graphs: Faithful and Interpretable Large Language Model Reasoning. arXiv:2310.01061 [cs.CL] <https://arxiv.org/abs/2310.01061>
- [30] Costas Mavromatis and George Karypis. 2024. GNN-RAG: Graph Neural Retrieval for Large Language Model Reasoning. arXiv:2405.20139 [cs.CL] <https://arxiv.org/abs/2405.20139>
- [31] Steven M Paul, Daniel S Mytelka, Christopher T Dunwiddie, Charles C Persinger, Bernard H Munos, Stacy R Lindborg, and Aaron L Schacht. 2010. How to improve R&D productivity: the pharmaceutical industry’s grand challenge. *Nature reviews Drug discovery* 9, 3 (2010), 203–214.
- [32] Ciyan Peng, Feng Xia, Mehdi Nasiriparsa, and Francesco Osborne. 2023. Knowledge graphs: Opportunities and challenges. *Artificial Intelligence Review* 56, 11 (2023), 13071–13102.
- [33] Bryan Perozzi, Bahare Fatemi, Dustin Zelle, Anton Tsitsulin, Mehran Kazemi, Rami Al-Rfou, and Jonathan Halcrow. 2024. Let your graph do the talking: Encoding structured data for llms. *arXiv preprint arXiv:2402.05862* (2024).
- [34] M. Schlichtkrull, Thomas Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2017. Modeling Relational Data with Graph Convolutional Networks. In *Extended Semantic Web Conference*. Springer, New York, NY, USA, 593–607. <https://api.semanticscholar.org/CorpusID:5458500>
- [35] Dong Shu, Tianle Chen, Mingyu Jin, Chong Zhang, Mengnan Du, and Yongfeng Zhang. 2024. Knowledge Graph Large Language Model (KG-LLM) for Link Prediction. *Proceedings of Machine Learning Research* 260, 62 (2024).
- [36] Jessica Vamathevan, Dominic Clark, Paul Czodrowski, Ian Dunham, Edgardo Ferran, George Lee, Bin Li, Anant Madabhushi, Parantu Shah, Michaela Spitzer, et al. 2019. Applications of machine learning in drug discovery and development. *Nature reviews Drug discovery* 18, 6 (2019), 463–477.
- [37] Eric Wang, Samuel Schmidgall, Paul F Jaeger, Fan Zhang, Rory Pilgrim, Yossi Matias, Joelle Barral, David Fleet, and Shekoofeh Azizi. 2025. TxGemma: Efficient and Agentic LLMs for Therapeutics. *arXiv preprint arXiv:2504.06196* (2025).
- [38] Yaqing Wang, Zaifei Yang, and Quanming Yao. 2024. Accurate and interpretable drug-drug interaction prediction enabled by knowledge subgraph learning. *Communications Medicine* 4, 1 (2024), 59.
- [39] Michelle Whirl-Carrillo, Rachel Huddart, Li Gong, Katrin Sangkuhl, Caroline F. Thorn, Ryan Whaley, and Teri E. Klein. 2021. An Evidence-Based Framework for Evaluating Pharmacogenomics Knowledge for Personalized Medicine. *Clinical Pharmacology & Therapeutics* 110, 3 (July 2021), 563–572. doi:10.1002/cpt.2350
- [40] David S Wishart, Yannick D Feunang, An C Guo, Elvis J Lo, Anna Marcu, Jason R Grant, Tanvir Sajed, Daniel Johnson, Carin Li, Zinat Sayeeda, et al. 2018. DrugBank 5.0: a major update to the DrugBank database for 2018. *Nucleic acids research* 46, D1 (2018), D1074–D1082.
- [41] Junde Wu, Jiayuan Zhu, Yunli Qi, Jingkun Chen, Min Xu, Filippo Menolascina, and Vicente Grau. 2024. Medical graph rag: Towards safe medical large language model via graph retrieval-augmented generation. *arXiv preprint arXiv:2408.04187* (2024).
- [42] Guoli Xiong, Zhijiang Yang, Jiacai Yi, Ningning Wang, Lei Wang, Huimin Zhu, Chengkun Wu, Aiping Lu, Xiang Chen, Shao Liu, et al. 2022. DDInter: an online drug-drug interaction database towards improving clinical decision-making and patient safety. *Nucleic acids research* 50, D1 (2022), D1200–D1207.
- [43] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengan Huang, Chenxu Lv, et al. 2025. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388* (2025).
- [44] Jin Y Yen. 1971. Finding the k shortest loopless paths in a network. *management Science* 17, 11 (1971), 712–716.
- [45] Yue Yu, Kexin Huang, Chao Zhang, Lucas M Glass, Jimeng Sun, and Cao Xiao. 2021. SumGNN: multi-typed drug interaction prediction via efficient knowledge graph summarization. *Bioinformatics* 37, 18 (2021), 2988–2995.
- [46] Yongqi Zhang, Quanming Yao, Ling Yue, Xian Wu, Ziheng Zhang, Zhenxi Lin, and Yefeng Zheng. 2023. Emerging Drug Interaction Prediction Enabled by Flow-based Graph Neural Network with Biomedical Network. arXiv:2311.09261 [q-bio.QM] <https://arxiv.org/abs/2311.09261>
- [47] Marinka Zitnik, Monica Agrawal, and Jure Leskovec. 2018. Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics* 34, 13 (2018), i457–i466.

A LLM prompting

We use the following prompt templates across datasets. As shown below, we provide predefined answer options for the DDInter and PharmacotherapyDB datasets. However, we do not include options for the DrugBank dataset for two reasons: (1) DrugBank contains 86 possible interaction types, making inference for approximately 30,000 examples computationally expensive. and (2) preliminary experiments showed that the model performed better without predefined options. As baselines, we either exclude knowledge graph information entirely or provide textual definitions of the drugs or diseases. For Tx-Gemma, we followed the original authors’ prompt instruction format [37].

Prompt template

Unified prompt template

System Prompt:

You are a pharmacodynamics expert. Answer the question using the given knowledge graph information (if available), essential drug definitions, and your medical expertise. Base your answer on evidence of known interaction mechanisms, pharmacological effects, or similarities to related compounds, if applicable. Avoid generalizations unless directly supported.

Your answer must be concise and formatted as follows:
##Answer:<Dataset Specific>

Dataset-Specific Instructions:

- **DDInter:** Assess interaction severity & format as: **Answer:** <Major / Moderate / Minor>
- **PharmacotherapyDB:** Determine therapeutic indication & format as: **Answer:** <Disease modifying / Palliates / Nonindication>
- **DrugBank:** Identify mechanism/effect type & format as: **Answer:** <DrugX mechanism/effect on DrugY>

Question:

Determine the interaction type or therapeutic indication when (EntityX) and (EntityY) are used together.

Knowledge Graph Information:

<Knowledge graph data>

B Qualitative analysis for edge cases

We inspect two representative queries that highlight the strengths and limitations of K-Paths.

Partial KG coverage.

Query: The interaction severity for **Mipomersen & Oxymetholone?**

(Label: MAJOR)

Paths: Mipomersen → MAJOR → (Regorafenib, Diclofenac)

Note: No retrieved evidence for Oxymetholone.

Predictions: **Base:** MINOR, **+Def:** MAJOR, **+K-Paths:** MODERATE

This case shows that while partial KG coverage offers helpful cues (e.g., Mipomersen’s links to severe interactions), models may underutilize these cues without additional context.

No KG coverage.

Query: An example where no paths exist for either entity.

In such cases, K-Paths defaults to the Base LLM’s parametric knowledge. We observe mixed results: Base sometimes predicts correctly, and definitions offer occasional gains. This exposes a natural ceiling for any retrieval-based approach when the external source is unavailable.

C Textual definitions with K-Paths

We explored using definitions with K-Paths in zero-shot settings. It leads to longer input contexts and does not consistently improve performance. For instance, on PharmacotherapyDB with Llama-3.1-70B, the F1 score slightly decreases from 67.46 (K-Paths) to 66.00 (K-Paths + Descriptions); on DDInter, performance remains unchanged (46.19 to 46.18). The added context also increases inference time and cost, making fine-tuning expensive. Based on these findings, we focus on path-only retrieval for scalable and cost-effective use.

D QLORA fine-tuning

For the supervised LLM experiments, we fine-tuned Llama 3.1 8B Instruct using QLoRA. We conducted training experiments under two distinct scenarios across our datasets. In the first scenario, we trained the model using the retrieved paths for each training query. In the second scenario, we trained the model using text definitions of the drugs or diseases. Definitions were capped at 200 tokens, reflecting the dataset’s average definition length. In both scenarios, we trained for 10 epochs using the default settings of the QLoRA repository, with the following modifications: A learning rate of 1e-3 and the maximum input length to the average token length of the input across the respective dataset. The training was conducted on 8 A100-80G GPUs and typically completed within 24 hours, depending on the dataset. During inference, we first retrieved reasoning paths using K-Paths. These retrieved paths were then appended to the original query and fed into the fine-tuned LLM to generate the final answers.

E GNN baselines

E.1 Relational graph convolutional network

We implement the Relational Graph Convolutional Network (RGCN) [34], which operates on the augmented graph with multiple relation types and employs message passing to propagate structured information across nodes. The model is implemented using PyTorch and PyTorch Geometric.

Node feature initialization: Drug and disease nodes are initialized using RoBERTa [28] embeddings extracted from PubMed-scraped descriptions. Other entity nodes (genes, anatomy, etc.) are initialized randomly, allowing the model to learn meaningful representations during training.

Training setup: We follow the inductive setting for dataset splitting, as described in section 4.1. The training follows a link prediction framework where training nodes are sampled with all their relations observed, while test nodes are introduced to evaluate generalization. We consider two training settings: (1) Training on the entire augmented KG (*Complete KG*) and testing on test nodes along with their retrieved test KG, and (2) Training on the diverse retrieved train paths and testing on test nodes along with their retrieved test paths.

Model architecture & training details: We train a three-layer RGCN using the Adam optimizer with a learning rate of 1e-3 and a scheduler based on validation loss. Cross-entropy loss is used to predict drug-drug or drug-disease interactions. Training runs for up to 1,000 epochs with early stopping. To ensure class balance, stratified sampling selects up to 1,000 samples per epoch and 10 per class. Each GCN layer is followed by ReLU, projecting node embeddings into a 128-dimensional space with batch normalization, ReLU, and dropout (rate 0.5). An edge classifier predicts interaction types using the final embeddings of entity pairs. To improve efficiency, we use basis decomposition with two bases.

E.2 EmerGNN

To compare against RGCN, we evaluate EmerGNN, a graph neural network designed for emerging drug-drug interaction prediction [46]. We use the official implementation and apply it to our datasets without modifying the model architecture or training pipeline. Unlike RGCN, which relies on RoBERTa embeddings for node initialization, EmerGNN incorporates molecular features, leveraging structural and chemical properties to enhance node representation. We compare the performance of both models in terms of interaction prediction accuracy, assessing the impact of different node initialization strategies and augmented KG utilization.

F Additional experimental results

This section presents additional experimental results, including retrieved paths from the augmented KG, dataset statistics, path retrieval efficiency, and model performance comparisons.

F.1 Dataset overview

Table 6 summarizes the datasets used in our experiments, categorized by prediction task and the connectivity between interaction query nodes (entities) in the augmented KG.

- DrugBank involves inductive and transductive tasks, predicting drug-drug interactions among 86 labels. The transductive setting has more drug pairs connected in the augmented KG (38,411) than the inductive setting (27,983).
- DDInter predicts drug-drug interaction severity levels (Major, Moderate, or Minor). It contains 13,841 connecting drug pairs, and 5,494 interaction queries contain information about a single entity.
- PharmacotherapyDB focuses on whether a drug is disease-modifying, palliates, or has no indication of a disease.

Table 6: Summary of datasets and tasks.

Dataset	Task	Two Nodes	Single Nodes	No Node
DrugBank (Ind.)	Open-ended	27,983	3,987	14
DrugBank (Trans.)	Open-ended	38,411	8	0
DDInter	Categorical	13,841	5,494	104
PharmacotherapyDB	Categorical	252	0	0

Table 7: Performance of models on the Transductive DrugBank setting. Bold indicates the best performance, and underlined denotes the second-best. LLMs leverage both knowledge types effectively with supervision, and K-Paths enhance GNN efficiency without significant performance loss.

Model	Setting	Accuracy	F1	Kappa
Graph-Based Models				
EmerGNN	Complete KG	97.40	94.00	96.60
EmerGNN	K-Paths	<u>97.01</u>	94.25	97.01
RGCN	Complete KG	90.01	87.62	89.85
RGCN	K-Paths	90.86	88.43	90.11
SumGNN	Reported	86.85	92.66	92.66
KnowDDI	Reported	91.53	93.17	91.89
Decagon	Reported	87.20	57.40	86.10
LLM-Based Models				
QLoRA-Llama	Definitions	93.45	91.71	92.28
QLoRA-Llama	K-Paths	93.58	88.98	92.41

Table 8: Statistics: Comparison of the Augmented KG with extracted subgraph at test time. Retrieval time reflects the cost of extracting K-Paths for all test queries.

Dataset	Before Retrieval (Augmented KG)			After Retrieval (K-Paths)			
	#Nodes	#Relations	#Triplets	#Nodes	#Relations	#Triplets	Time (min)
DrugBank	35,146	102	1,722,677	6,378	94	175,698	13.00
DDInter	35,169	26	1,710,079	5,647	18	102,854	18.01
PharmDB	33,952	26	1,690,945	2,847	23	13,942	10.15

F.2 Transductive results

Table 7 compares the performance of GNNs and LLM-based models on the Drugbank transductive dataset. Among GNNs, EmerGNN performs best, achieving 97.40% accuracy with the complete KG, while using K-Paths slightly lowers accuracy (97.01%) but improves the F1-score. RGCN performs worse than EmerGNN but benefits from our KG, increasing accuracy from 90.01% to 90.86%. For LLM-based models, QLoRA-Llama achieves 93.58% accuracy when using K-Paths, while using text-based descriptions instead of structured knowledge results in similar accuracy (93.45%) but a slightly higher F1-score. Overall, EmerGNN performs best and K-Paths improves efficiency without significant performance loss.

F.3 Augmented KG vs. Test-Time subgraph

Table 8 compares the augmented KG’s overall structure to the subgraph extracted at test time. We report the number of nodes, relations, and triplets before and after query-specific retrieval. Filtering for relevant subgraphs significantly reduces graph size: DrugBank shrinks from 1.7M to 175K triplets, DDInter from 1.71M to 102K, and PharmacotherapyDB from 1.69M to 13K. This demonstrates the efficiency of query-specific retrieval in extracting only the most relevant paths for inference.

Table 9: Hetionet relations retained by K-Paths. Relation names use: A = Anatomy, D = Disease, G = Gene, C = Compound, SE = Side Effect, PC = Pharmacologic Class, BP = Biological Process, CC = Cellular Component, MF = Molecular Function, PW = Pathway. ✓ = present, ✗ = absent.

ID	Relation	Drugbank	DDinter	PharmDB
0	A-downregulates-G	✗	✗	✓
1	A-expresses-G	✗	✗	✓
2	A-upregulates-G	✗	✗	✓
3	C-binds-G	✓	✓	✓
4	C-causes-SE	✓	✓	✓
5	C-downregulates-G	✓	✓	✓
6	C-palliates-D	✓	✓	✓
7	C-resembles-C	✓	✓	✓
8	C-treats-D	✓	✓	✓
9	C-upregulates-G	✓	✓	✓
10	D-associates-G	✓	✓	✓
11	D-downregulates-G	✓	✓	✓
12	D-localizes-A	✗	✗	✓
13	D-presents-SE	✗	✗	✓
14	D-resembles-D	✓	✓	✓
15	D-upregulates-G	✓	✓	✓
16	G-covaries-G	✓	✓	✓
17	G-interacts-G	✓	✓	✓
18	G-participates-BP	✗	✗	✗
19	G-participates-CC	✗	✗	✗
20	G-participates-MF	✗	✗	✗
21	G-participates-PW	✗	✗	✗
22	G-regulates-G	✓	✓	✓
23	PC-includes-C	✓	✓	✓

F.4 Retained Hetionet relations

Table 9 presents the Hetionet relations retained in the training and test subgraphs after K-Paths was applied to each dataset. These relations encapsulate biological interactions, such as gene participation in molecular functions or compounds treating diseases.

Since Hetionet serves as a structured biomedical KG, these relations may not be inherent to the datasets (DrugBank, DDInter, PharmacotherapyDB) themselves but instead provide additional contextual knowledge that enhances reasoning within the models. The path retrieval process selectively retains the most informative relations while discarding those less relevant to each dataset.

For example, in the DDInter dataset:

- Hetionet originally contained 23 distinct relation types, covering diverse biological interactions.
- After dataset-specific path extraction, only 15 relation types were retained by K-Paths and used for training, ensuring that only the most relevant interactions contributed to the model.

Interestingly, while PharmacotherapyDB is the smallest dataset, it retains more relations than DrugBank and DDInter. This is because PharmacotherapyDB encompasses both drug and disease entities, covering a broader set of biomedical interactions that span multiple entities.