Brittleness and Promise: Knowledge Graph–Based Reward Modeling for Diagnostic Reasoning

Saksham Khatwani[1, 2], He Cheng[1], Majid Afshar[3], Dmitriy Dligach[4], Yanjun Gao[1]
[1]University of Colorado Anschutz Medical Campus
[2]University of Colorado Boulder
[3]University of Wisconsin Madison
[4]Loyola University
Correspondence: yanjun.gao@cuanschutz.edu

Abstract

Large language models (LLMs) show promise for diagnostic reasoning but often lack reliable, knowledge-grounded inference. Knowledge graphs (KGs), such as the Unified Medical Language System (UMLS), offer structured biomedical knowledge that can support trustworthy reasoning. Prior approaches typically integrate KGs via retrieval-augmented generation or fine-tuning, inserting KG content into prompts rather than enabling structured reasoning. We explore an alternative paradigm: treating the LLM as a *reward* model of KG reasoning paths, where the model learns to *judge* whether a candidate path leads to correct diagnosis for a given patient input. This approach is inspired by recent work that leverages reward training to enhance model reasoning abilities, and grounded in computational theory, which suggests that *verifying* a solution is often easier than generating one from scratch. It also parallels physicians' diagnostic assessment, where they *judge* which sequences of findings and intermediate conditions most plausibly support a diagnosis. We first systematically evaluate five task formulation for knowledge path judging and eight training paradigm. Second, we test whether the path judging abilities generalize to downstream diagnostic tasks, including diagnosis summarization and medical question answering. Experiments with three open-source instruct-tuned LLMs reveal both promise and brittleness: while specific reward optimization and distillation lead to strong path-judging performance, the transferability to downstream tasks remain weak. Our finding provides the first systematic assessment of "reward model style" reasoning over clinical KGs, offering insights into how structured, reward-based supervision influences diagnostic reasoning in GenAI systems for healthcare. Our implementation can be found at `https://github.com/LARK-NLP-Lab/kg-rl-reasoner`.

1 Introduction

Large language models (LLMs) have shown remarkable potential in supporting diagnostic reasoning, enabling them to generate differential diagnoses, summarize clinical information, and answer complex medical questions Goh et al. [2024], Wang and Liu [2025], Liu et al. [2025]. Yet, despite their fluency, LLMs frequently fall short in terms of faithful reasoning: they may overlook relevant evidence, hallucinate unsupported conclusions, or rely on superficial statistical associations rather than structured medical knowledge Kim et al. [2025], Tonmoy et al. [2024], Asgari et al. [2025]. For deployment in clinical practice, these limitations highlight the urgent need to augment LLMs with knowledge-grounded reasoning mechanisms.

39th Conference on Neural Information Processing Systems (NeurIPS 2025) Workshop: The Second Workshop on GenAI for Health: Potential, Trust, and Policy Compliance.

Knowledge graphs (KGs) offer a promising pathway toward trustworthy diagnostic reasoning. Richly structured resources such as the Unified Medical Language System (UMLS) Bodenreider [2004] encode decades of curated biomedical concepts and relations, serving as a foundation for aligning free-text patient descriptions with symbolic diagnostic pathways. Recent efforts demonstrate the potential of combining KGs with LLMs to improve diagnostic generation, such as DR.KNOWS Gao et al. [2025a], KG4DIAGNOSIS Zuo et al. [2025], and other approaches like prompting with graph-of-thoughts Bedi et al. [2025], Wen et al. [2024]. These studies suggest that neural–symbolic integration can increase both the accuracy and interpretability of diagnostic reasoning.

Two primary strategies have emerged for incorporating KGs into LLMs. The first is graph-based retrieval-augmented generation (RAG), where an external retriever identifies relevant KG subgraphs or paths and inserts them into the LLM's prompt as additional context, as exemplified in Gao et al. [2025a], Wen et al. [2024], Chen et al. [2024]. The second is to fine-tune LLMs in KG-enriched corpora, so that the model implicitly learns the graph structure during parameter updates Wang and Liu [2025], Tian et al. [2024], Chen et al. [2025a, 2024]. While effective to some extent, both approaches face challenges: RAG suffers from incomplete or noisy retrieval, while fine-tuning is costly and often fails to generalize across domains or tasks.

Recent work has introduced the paradigm of the "reward model as reasoning": showing that reward models can go beyond scalar scoring to perform meaningful reasoning during evaluation, yielding better downstream performance than traditional reward modeling approaches Chen et al. [2025b,c]. We extend this idea to the medical KG settings, where the task is not to generate a diagnostic path but to *judge whether a candidate path is clinically valid for a given patient*. This framing is motivated by a simple but powerful observation: verification is often easier than solution. In computational theory, many problems are hard to solve but straightforward to verify Cook [2023], Sipser [2012], Gödel [1956], Swamy et al. [2025]. Likewise, searching through a large, sparse KG such as UMLS to identify the correct diagnostic path is challenging. In contrast, assessing whether a given path is clinically relevant to the patient's condition, while still requiring reasoning, can be more tractable than generating the path from scratch.

In our setting, we define *reward-model-as-reasoning* as training an LLM to assess whether a KG-derived path meaningfully connects a patient's clinical findings to their diagnoses. This goes beyond lexical or semantic similarity: the model must reason over potentially overlapping or competing concepts (e.g., "Type 1" vs. "Type 2" diabetes), determine which paths reflect appropriate causal or diagnostic mechanisms, and weigh relevance based on the patient's condition. In short, the reward model must emulate how a clinician might evaluate whether a given explanation fits the clinical picture. We explore this paradigm using the Unified Medical Language System (UMLS), one of the most widely adopted KGs in clinical informatics, maintained by the U.S. National Library of Medicine for over three decades Bodenreider [2004]. While our study is limited in scale, we take a systematic approach to investigate: (i) which task formulations and training methods best support reward-style reasoning over KG paths, and (ii) whether such capabilities transfer to broader diagnostic reasoning tasks such as diagnosis summarization and medical question answering.

2 Related Work

KG–LLM Integration for Diagnosis An increasing number of studies integrate KGs with LLMs to support diagnostic predictions. Gao et al. [2025b] propose *DR.KNOWS*, which retrieves relevant paths based on patient-specific information from UMLS KGs to improve LLMs' diagnostic predictions. Zuo et al. [2025] present an end-to-end hierarchical multi-agent framework, *KG4Diagnos*, where one primary diagnostic agent processes user queries and multiple specialized diagnostic agents collaborate to improve diagnostic predictions. Jia et al. [2024] develop *medIKAL*, in which KG knowledge assists LLMs in generating and ranking candidate diagnoses, leading to more robust performance on EMR-based tasks. Xie et al. [2025] introduce *KERAP*, a multi-agent pipeline that links patient information to KG entities, retrieves relevant knowledge, and predicts diagnoses in a zero-shot setting. Zhao et al. [2025] present *MedRAG*, which combines RAG with a structured diagnostic KG to refine EHR-based predictions. In general, these systems demonstrate that KG integration can make diagnostic LLMs more accurate, interpretable, and clinically useful.

Reward Models as Reasoning Recent work views reward modeling as a reasoning task and moves beyond simple preference scores toward models that check the correctness of a solution. Chen et al.

[2025c] frame reward modeling as a reasoning task by distilling high-quality traces and combining them with reinforcement learning and structured Chain-of-Rubrics rollouts. Guo et al. [2025] propose *RRMs* that perform explicit reasoning at inference time before scoring, training without reasoning traces but learning to use extra test-time compute on difficult cases. Chen et al. [2025b] propose *ReasonGRM*, which improves reward models by generating outcome-aligned reasoning traces, filtering them with a new $R^\star$ metric, and refining with RL on hard cases. Xu et al. [2025] introduce *DRO*, which defines a reasoning reflection reward that aligns a model's chain-of-thought with its final outcome by emphasizing reasoning-reflective tokens. In summary, reward models as reasoning go beyond scalar outcome evaluation by treating the reward model itself as a reasoning agent that generates, critiques, and aligns reasoning traces to produce more interpretable and reliable judgments.

LLM Graph Reasoning Techniques Recent work augments LLMs with explicit graph structures, moving beyond linear chain-of-thought prompting toward more faithful and interpretable reasoning. Luo et al. [2024] introduce RoG, which grounds reasoning in KG relation paths through a planning–retrieval–reasoning framework. Jin et al. [2024] present Graph-CoT, which improves LLM reasoning by alternating between generating reasoning steps and interacting with text-attributed graphs. Han et al. [2025] propose *RwG*, which builds explicit graphs from context through iterative generation and verification, enabling reasoning over structured knowledge. Hu et al. [2024] propose *GAR*, a multi-agent framework where graph nodes act as LLM agents exchanging messages, coordinated by a Master LLM through a message-passing scheme. Together, these works mark a shift from text-only prompting to graph-based methods that structure reasoning through planning, traversal, and distributed computation.

Our Contribution Building on the aforementioned work, we explore a new direction: whether fine-tuning an LLM to reason diagnostically by learning to assess the validity of a KG knowledge path. While prior work has focused on retrieval systems, symbolic pipelines, or reward models for evaluation, our work is the *first to apply* reward style path supervision directly for training diagnostic language models.

3 Data Overview

3.1 Dataset

UMLS Knowledge Graph UMLS is a large-scale resource that integrates a wide range of biomedical vocabularies and standards, maintained by U.S. National Library of Medicine over three decades. It provides comprehensive concept vocabularies from 180 biomedical sources, and defines 270 semantic relationships, which enable the construction of KGs through concepts and relations. In this work, we leverage the KG version introduced in Gao et al. [2025b], which is constructed specifically for diagnostic reasoning with physician-selected relations and concept vocabularies. This KG includes 107 relations most pertinent to diagnostic reasoning, and concepts specifically from the Systematized Nomenclature of Medicine – Clinical Terms (SNOMED CT) vocabulary. Each concept in the UMLS knowledge graph is assigned a SNOMED CT concept unique identifier (CUI) and categorized by semantic type, enabling multi-hop exploration across the graph.

ProbSum Dataset We employed the dataset introduced in the BioNLP 2023 ProbSum shared task Gao et al. [2023], which consists of daily patient progress notes paired with physician annotated diagnoses. These notes were collected from MIMIC-III Johnson et al. [2016], representing a cohort of ICU patients who on average have 5 active diagnoses Gao et al. [2023]. The training set comprises 768 progress notes covering 2,783 diagnoses, while the test set contains 237 progress notes.

MedQA The MedQA dataset Jin et al. [2020] contains multiple-choice medical questions with four answer options and one correct answer. It spans various clinical scenarios, including treatment, management, and diagnosis. For our study, we focused on diagnosis-related questions, yielding 1,796 training and 251 test samples. These were used to fine-tune and evaluate our models in a different diagnostic reasoning setting.

3.2 Building supervision paths for diagnostic reasoning

To construct diagnosis-oriented reasoning paths, we leverage both patient progress notes and corresponding diagnosis labels from the ProbSum dataset. For each patient, we first apply QuickUMLS Soldaini [2016] to extract medical concepts from the clinical note, which serve as candidate starting points in the UMLS knowledge graph. In parallel, we identify the patient's gold-standard diagnoses, also mapped to UMLS concepts. While UMLS concepts span a wide range of semantic types, not all are directly relevant to diagnosis. Accordingly, we restricted our analysis to the following categories: T033 (Finding), T037 (Injury or Poisoning), T046 (Pathologic Function), T047 (Disease or Syndrome), T048 (Mental or Behavioral Dysfunction), T049 (Cell or Molecular Dysfunction), and T184 (Sign or Symptom).

From each starting concept node, we performed a depth-first search (DFS) over the UMLS KG, exploring all connected nodes with maximum two-hop. If a path terminates at the gold-standard diagnosis concept, it is labeled as a positive path; otherwise, it is treated as a negative path (see figure 2). This procedure yields multiple path samples (positive and negative) per patient note.

To create training data, we randomly select 326 patient notes from the 768 in the ProbSum training set. Across these notes, we collect 13,377 unique starting concepts, resulting in 27,535 path-based training examples (average of 84 per note). These examples are then formatted into two task-specific datasets:

- In the path selection task, the model receives a progress note and a mixed set of candidate paths (positive and negative) and must identify the valid ones. This requires judgment, comparison, and an understanding of global coherence in the KG. It's akin to reviewing multiple differential diagnoses and determining which fit the patient context.
- In the path completion task, the model is shown a progress note and a partially completed positive path and must predict the missing step. This requires the model to simulate plausible causal or diagnostic chains, leveraging local graph structure, relation semantics, and disease progression patterns.

Evaluation All models are trained on the same pool of 27,535 samples, formatted according to the respective task setup. Because our framework follows a generative AI paradigm, we employ two widely used generative evaluation metrics: ROUGE-1 and ROUGE-L Lin [2004]. These are used for all path-specific judging tasks. Additionally, we utilize the CUI-F metric, which assesses concept-level correctness by mapping generated text back to UMLS concept vocabulary. For downstream evaluation, we adopt the metric convention of each benchmark: ROUGE-L and CUI-F for the ProbSum summarization task, and exact match accuracy for MedQA, which is framed as multiple-choice question answering.In addition to conventional metrics, we adopt PDSQI-9 Croxford et al. [2025], a clinically validated rubric for LLM-generated summaries. It includes nine criteria covering abstraction accuracy, evidence use, thoroughness, and organization. Following prior work, we use Azure GPT-4o-mini (ICC ≥ 0.8) as the LLM-as-judge to score the ProbSum test set. The Azure is a HIPPA-Complaint environment therefore obey the MIMIC Data Use Agreement.

4 Methods

4.1 Task formulation

To teach and evaluate whether LLMs can reason over medical knowledge graphs in support of diagnosis, we design five task settings corresponding to the two setup mentioned above:

- **P@10**-PATH SELECTION: The model is given 10 candidate KG paths, exactly one of which is valid.
- **P@2**-PATH SELECTION: The model is given 2 candidate KG paths, one valid and one invalid.
- **PN@10**-PATH SELECTION: The model is given 10 candidate KG paths, of which multiple may be valid, and is required to identify all correct paths.
- **NEXT-HOP PREDICTION (NHP)**-PATH COMPLETION: Given a partial KG path, the model must predict the next hop.

- **PATH COMPLETION (PC)**: Given a KG partial path, the model must predict the remaining sequence of hops to complete the path.

We hypothesize that these task settings expose complementary aspects of reasoning over the KGs. The P@10, P@2, and PN@10 settings emphasize the ability to evaluate entire candidate paths, while the NHP and PC tasks operate at the node level, testing the model's ability to capture local structure by learning different types of entities and relation edges. Figures depicting prompt structures for each formulation can be found at appendix B.

4.2 Training paradigm

We conducted our experiments using Qwen2.5-7B-Instruct Qwen et al. [2025], Qwen3-8B Yang et al. [2025], and Gemma-7B-IT Team et al. [2024]. The Qwen models are pretrained with the Mixture of Experts (MoE) paradigm, making them efficient and well-suited for complex reasoning tasks such as KG path selection. Qwen3 further introduces explicit "thinking" and "non-thinking" modes to enhance multi-step inference. The Gemma-7B-IT model complements these, offering strong instruction-following and reasoning capabilities, serving as a robust baseline for comparison.

Supervised fine-tuning (SFT) For the P@10, P@2, and PN@10 tasks, the supervision signal corresponds to identifying the valid paths among the provided candidates. For the NHP and PC tasks, the model is trained to predict either the next node or the next relation edge given the partial path. Training is performed using the cross-entropy loss over the task-specific labels. Formally, given an input x with label $y \in \{1, \ldots, K\}$, and model output probabilities $\hat{p}_\theta(y \mid x)$, the SFT loss is:

$$\mathcal{L}_{\text{SFT}}(\theta) = -\sum_{i=1}^{N} \log \hat{p}_\theta(y_i \mid x_i) \tag{1}$$

The objective of the SFT stage is to enable the model to internalize the structure of the UMLS KG by learning how different nodes are connected.

DPO Proposed by Rafailov et al. [2024], Direct Preference Optimization (DPO) method streamlines the reward modeling and tuning phases typically required in RLHF approaches. In our setting, we apply DPO to models that have already undergone SFT. Due to the maximum likelihood objective of DPO (equation 2), the model is encouraged to assign higher probability to valid paths relative to invalid ones. We theorize that by framing our task as a preference-based problem, the fine-tuned models can be further aligned to reliably select positive candidate paths.

$$\mathcal{L}_{DPO}(\pi_\theta; \pi_{ref}) = -\mathbb{E}_{(x,y_w,y_l)\sim D}[\log \sigma(\beta \log \frac{\pi_\theta(y_w|x)}{\pi_{ref}(y_w|x)} - \beta \log \frac{\pi_\theta(y_w|x)}{\pi_{ref}(y_w|x)})] \tag{2}$$

GRPO In our path judging paradigm, each training instance includes a group of candidate paths (e.g., 10 paths in P@10 or PN@10), with the objective of assigning higher likelihood to valid diagnostic paths. Group Relative Policy Optimization (GRPO) Shao et al. [2024] is well suited for this structure, as it directly optimizes over entire groups of candidates by reinforcing the relative preference for correct paths across the set. This allows the model to learn richer path selection strategies beyond isolated pairwise ranking. We define a discrete reward function in Equation 4, which provides a positive signal exclusively when the model produces the correct label.

$$\begin{aligned}\mathcal{L}_{\text{GRPO}}(\theta) = \mathbb{E}_{q\sim P(Q),\, \{o_i\}_{i=1}^{G}\sim \pi_\theta^{\text{old}}(O|q)} \Bigg[\frac{1}{G}\sum_{i=1}^{G} \frac{1}{|o_i|} \sum_{t=1}^{|o_i|} \min \Bigg(\frac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_\theta^{\text{old}}(o_{i,t} \mid q, o_{i,<t})} \hat{A}_{i,t}, \\ \text{clip}\Big(\frac{\pi_\theta(o_{i,t} \mid q, o_{i,<t})}{\pi_\theta^{\text{old}}(o_{i,t} \mid q, o_{i,<t})}, 1-\epsilon, 1+\epsilon \Big) \hat{A}_{i,t} \Bigg) \Bigg] \quad - \beta\, D_{\text{KL}}(\pi_\theta \| \pi_{\text{ref}})\end{aligned} \tag{3}$$

$$\mathcal{R}(x, y) = \begin{cases} 1, & \text{if the model prediction } \hat{y} = y \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Model Merging This approach combines two models fine-tuned on different tasks to produce a single model that incorporates aspects of both, without requiring additional training. The combination is achieved by iterating over the parameters of the two models and computing either a simple average Wortsman et al. [2022] or a weighted average Matena and Raffel [2022] of the weights. Formally, performing a weighted average in our case looks like:

$$\theta_{\text{merged}} = \lambda \cdot \theta_1 + (1 - \lambda) \cdot \theta_2$$

We have merged P@10 SFT model with P@2 and PN@10 models. In both cases, θ_1 corresponds to P@10 model parameters. Furthermore, we given higher importance to the parameters of P@10 by setting $\lambda = 0.7$.

Reasoning Distillation Recent approaches frame reward model training as a reasoning task, where Chain-of-Thought (CoT) supervision enhances reward signal quality by modeling deeper inference steps Chen et al. [2025d]. In our reasoning distillation setup, we sampled 2,577 training examples and used GPT-o3-mini (Microsoft Azure, HIPAA-compliant) to generate explanations for why a correct KG path should be preferred over alternatives (from the P@10 setting), given a patient note and ground-truth diagnoses. These high-quality traces then supervised the reward model under multiple training paradigms, in strict accordance with the MIMIC Data Use Agreement.

RM-R1 Chen et al. [2025c] introduced Reasoning Reward Models (REASRMS) and proposed a reasoning-oriented training framework, RM-R1, which operates in two stages: (1) distillation of reasoning traces, and (2) reinforcement learning with verifiable rewards.

We adopted this framework by first performing SFT on the generated reasoning traces. This stage trains the model to produce reasoning traces and subsequently predict the preferred candidate path. Following the approach in Chen et al. [2025d], we then applied GRPO using a rule-based reward function. Formally, this reward function is defined as:

$$\mathcal{R}(x, j \mid y_a, y_b) = \begin{cases} 1, & \text{if } \hat{l} = l \\ -1, & \text{otherwise} \end{cases}$$

DogeRM Lin et al. [2024] introduced the Domain Knowledge Merged Reward Model (DOGERM), which merges a general reward model with a task-specific SFT model. We adopted this strategy by merging our reasoning model, fine-tuned under the RM-R1 framework, with the model fine-tuned on the P@10 task. We follow the same process as our model merging technique with a simple average of the parameters.

$$\theta_{\text{MERGE}} = 0.5 \cdot \theta_{\text{SFT}} + 0.5 \cdot \theta_{\text{RM}}$$

Distilling Step by Step (DSS) Hsieh et al. [2023]proposed the Distilling Step by Step framework, which extends reasoning distillation by treating reasoning generation and path prediction as two distinct tasks. Specifically, at each training step, the model is prompted twice: first to produce a reasoning trace, and then to predict the correct KG path. The reasoning traces obtained from earlier methods serve as supervision for the reasoning task, while path prediction aligns with the P@10 setting. During training, losses for both tasks are computed according to Equation 1 and combined to optimize the model jointly.

$$\mathcal{L}_{total} = 0.5 * \mathcal{L}_{\text{path}} + 0.5 * \mathcal{L}_{\text{rationale}}$$

5 Results

This work investigates two central questions: (i) Can LLMs be trained as reward-style models to verify the relevance of KG paths for diagnosis? (ii) Can such learned reasoning transfer to downstream diagnostic tasks, such as summarization or medical QA?

Tables 1 and 2 present results addressing the first research question. These tables compare the performance of various training paradigms across different path selection settings, allowing us to identify the most effective training strategies. In particular, Table 1 examines the impact of different

SFT settings	Test on P@10		Test on P@2		Test on NHP		Test on PC		Test on PN@10	
	Rouge-1	Rouge-L	Rouge-1	Rouge-L	Rouge-1	Rouge-L	Rouge-1	Rouge-L	Rouge-1	Rouge-L
No SFT	19.34	18.79	78.78	78.50	1.19	1.20	4.27	3.90	28.24	27.45
SFT on P@10	52.93	52.64	33.98	33.31	6.26	6.22	6.64	6.35	43.14	42.65
SFT on P@2	37.59	36.69	74.84	74.59	6.52	6.33	7.17	6.74	30.39	29.64
SFT on NHP	8.20	7.89	8.45	7.83	16.14	16.09	14.33	14.24	11.34	10.61
SFT on PC	45.62	44.74	69.90	69.25	8.72	8.69	18.52	18.35	8.51	8.12
SFT on PN@10	12.53	12.14	73.89	73.65	1.37	1.37	9.05	8.84	18.34	17.93

Table 1: Rouge-1 and Rouge-L scores for different SFT configurations of Qwen2.5-7B-Instruct across five tasks. Gray-highlighted cells indicate the scores when the training task matches the test task, demonstrating task-specific effectiveness.

path selection finetuning configurations for the Qwen2.5-7B-Instruct model. Additionally, each SFT model was evaluated across all other path selection settings to assess its generalization performance.

In Table 1, the highlighted cells correspond to models that were both finetuned and evaluated on the same setting, where strong performance relative to other models would generally be expected. Interestingly, for the P@2 and PN@10 settings, the finetuned models perform worse than their non-SFT counterparts. Moreover, the NHP and PC settings appear to be the most challenging, exhibiting the lowest overall performance across models.

Training Paradigm	Task	Qwen2.5-7B-Instruct			Qwen3-8B			Gemma-7B-IT		
		Rouge-1	Rouge-L	CUI-F	Rouge-1	Rouge-L	CUI-F	Rouge-1	Rouge-L	CUI-F
No SFT (baseline)		19.34	18.79	22.49	4.82	4.66	02.19	26.99	26.09	28.33
SFT	P@10	52.93	52.64	57.01	18.77	18.21	27.15	17.45	16.77	13.74
	P@2	37.59	36.69	32.24	53.17	52.61	38.82	38.68	37.93	43.05
	PC	45.62	44.74	59.73	3.18	3.11	02.28	16.74	16.33	20.59
	PN@10	12.53	12.14	14.80	7.65	7.02	04.25	23.83	22.84	23.09
SFT Model Merge	0.7 P@10 + 0.3P@2	63.16	62.35	54.35	4.70	4.53	03.60	23.34	22.58	17.24
	0.7 P@10 + 0.3PN@10	62.94	61.89	42.07	4.35	4.19	05.07	17.36	16.38	11.72
SFT + DPO	P@10	15.90	15.06	14.80	29.35	28.92	04.25	18.01	17.22	23.09
SFT + GRPO		**85.03**	**84.96**	**91.38**	20.04	19.58	22.87	17.70	17.07	11.72
DogeRM		40.73	39.97	33.42	5.55	5.38	00.30	19.79	18.60	12.89
RM-R1	P@10	46.25	45.64	44.04	5.66	5.47	03.75	20.63	19.24	17.57
DSS		59.50	59.03	53.70	73.67	72.95	73.50	84.26	84.22	71.15
DSS + GRPO		59.67	59.27	51.13	**77.24**	**76.91**	**75.25**	**84.43**	**84.41**	**72.27**

Table 2: Rouge-1, Rouge-L and CUI-F scores comparing various training techniques across Qwen2.5-7B-Instruct, Qwen3-8B, and Gemma-7B-IT. Bold values indicate the best training paradigm for each model, while green-highlighted cells show the second-best training paradigms.

Table 2 compares the different training paradigms. The models are evaluated on the test setting of P@10. DSS combined with GRPO consistently achieves the highest Rouge-1, Rouge-L and CUI-F scores across all models, highlighting its effectiveness for KG path judgment. In contrast, standard SFT and reward-based methods like DPO and DogeRM underperform, with gains from model merging remaining modest. Tables 3, 4, and 5 address the question of whether our reward models be effectively applied to diagnostic prediction tasks. For this analysis, we selected the top three performing settings for each model from Table 2 and evaluated their ability to generate diagnoses. In Table 3, patient records from the ProbSum test set were provided as input, and the models were tasked with predicting the patient's diagnoses. To provide a benchmark, we also evaluated non-finetuned models as well as models finetuned directly on the diagnosis prediction task.

We observe that the non-finetuned Gemma-7B-IT and Qwen2.5-7B-Instruct models outperform the non-finetuned Qwen3-8B model on the diagnosis prediction task. After finetuning specifically for diagnosis prediction, the performance of all three models becomes comparable. Notably, the KG-based training strategies explored in this study do not yield significant improvements in diagnosis prediction performance.

Results on the MedQA dataset are summarized in Table 4. For this evaluation, we also trained task-specific models as baseline. Notably, for Qwen2.5-7B and Qwen3-8B, our training strategies show modest improvements over the baseline models.

Table 5 presents the PDSQI-9 scores for the best-performing models identified in Table 3, alongside non-finetuned models as a baseline. For this evaluation, patient records from the ProbSum test set

	Qwen2.5-7B-Instruct			Qwen3-8B			Gemma-7B-IT		
	Training	Rouge-L	CUI-F	Training	Rouge-L	CUI-F	Training	Rouge-L	CUI-F
Baseline	No SFT	21.10	24.45	No SFT	5.73	12.07	No SFT	22.50	24.61
	SFT	22.71	26.02	SFT	22.19	25.44	SFT	20.74	29.26
Reward-like Reasoning SFT	SFT Model Merge 0.7 P@10 + 0.3 P@2	21.70	24.39	SFT P@2	19.47	12.43	SFT P@2	15.00	15.66
	SFT Model Merge 0.7 P@10 + 0.3 PN@10	21.60	25.70	DSS	17.18	23.24	DSS	22.17	24.30
	SFT + GRPO	12.98	15.86	DSS + GRPO	15.93	25.43	DSS + GRPO	22.17	24.31

Table 3: Rouge-L and CUI-F scores for diagnosis prediction on ProbSum test set. Top three performing training paradigms for each model from 2 are evaluated alongside baseline methods (no SFT and SFT for diagnosis prediction task).

Training	Qwen2.5-7B-Instruct	Qwen3-8B	Gemma-7B-IT
No SFT	63.74	44.22	31.08
SFT on MedQA	63.34	50.59	50.59
Best training paradigm from the checkpoints in Table 3	64.90	56.97	43.42

Table 4: QA results on MedQA dataset. Best performing training paradigm for each model from 3 are evaluated alongside baseline methods (no SFT and SFT on MedQA).

were provided as input, and the models were tasked with generating both the diagnosis and the corresponding reasoning. Even in this setting, we observe inconsistencies among the finetuned models. Specifically, Qwen2.5-7B-Instruct shows only marginal improvements across the metrics, while Qwen3-8B and Gemma-7B-IT exhibit decreases in most metrics relative to their baselines.

Collectively, Tables 3, 4, and 5 indicate that proficiency in identifying valid KG paths does not necessarily translate to improved performance on downstream tasks such as diagnostic reasoning.

6 Discussion: Brittleness and Promise

Our findings highlight both the brittleness and the emerging promise of fine tuning LLMs on KG reasoning tasks. Across a series of SFT experiments, we observe that models often perform well on the format they are trained on, yet struggle to generalize to even closely related reasoning formats. This inconsistency reveals a persistent challenge in aligning LLMs with the structured, multi hop logic of KG based clinical reasoning.

Brittleness: shallow adaptation and fragile generalization Table 1 shows that models fine tuned on one task (e.g., P@10 or NHP) exhibit strong performance on that specific test but often perform worse than the no SFT baseline on other reasoning tasks. This suggests that SFT primarily teaches models to conform to a particular task format, rather than imparting deeper reasoning capabilities over the KG.

PDSQI-9 Metrics	Baseline (no SFT)			Best output from Table 3 per LLM		
	Qwen2.5-7B-Instruct	Qwen3-8B	Gemma-7B-IT	Qwen2.5-7B-Instruct (SFT Model Merge 0.7 P@10 + 0.3 P@2)	Qwen3-8B (SFT P@2)	Gemma-7B-IT (DSS)
Accuracy Extractive	1.62	2.04	1.70	1.76 ↑	1.46	1.54
Thoroughness	1.94	1.68	1.84	2.00 ↑	1.44	1.50
Usefulness	2.48	2.20	2.24	2.44	1.64	1.94
Organization	3.68	2.30	3.32	3.80 ↑	1.84	2.70
Comprehensibility	4.54	3.28	4.50	4.58 ↑	2.84	4.64 ↑
Succinctness	2.70	2.10	3.10	2.78 ↑	1.38	3.92 ↑
Synthesis Abstraction	1.80	2.18	1.60	2.18 ↑	1.20	0.54
Average Score	2.68	2.25	2.61	2.79 ↑	1.68	2.39

Table 5: ProbSum results using PDSQI-9 criteria with GPT-o4-mini as the judge.

```
Prompt:
..............Patient Progress Notes ..............
Given the PATH 'Elevated k->has_member|Chronic kidney
disease (smq)->member_of', predict the next hop of
the PATH

Model: |Hyperkalemia
Actual: |K excess
```

(a) Qwen2.5-7B-Instruct SFT on NHP

```
Prompt:
..............Patient Progress Notes ..............
Given the PATH 'Elevated k', complete the rest
of the PATH.

Model: ->subset_includes_item
Actual: ->has_member|Chronic kidney disease (smq)
        ->member_of|K excess
```

(b) Qwen2.5-7B-Instruct SFT on PC

Figure 1: Sample responses from Qwen2.5-7B-Instruct for NHP and PC tasks.

Error analysis (Figure 1) further underscores this limitation. Models frequently hallucinate incorrect relations, terminate paths prematurely, or output synonyms (e.g., "Hyperkalemia") instead of canonical KG terms (e.g., "K excess"), breaking alignment with ground truth outputs. In other cases, models generate relations that are semantically equivalent to the labeled ones but differ in surface form. For example, predicting '→subset_includes_item' instead of the labeled '→has_member'.

We further observe that models fine tuned on NHP (Next Hop Prediction) struggle when transferred to P@10 tasks, particularly when a clear starting node is missing. In such cases, the model often begins its path from an arbitrary node, resulting in broken or incomplete reasoning chains.

Promise: very specific reward optimization and checkpoint merging Despite this brittleness, we also find signals of promise. As shown in Table 2 and 5, the Qwen2.5-7B-Instruct model fine tuned using our GRPO method outperforms all other variants, including baselines and other distillation setups, on human aligned PDSQI-9 criteria. Improvements are seen not only in surface qualities like organization and succinctness, but also in deeper aspects such as extractive accuracy and abstract synthesis. This suggests that when the reward signal is well aligned and the model is guided by end task utility, it can meaningfully integrate KG derived reasoning into its generative output. Interestingly, this promise is model specific: only one LLM (Qwen2.5-7B) showed consistent gains across multiple criteria. Others (e.g., Gemma-7b-it) performed well on select metrics (e.g., succinctness) but failed on others, highlighting that model architecture and initialization likely affect how reasoning priors interact with KG based supervision.

Overall, our results point to a narrow path forward: while current SFT approaches do not reliably induce generalizable reasoning over knowledge graphs, carefully aligned reward based tuning (e.g., GRPO) holds potential for unlocking clinically useful capabilities. It is also important to note that our experiments use a single knowledge graph (UMLS) as the underlying structure. While other KGs may offer different reasoning affordances, UMLS is one of the most widely adopted in clinical applications, and our findings are therefore likely to reflect meaningful patterns in KG based diagnostic reasoning more broadly.

To move beyond brittle adaptations, future work must focus on strategies that emphasize structural reasoning, canonical grounding, and consistency across task formats. These may include curriculum learning, constrained decoding over KG structures, and richer reward modeling that integrates both symbolic accuracy and language fluency. As LLMs continue to evolve, the interplay between symbolic supervision and generative capacity offers a promising, though still fragile, frontier.

7 Conclusion

Framing reasoning as a reward-like judgment task, we introduced a suite of path-based training setups and evaluated their effects on both KG verification and downstream diagnostic tasks. While certain strategies, such as GRPO and CoT distillation, enhance the model's ability to assess KG path validity, their benefits do not consistently generalize to broader clinical reasoning settings. Our findings reveal a narrow corridor of promise: reward supervision over KGs can encode reasoning signals, but realizing their full clinical utility will require deeper integration of structure, context, and generative capabilities.

8 Acknowledgments

This work is supported by U.S. National Library of Medicine R00 LM014308.

References

Ethan Goh, Robert Gallo, Jason Hom, Eric Strong, Yingjie Weng, Hannah Kerman, Joséphine A Cool, Zahir Kanjee, Andrew S Parsons, Neera Ahuja, et al. Large language model influence on diagnostic reasoning: a randomized clinical trial. *JAMA network open*, 7(10):e2440969–e2440969, 2024.

Guangyu Wang and Xiaohong Liu. Medical large language model for diagnostic reasoning across specialties, 2025.

Xiaohong Liu, Hao Liu, Guoxing Yang, Zeyu Jiang, Shuguang Cui, Zhaoze Zhang, Huan Wang, Liyuan Tao, Yongchang Sun, Zhu Song, et al. A generalist medical language model for disease diagnosis assistance. *Nature medicine*, 31(3):932–942, 2025.

Yubin Kim, Hyewon Jeong, Shan Chen, Shuyue Stella Li, Mingyu Lu, Kumail Alhamoud, Jimin Mun, Cristina Grau, Minseok Jung, Rodrigo Gameiro, et al. Medical hallucinations in foundation models and their impact on healthcare. *arXiv preprint arXiv:2503.05777*, 2025.

SM Tonmoy, SM Zaman, Vinija Jain, Anku Rani, Vipula Rawte, Aman Chadha, and Amitava Das. A comprehensive survey of hallucination mitigation techniques in large language models. *arXiv preprint arXiv:2401.01313*, 6, 2024.

Elham Asgari, Nina Montaña-Brown, Magda Dubois, Saleh Khalil, Jasmine Balloch, Joshua Au Yeung, and Dominic Pimenta. A framework to assess clinical safety and hallucination rates of llms for medical text summarisation. *npj Digital Medicine*, 8(1):274, 2025.

Olivier Bodenreider. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32 Database issue:D267–70, 2004. URL `https://api.semanticscholar.org/CorpusID:205228801`.

Yanjun Gao, Ruizhe Li, Emma Croxford, John Caskey, Brian W Patterson, Matthew Churpek, Timothy Miller, Dmitriy Dligach, and Majid Afshar. Leveraging medical knowledge graphs into large language models for diagnosis prediction: Design and application study. *Jmir Ai*, 4:e58670, 2025a.

Kaiwen Zuo, Yirui Jiang, Fan Mo, and Pietro Lio. Kg4diagnosis: A hierarchical multi-agent llm framework with knowledge graph enhancement for medical diagnosis. In *AAAI Bridge Program on AI for Medicine and Healthcare*, pages 195–204. PMLR, 2025.

Punam Bedi, Anjali Thukral, and Shivani Dhiman. Xlr-kgdd: leveraging llm and rag for knowledge graph-based explainable disease diagnosis using multimodal clinical information. *Knowledge and Information Systems*, pages 1–21, 2025.

Yilin Wen, Zifeng Wang, and Jimeng Sun. Mindmap: Knowledge graph prompting sparks graph of thoughts in large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10370–10388, 2024.

Zhongwu Chen, Long Bai, Zixuan Li, Zhen Huang, Xiaolong Jin, and Yong Dou. A new pipeline for knowledge graph reasoning enhanced by large language models without fine-tuning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1366–1381, 2024.

Shiyu Tian, Yangyang Luo, Tianze Xu, Caixia Yuan, Huixing Jiang, Chen Wei, and Xiaojie Wang. KG-adapter: Enabling knowledge graph integration in large language models through parameter-efficient fine-tuning. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3813–3828, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.229. URL `https://aclanthology.org/2024.findings-acl.229/`.

Hanzhu Chen, Xu Shen, Jie Wang, Zehao Wang, Qitan Lv, Junjie He, Rong Wu, Feng Wu, and Jieping Ye. Knowledge graph finetuning enhances knowledge manipulation in large language models. In *The Thirteenth International Conference on Learning Representations*, 2025a.

Bin Chen, Xinzge Gao, Chuanrui Hu, Penghang Yu, Hua Zhang, and Bing-Kun Bao. Reasongrm: Enhancing generative reward models through large reasoning models. *arXiv preprint arXiv:2506.16712*, 2025b.

Xiusi Chen, Gaotang Li, Ziqi Wang, Bowen Jin, Cheng Qian, Yu Wang, Hongru Wang, Yu Zhang, Denghui Zhang, Tong Zhang, et al. Rm-r1: Reward modeling as reasoning. *arXiv preprint arXiv:2505.02387*, 2025c.

Stephen A Cook. The complexity of theorem-proving procedures. In *Logic, automata, and computational complexity: The works of Stephen A. Cook*, pages 143–152. 2023.

Michael Sipser. *Introduction to the Theory of Computation*. Cengage Learning, 3rd edition, 2012. ISBN 9781133187790.

Kurt Gödel. Letter to john von neumann. Manuscript, 1956. Available at `https://ecommons.cornell.edu/server/api/core/bitstreams/46aef9c4-288b-457d-ab3e-bb6cb1a4b88e/content`.

Gokul Swamy, Sanjiban Choudhury, Wen Sun, Zhiwei Steven Wu, and J Andrew Bagnell. All roads lead to likelihood: The value of reinforcement learning in fine-tuning. *arXiv preprint arXiv:2503.01067*, 2025.

Yanjun Gao, Ruizhe Li, Emma Croxford, John Caskey, Brian W Patterson, Matthew Churpek, Timothy Miller, Dmitriy Dligach, and Majid Afshar. Leveraging medical knowledge graphs into large language models for diagnosis prediction: Design and application study. *JMIR AI*, 4:e58670, February 2025b. ISSN 2817-1705. doi: 10.2196/58670. URL `http://dx.doi.org/10.2196/58670`.

Mingyi Jia, Junwen Duan, Yan Song, and Jianxin Wang. medikal: Integrating knowledge graphs as assistants of llms for enhanced clinical diagnosis on emrs. *arXiv preprint arXiv:2406.14326*, 2024.

Yuzhang Xie, Hejie Cui, Ziyang Zhang, Jiaying Lu, Kai Shu, Fadi Nahab, Xiao Hu, and Carl Yang. Kerap: A knowledge-enhanced reasoning approach for accurate zero-shot diagnosis prediction using multi-agent llms. *arXiv preprint arXiv:2507.02773*, 2025.

Xuejiao Zhao, Siyan Liu, Su-Yin Yang, and Chunyan Miao. Medrag: Enhancing retrieval-augmented generation with knowledge graph-elicited reasoning for healthcare copilot. In *Proceedings of the ACM on Web Conference 2025*, pages 4442–4457, 2025.

Jiaxin Guo, Zewen Chi, Li Dong, Qingxiu Dong, Xun Wu, Shaohan Huang, and Furu Wei. Reward reasoning model. *arXiv preprint arXiv:2505.14674*, 2025.

Yifei Xu, Tusher Chakraborty, Srinagesh Sharma, Leonardo Nunes, Emre Kıcıman, Songwu Lu, and Ranveer Chandra. Direct reasoning optimization: Llms can reward and refine their own reasoning for open-ended tasks. *arXiv preprint arXiv:2506.13351*, 2025.

Linhao Luo, Yuan-Fang Li, Gholamreza Haffari, and Shirui Pan. Reasoning on graphs: Faithful and interpretable large language model reasoning. In *The Twelfth International Conference on Learning Representations*, 2024. URL `https://openreview.net/forum?id=ZGNWW7xZ6Q`.

Bowen Jin, Chulin Xie, Jiawei Zhang, Kashob Kumar Roy, Yu Zhang, Zheng Li, Ruirui Li, Xianfeng Tang, Suhang Wang, Yu Meng, et al. Graph chain-of-thought: Augmenting large language models by reasoning on graphs. *arXiv preprint arXiv:2404.07103*, 2024.

Haoyu Han, Yaochen Xie, Hui Liu, Xianfeng Tang, Sreyashi Nag, William Headden, Yang Li, Chen Luo, Shuiwang Ji, Qi He, et al. Reasoning with graphs: Structuring implicit knowledge to enhance llms reasoning. *arXiv preprint arXiv:2501.07845*, 2025.

Yuwei Hu, Runlin Lei, Xinyi Huang, Zhewei Wei, and Yongchao Liu. Scalable and accurate graph reasoning with llm-based multi-agents. *arXiv preprint arXiv:2410.05130*, 2024.

Yanjun Gao, Dmitriy Dligach, Timothy Miller, and Majid Afshar. Overview of the problem list summarization (ProbSum) 2023 shared task on summarizing patients' active diagnoses and problems from electronic health record progress notes. In Dina Demner-fushman, Sophia Ananiadou, and Kevin Cohen, editors, *Proceedings of the 22nd Workshop on Biomedical Natural Language Processing and BioNLP Shared Tasks*, pages 461–467, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.bionlp-1.43. URL https://aclanthology.org/2023.bionlp-1.43/.

Alistair E. W. Johnson, Tom J. Pollard, Lu Shen, Li wei H. Lehman, Mengling Feng, Mohammad Mahdi Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G. Mark. Mimic-iii, a freely accessible critical care database. *Scientific Data*, 3, 2016. URL https://api.semanticscholar.org/CorpusID:33285731.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. What disease does this patient have? a large-scale open domain question answering dataset from medical exams, 2020. URL https://arxiv.org/abs/2009.13081.

Luca Soldaini. Quickumls: a fast, unsupervised approach for medical concept extraction. 2016. URL https://api.semanticscholar.org/CorpusID:2990304.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

Emma Croxford, Yanjun Gao, Nicholas Pellegrino, Karen K. Wong, Graham Wills, Elliot First, Miranda Schnier, Kyle Burton, Cris G. Ebby, Jillian Gorskic, Matthew Kalscheur, Samy Khalil, Marie Pisani, Tyler Rubeor, Peter Stetson, Frank Liao, Cherodeep Goswami, Brian Patterson, and Majid Afshar. Development and validation of the provider documentation summarization quality instrument for large language models, 2025. URL https://arxiv.org/abs/2501.08977.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, Amélie Héliou, Andrea Tacchetti, Anna Bulanova, Antonia Paterson, Beth Tsai, Bobak Shahriari, Charline Le Lan, Christopher A. Choquette-Choo, Clément Crepy, Daniel Cer, Daphne Ippolito, David Reid, Elena Buchatskaya, Eric Ni, Eric Noland, Geng Yan, George Tucker, George-Christian Muraru, Grigory Rozhdestvenskiy, Henryk Michalewski, Ian Tenney, Ivan Grishchenko, Jacob Austin, James Keeling, Jane Labanowski, Jean-Baptiste Lespiau, Jeff Stanway, Jenny Brennan, Jeremy Chen, Johan Ferret, Justin Chiu, Justin Mao-Jones, Katherine Lee, Kathy Yu, Katie Millican, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon, Machel Reid, Maciej Mikuła, Mateo Wirth, Michael Sharman, Nikolai Chinaev, Nithum Thain, Olivier Bachem, Oscar Chang, Oscar Wahltinez, Paige Bailey, Paul Michel, Petko Yotov, Rahma Chaabouni, Ramona Comanescu, Reena Jana, Rohan Anil, Ross McIlroy, Ruibo Liu, Ryan Mullins, Samuel L Smith, Sebastian Borgeaud, Sertan Girgin, Sholto Douglas, Shree Pandya, Siamak Shakeri, Soham De, Ted Klimenko, Tom Hennigan, Vlad Feinberg, Wojciech Stokowiec, Yu hui Chen, Zafarali Ahmed,

Zhitao Gong, Tris Warkentin, Ludovic Peran, Minh Giang, Clément Farabet, Oriol Vinyals, Jeff Dean, Koray Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani, Douglas Eck, Joelle Barral, Fernando Pereira, Eli Collins, Armand Joulin, Noah Fiedel, Evan Senter, Alek Andreev, and Kathleen Kenealy. Gemma: Open models based on gemini research and technology, 2024. URL https://arxiv.org/abs/2403.08295.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2024. URL https://arxiv.org/abs/2305.18290.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024. URL https://arxiv.org/abs/2402.03300.

Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time, 2022. URL https://arxiv.org/abs/2203.05482.

Michael Matena and Colin Raffel. Merging models with fisher-weighted averaging, 2022. URL https://arxiv.org/abs/2111.09832.

Xiusi Chen, Gaotang Li, Ziqi Wang, Bowen Jin, Cheng Qian, Yu Wang, Hongru Wang, Yu Zhang, Denghui Zhang, Tong Zhang, Hanghang Tong, and Heng Ji. Rm-r1: Reward modeling as reasoning, 2025d. URL https://arxiv.org/abs/2505.02387.

Tzu-Han Lin, Chen-An Li, Hung yi Lee, and Yun-Nung Chen. Dogerm: Equipping reward models with domain knowledge through model merging, 2024. URL https://arxiv.org/abs/2407.01470.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes, 2023. URL https://arxiv.org/abs/2305.02301.

A Path Extraction Workflow

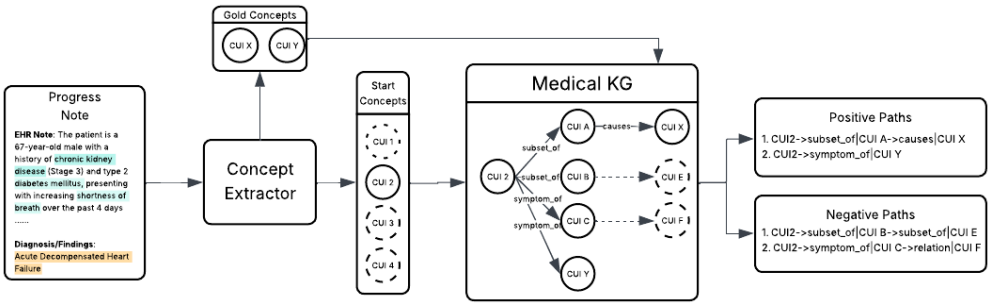

Figure 2: Workflow depicting UMLS KG paths extraction. A concept extractor like QuickUMLS is used to extract starting and gold concepts from the patient progress notes. Then, 2-HOP DFS is performed starting from each starting node. Paths that end with one of the gold concepts become part of the positive path set, while paths that do not end with a gold concept become part of the negative path set.

B Prompt formats of different task formulations

Here, we present examples illustrating the prompt structure for each task formulation used during the fine-tuning process. In each case, we provide the patient progress notes. For path selection tasks,

we include several candidate paths and ask the model to identify the valid one(s). Figures 3, 4, and 7 show examples of tasks P@10, P@2, and PN@10, respectively. For path completion tasks, we provide a partial KG path and ask the model to either predict the next hop or complete the remaining portion of the path. Figures 6 and 5 illustrate the formats of tasks NHP and PC, respectively. During model evaluation on these tasks, the prompt format remains the same, except that the DIAGNOSIS and FINDINGS information is removed to test whether the model can predict the correct KG path solely from the patient progress notes.

```
Prompt:

You are a medical assistance tool provided with the PROGRESS_NOTES and the
DIAGNOSES AND FINDINGS of a patient. You must review the knowledge base
paths in CANDIDATE_PATHS based on concepts from PROGRESS_NOTES and select
ONE PATH which can lead to one of the provided DIAGNOSES and FINDINGS.

Each path in CANDIDATE_PATHS follow the format: 'Node1->Relation|Node2->.'
where 'Node1' and 'Node2' signify the concept nodes in the knowledge graph
and 'Relation' signifies the relation between 'Node1' and 'Node2'.
The '->' symbol signifies a single hop from Node1 to Node2 via Relation
edge.

PROGRESS_NOTES:
...............Patient Progress Notes ...............

DIAGNOSES and FINDINGS:
...................................................

CANDIDATE_PATHS:
1. Rash->has_member|Hypersensitivity (smq)->member_of|Skin response
2. Rash->has_member|Hypersensitivity (smq)->member_of|Blister
....
10. Rash->has_member|Anaphylactic reaction (smq)->member_of|Cheek swell

Pick the path which is valid for a diagnosis.\nOutput:
```

Figure 3: Prompt Structure for Task P@10

```
Prompt:

You are a medical assistance tool provided with the PROGRESS_NOTES and the
DIAGNOSES AND FINDINGS of a patient. You must review the knowledge base
paths in CANDIDATE_PATHS based on concepts from PROGRESS_NOTES and select
ONE PATH which can lead to one of the provided DIAGNOSES and FINDINGS.

Each path in CANDIDATE_PATHS follow the format: 'Node1->Relation|Node2->.'
where 'Node1' and 'Node2' signify the concept nodes in the knowledge graph
and 'Relation' signifies the relation between 'Node1' and 'Node2'.
The '->' symbol signifies a single hop from Node1 to Node2 via Relation
edge.

PROGRESS_NOTES:
...............Patient Progress Notes ...............

DIAGNOSES and FINDINGS:
...................................................

CANDIDATE_PATHS:
1. Rash->has_member|Hypersensitivity (smq)->member_of|Skin response
2. Rash->has_member|Hypersensitivity (smq)->member_of|Blister

Pick the path which is valid for a diagnosis.\nOutput:
```

Figure 4: Prompt Structure for Task P@2

```
Prompt:

You are a medical assistance tool provided with the PROGRESS_NOTES and the
DIAGNOSES AND FINDINGS of a patient. You are given a partial path
(PARTIAL_PATH) based on concepts from PROGRESS_NOTES. You are
required to predict the next hop of the path so that it can reach
one of the concepts from DIAGNOSES AND FINDINGS.

Each path in CANDIDATE_PATHS follow the format: 'Node1->Relation|Node2->.'
where 'Node1' and 'Node2' signify the concept nodes in the knowledge graph
and 'Relation' signifies the relation between 'Node1' and 'Node2'.
The '->' symbol signifies a single hop from Node1 to Node2 via Relation
edge.

PROGRESS_NOTES:
...............Patient Progress Notes ...............

DIAGNOSES and FINDINGS:
...................................................

Given the PATH 'Rash->has_member|Hypersensitivity (smq)', predict
the next relation edge that can be explored to reach a valid diagnosis.
\nOutput:
```

Figure 5: Prompt Structure for Task NHP

```
Prompt:

You are a medical assistance tool provided with the PROGRESS_NOTES and the
DIAGNOSES AND FINDINGS of a patient. You are given a partial path
(PARTIAL_PATH) based on concepts from PROGRESS_NOTES. You are
required to complete the rest of the path so that it can reach
one of the concepts from DIAGNOSES AND FINDINGS.

Each path in CANDIDATE_PATHS follow the format: 'Node1->Relation|Node2->.'
where 'Node1' and 'Node2' signify the concept nodes in the knowledge graph
and 'Relation' signifies the relation between 'Node1' and 'Node2'.
The '->' symbol signifies a single hop from Node1 to Node2 via Relation
edge.

PROGRESS_NOTES:
...............Patient Progress Notes ...............

DIAGNOSES and FINDINGS:
.....................................................

Given the PATH 'Rash->has_member|Hypersensitivity (smq)', predict the
complete path which can reach a valid diagnosis.
\nOutput:
```

Figure 6: Prompt Structure for Task PC

```
Prompt:

You are a medical assistance tool provided with the PROGRESS_NOTES and the
DIAGNOSES AND FINDINGS of a patient. You must review the knowledge base
paths in CANDIDATE_PATHS based on concepts from PROGRESS_NOTES and select
all the PATHs which can lead to one of the provided DIAGNOSES and FINDINGS.

Each path in CANDIDATE_PATHS follow the format: 'Node1->Relation|Node2->.'
where 'Node1' and 'Node2' signify the concept nodes in the knowledge graph
and 'Relation' signifies the relation between 'Node1' and 'Node2'.
The '->' symbol signifies a single hop from Node1 to Node2 via Relation
edge.

PROGRESS_NOTES:
...............Patient Progress Notes ...............

DIAGNOSES and FINDINGS:
.....................................................

CANDIDATE_PATHS:
1. Rash->has_member|Hypersensitivity (smq)->member_of|Skin response
2. Rash->has_member|Hypersensitivity (smq)->member_of|Blister
....
10. Rash->has_member|Anaphylactic reaction (smq)->member_of|Cheek swell

Pick all the paths which can lead to valid diagnoses.\nOutput:
```

Figure 7: Prompt Structure for Task PN@10