

End-to-End Agentic RAG System Training for Traceable Diagnostic Reasoning

Qiaoyu Zheng^{1,2}, Yuze Sun¹, Chaoyi Wu¹, Weike Zhao^{1,2}, Pengcheng Qiu^{1,2},
Yongguo Yu³, Kun Sun³, Yanfeng Wang^{1,2}, Ya Zhang^{1,2,†} and Weidi Xie^{1,2,†}

¹Shanghai Jiao Tong University, Shanghai, China ²Shanghai AI Laboratory, Shanghai, China

³Xinhua Hospital affiliated to Shanghai Jiao Tong University School of Medicine, Shanghai, China

Accurate diagnosis remains a central challenge for medical large language models due to inherent knowledge limitations and hallucinations. While retrieval-augmented generation (RAG) and tool-augmented agentic methods show potential in mitigating these issues, their suboptimal utilization of external knowledge and the decoupling of the feedback-reasoning traceability, stemming from insufficient supervision, remain key limitations. To address these challenges, We introduce **Deep-DxSearch, an agentic RAG system trained end-to-end with reinforcement learning (RL) that enables steer traceable retrieval-augmented reasoning for medical diagnosis**. In Deep-DxSearch, we first construct a large-scale medical retrieval corpus comprising patient records and reliable medical knowledge sources to support retrieval-aware reasoning across diagnostic scenarios. **More crucially**, we frame the LLM as the core agent and the retrieval corpus as its environment, using tailored rewards on format, retrieval, reasoning structure, and diagnostic accuracy, thereby evolving the agentic RAG policy from large-scale data through RL.

Experiments demonstrate that our end-to-end agentic RL training framework consistently outperforms prompt-engineering and training-free RAG approaches across multiple data centers. After training, Deep-DxSearch achieves substantial gains in diagnostic accuracy, surpassing strong diagnostic baselines such as GPT-4o, DeepSeek-R1, and other medical-specific frameworks for both common and rare disease diagnosis under in-distribution (ID) and out-of-distribution (OOD) settings. Moreover, ablation studies on reward design and retrieval corpus components confirm their critical roles, underscoring the uniqueness and effectiveness of our approach compared with traditional implementations. Finally, case studies and interpretability analyses highlight improvements in Deep-DxSearch’s diagnostic policy, providing deeper insight into its performance gains and supporting clinicians in delivering more reliable and precise preliminary diagnoses. Data, code, and checkpoints are available at <https://github.com/MAGIC-AI4Med/Deep-DxSearch>.

1 INTRODUCTION

AI-driven medical diagnosis [1] presents unique challenges, as it must replicate the precision and context-awareness of clinical decision-making [2]. Such decision-making is inherently evidence-based, drawing on up-to-date guidelines, historical patient records, and structured medical knowledge to map presenting symptoms to plausible diseases [3, 4]. Recent LLM-based agentic retrieval-augmented generation (RAG) systems [5, 6, 7] have highlighted promising directions for building more powerful LLM-based diagnostic systems. By leveraging the orchestration capabilities of LLMs in conjunction with retrieval tools [8, 9], these systems can look up disease guidelines [10], search for related background knowledge [11], and, most critically for diagnosis, match similar diagnostic cases [12], ultimately synthesizing transparent and traceable diagnostic reasoning interwoven with retrieved evidence and analytical insights.

While promising, current agentic RAG system designs are typically inference-only and not trained end-to-end, which makes them fragile in high-stakes diagnostic environments where the agent may need to perform multiple retrievals [13] interleaved with evolving reasoning processes and former noisy retrieval feedback [14]. In particular, they exhibit **THREE** key limitations:

- **Rigid retrieval-reasoning interleaved workflow.** Inference-only designs [15, 16] lack joint optimization, leaving models unable to decide when tools or reasoning should be performed. This is especially restrictive in diagnostic settings, where reasoning, case matching, guideline lookup, and knowledge

† Corresponding author. Email addresses: {three-world, ya_zhang, weidi}@sjtu.edu.cn

searching must be interleaved with high freedom to allow continually evolving analytic focuses.

- **Heavy reliance on manually crafted query prompts.** These systems rely on extensive human priors to define retrieval query rules [17, 18, 19], yet in diagnostic settings universal heuristics are infeasible, since the focal symptoms and suspected diseases vary substantially across contexts.
- **Limited feedback-driven adaptation.** Statistic agentic workflows [20] cannot adjust generation in response to retrieval feedback. Unlike purely knowledge-based tasks, diagnostic reasoning must cope with noisy evidence such as complex clinical cases, posing significant challenges for agentic RAG systems.

Therefore, we propose **Deep-DxSearch**, an agentic RAG system specialized for medical diagnosis. Deep-DxSearch not only initializes the fundamental components of diagnostic agentic RAG systems—diverse retrieval tools, a comprehensive corpus, and clarified action spaces—but also introduces a **fully trainable reinforcement learning (RL)-based design**, enabling agents to jointly optimize interleaved retrieval-reasoning action policies end-to-end, enabling the emergence of retrieval-aware diagnostic reasoning.

We first curate, to our knowledge, the largest medical retrieval corpus to date (Fig. 1b, right), enabling to adapt agentic RAG in diagnostic settings. It integrates: (i) guideline-derived profiles for 1,500+ diseases with characteristic symptoms and phenotypes; (ii) 170,000+ structured patient cases from five public centers; and (iii) a large-scale knowledge collection with billions of curated entries from online medical resources and the scientific literature. Together, these sources provide diverse, multi-origin retrieval tools and evidences, thereby supporting Deep-DxSearch’s traceable diagnostic decisions.

More importantly, Deep-DxSearch’s agentic RAG policy is trained end-to-end, self-learned from large-scale data. Our LLM-based agent core operates via five action modes—**reason**, **lookup**, **match**, **search**, **diagnose**—to acquire evidence stepwise and reason transparently. We design a final reward scheme over four dimensions: output formatting, retrieval quality, analytical organization, and diagnostic accuracy to guide the agentic RAG system. This design learns optimal RAG trajectories, adapts the reasoning–retrieval policy, and balances decision quality against resource use while preserving traceability. In line with the famous “bitter lesson” in AI [21], we contend that, for agentic RAG design, scalable end-to-end training also outperforms hand-crafted heuristics, especially given diagnostic complexity and the lack of clear human priors.

We conduct a thorough evaluation (Fig. 1c,d) on both in-distribution (ID) and out-of-distribution (OOD) cross-center data. The ID benchmark includes 20,000+ diagnostic cases from six public datasets covering common and rare diseases. For OOD evaluation, we add 757 common-disease cases from a Bangla dataset (Mendeley) and 798 in-house cases from Xinhua Hospital. Across this diverse testbed, we reveal four key findings: **(i)** Our agentic RL training strategy significantly outperforms training-free agentic RAG designs, surpassing them by 9%/3% in ID/OOD evaluation in top-1 accuracy for common diseases, and by 13.5%/5% (ID/OOD) for rare diseases. **(ii)** The post-trained Deep-DxSearch surpasses general LLMs and medical systems (Fig. 1d) in a large margin, improving top-1 accuracy over medical foundation models by up to 19%/17% (ID/OOD) for common diseases and 24%/17% for rare diseases. **(iii)** Ablation studies highlight two key aspects: the effectiveness of our reward design and the contribution of our curated retrieval corpus. Our reward designed for co-optimization of retrieval and reasoning policies yields a 17% improvement in top-1 accuracy for common diseases and 22% for rare diseases, outperforming a target-only supervision scheme. **(iv)** Final interpretability analysis of the learned RAG policy further quantifies how agents evolve during training across three critical dimensions: retrieval relevance, differential diagnosis, and irrelevance exclusion.

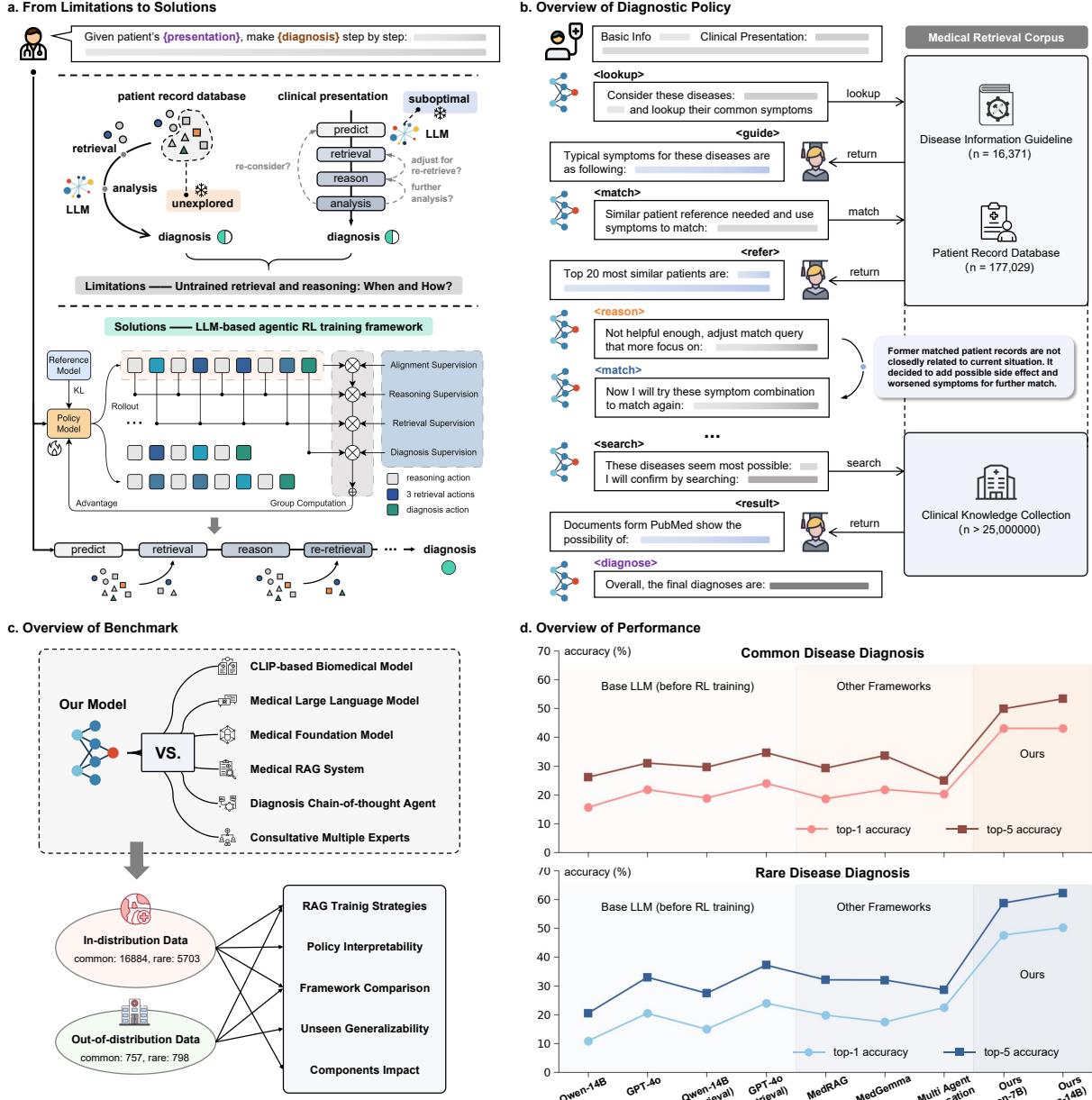


Figure 1 | Contribution Overview. **a.** Top: Limitations of existing medical foundation models in untrained retrieval and reasoning paradigms during inference. Bottom: Our method, which improves retrieval and reasoning through reinforcement learning. **b.** Left: Enhanced diagnostic workflow featuring deeper integration with the retrieval corpus and improved reasoning chains. Right: Structure of the medical retrieval corpus, including disease guidelines, patient records, and clinical knowledge collections. **c.** Top: Frameworks used for comparison. Bottom: Key evaluation metrics. **d.** Diagnostic performance for both common and rare diseases tasks, compared with baseline methods.

2 Problem Formulation

We formulate the agentic RAG system within a standard reinforcement learning (RL) framework, comprising two main components: (i) an LLM-based agent (\mathcal{M}_θ), and (ii) an external environment (\mathcal{E}) consisting of large-scale clinical corpora, including guidelines, knowledge bases, and patient case records. Details of workflow modeling can be found in Sec. 5.1.1.

Given a patient's clinical presentation (\mathcal{P})—including symptoms, medical history, and examination find-

ings—the agent functions as a sequential decision-making system. At each step, the agent first selects an action type from the finite set:

$$A = \{\langle \text{reason} \rangle, \langle \text{lookup} \rangle, \langle \text{match} \rangle, \langle \text{search} \rangle, \langle \text{diagnose} \rangle\}, \quad (1)$$

where the actions $\langle \text{reason} \rangle$ and $\langle \text{diagnose} \rangle$ represent the agent's internal analytical processes, corresponding to reasoning steps or final diagnostic decisions, and the other three denote interactions with external retrieval tools. Specifically, $\langle \text{lookup} \rangle$ accesses guidelines, $\langle \text{match} \rangle$ identifies similar cases, and $\langle \text{search} \rangle$ queries broader clinical knowledge sources.

After choosing an action type from A , the agent generates the corresponding textual specification τ . In the case of $\langle \text{reason} \rangle$ and $\langle \text{diagnose} \rangle$, τ represents the generated analytical content, while for the retrieval actions, it represents the corresponding search query. Thus, a complete action step is given by

$$a = (\alpha, \tau), \quad (2)$$

where $\alpha \in A$ denotes the selected action type and τ its textual specification.

Then the external clinical corpora environment (\mathcal{E}) will in response to the action, defined as:

$$f = \begin{cases} \mathcal{E}(\alpha, \tau), & \text{if } \alpha_i \in \{\langle \text{lookup} \rangle, \langle \text{match} \rangle, \langle \text{search} \rangle\} \\ \emptyset, & \text{if } \alpha_i \in \{\langle \text{reason} \rangle, \langle \text{diagnose} \rangle\} \end{cases}, \quad (3)$$

where f denotes the retrieval feedback.

Throughout the agentic RAG procedure, the agent executes a sequence of interleaved actions of various types ending with the $\langle \text{diagnose} \rangle$ action:

$$a_T = (\alpha_T, \tau_T) = \mathcal{M}_\theta(\mathcal{P}, a_1, f_1, \dots, a_{T-1}, f_{T-1}), \quad (4)$$

where T denotes the total steps and $\alpha_T = \langle \text{diagnose} \rangle$.

The ultimate objective is to optimize the entire agentic RAG action trajectory, covering both the selection of action types and the generation of their textual specifications, so as to achieve the best possible diagnostic outcome. Denoting the importance weight for each action step as λ_t , the final optimization objective is:

$$\max_{a_t} \sum_{t=1}^T \lambda_t \cdot Rwd_t(a_t, f_t), \quad (5)$$

where Rwd_t represents the reward function for each step.

Case Illustration. To further clarify the problem formulation, we illustrate an exact case about a 51-year-old man with a 15-year history of alcohol use disorder who presents with a series of symptoms and is ultimately confirmed as **acute myeloid leukemia**, collected from MIMIC-IV [22], accompanied by the traceable diagnostic reasoning generated by Deep-DxSearch:

Case Illustration: Diagnostic Process of A Patient with Confirmed Acute Myeloid Leukemia

Clinical Presentation Input: A 51-year-old male with 15-year alcohol use disorder presents with persistent thrombocytopenia (85,000/ μ L), thoracic aortic aneurysm (4.2 cm), sigmoid diverticulosis, refractory hypertension (160-180/95-105 mmHg), and new-onset atrial fibrillation. Physical examination reveals hepatomegaly, petechiae, and bleeding tendency with fatigue...

Traceable Diagnostic Reasoning Output of Deep-DxSearch:

<reason> Given alcoholism and thrombocytopenia, should first investigate alcohol-related conditions like cirrhosis and portal hypertension, as well as common hematologic causes... </reason>

<lookup> Alcoholic liver disease, Portal hypertension, Immune thrombocytopenic purpura </lookup>

<guide> **Alcoholic liver disease:** Hepatomegaly, portal hypertension, ascites, splenomegaly, varices... **Portal hypertension:** Splenomegaly, hypersplenism, thrombocytopenia, ascites... **ITP:** Isolated thrombocytopenia, mucocutaneous bleeding, normal spleen size... </guide>

<reason> Alcohol-related diseases lack ascites/splenomegaly, ITP doesn't explain multi-system involvement. Aortic aneurysm and atrial fibrillation pattern doesn't fit... </reason>

<match> Alcoholism, Aortic aneurysm, Bowel diverticulosis, Thrombocytopenia, Hypertension, Atrial fibrillation </match>

<refer> Retrieved cases: Primary CNS lymphoma (multiple cases), Primary myelofibrosis, Hepatocellular carcinoma, Giant cell arteritis... </refer>

<reason> Broad matching reveals hematologic malignancies, but results scattered. Should focus on thrombocytopenia triad to identify most relevant disorders... </reason>

<match> Thrombocytopenia, Hypertension, Atrial fibrillation </match>

<refer> **Highly relevant:** Acute myeloid leukemia (bone pain, atrial fibrillation, hypertension, thrombocytopenia), Chronic myelomonocytic leukemia, T-cell prolymphocytic leukemia... </refer>

<reason> Refined search points toward acute leukemias, particularly AML. Should verify typical presentations and explore alcohol-hematologic connections... </reason>

<lookup> Acute myeloid leukemia, Primary myelofibrosis, Chronic myelomonocytic leukemia </lookup>

<guide> **AML:** Thrombocytopenia, anemia, neutropenia, bleeding tendency, hepatomegaly, fatigue...

Primary myelofibrosis: Thrombocytopenia, splenomegaly, constitutional symptoms... </guide>

<search> [PMC] alcohol-induced thrombocytopenia mechanisms, secondary AML risk factors </search>

<result> Chronic alcohol consumption causes bone marrow suppression, increased therapy-related AML risk, cardiovascular complications... </result>

<diagnose> Acute myeloid leukemia, Primary myelofibrosis, Chronic myelomonocytic leukemia, T-cell prolymphocytic leukemia, Primary central nervous system lymphoma </diagnose>

3 RESULTS

In this section, we present our main results. We begin with key statistics of the medical retrieval corpus and datasets for training and evaluation, then assess diagnostic performance using Acc@1 and Acc@5 (details can be found in Sec. 5.4). We show that Deep-DxSearch, as an agentic RAG design, is more efficient than prior approaches across different LLM backbones. We then benchmark the best configuration (with Qwen2.5-14B) against state-of-the-art (SOTA) diagnostic baselines, followed by an ablation study and interpretability analysis illustrating how end-to-end RL shapes the agentic RAG system.

3.1 Data Statistics

This section summarizes the composition, statistics, and characteristics of the datasets used in this study. We first construct a comprehensive medical data resource to support retrieval, training, and evaluation, consisting of three major components: (1) a medical retrieval corpus, (2) a curated patient record database, and (3) a clinical knowledge collection. In addition, we assemble a dedicated training and evaluation dataset derived from multiple sources.

Medical Retrieval Corpus

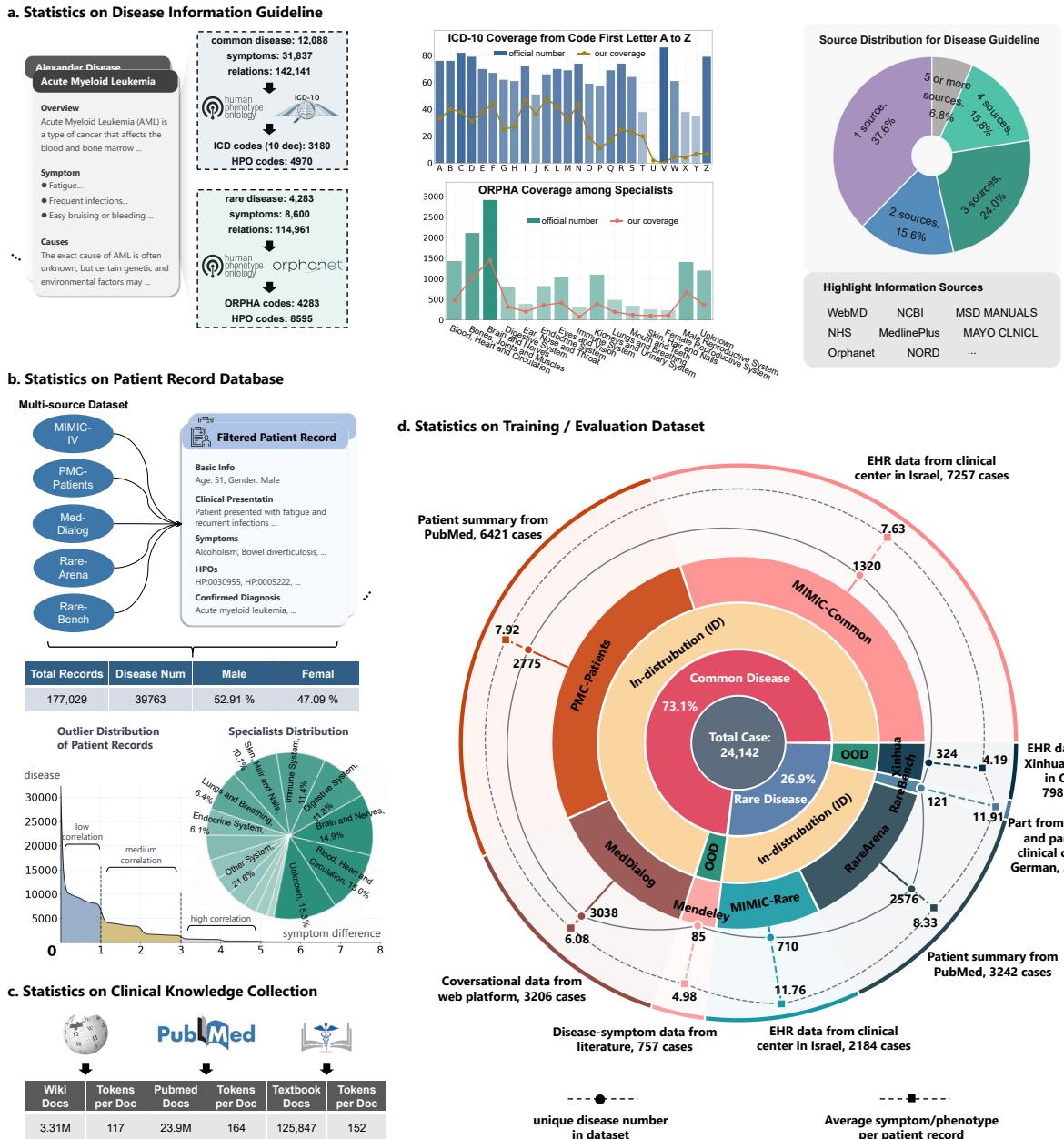


Figure 2 | Data statistics. **a.** Left: Overview of items and their relationships in the disease guideline. Middle: ICD coverage for common diseases and Orpha coverage for rare diseases. Right: Distribution of disease information sources, highlighting major public resources. **b.** Top: Summary statistics of patient records. Bottom: Distribution of outliers, illustrating discrepancies between real patient disease-symptom associations and guideline expectations; Breakdown of confirmed patient diagnoses by specialty. **c.** Summary statistics of the clinical knowledge collection. **d.** Detailed statistics of the seven-center datasets used for training and evaluation.

The retrieval corpus integrates diverse medical knowledge to mitigate coverage gaps and data imbalance, encompassing both common and rare diseases with large-scale, heterogeneous references. (i) *Disease Information Guideline*: As shown in Fig. 2a, data for 16,371 diseases—spanning common (ICD-10-CM¹) and rare (Orphanet²) conditions—are curated by extracting phenotype and symptom associations from literature and

¹<https://www.icd10data.com/ICD10CM/Codes>

²<https://www.orpha.net>

web sources. This yield 257,022 disease–phenotype/symptom pairs (142,141 for common and 114,881 for rare), mapped to ICD, ORPHA, and HPO³ terminologies. The dataset achieves complete coverage (100%) of ICD codes (to one decimal place) and 38.68% coverage of ORPHA codes, with over 50% of HPO terms included. Multi-source verification ensures data validity: each common-disease entry is supported by an average of 2.87 independent references, while rare-disease annotations are sourced from Orphanet. *(ii) Patient Record Database:* This subset comprises 177,029 curated patient records with validated diagnoses, clinical presentations, medication histories, and chief complaints. Phenotypes were extracted via automated and human-in-the-loop annotation (see Supplementary Materials). As shown in Fig. 2b, the disease distribution follows a long-tailed pattern across 14 major body systems [23]. Notably, significant discrepancies (Fig. 2b) exist between patient presentations and canonical diagnostic criteria, underscoring the complexity and diversity of real-world cases. *(iii) Clinical Knowledge Collection:* We further incorporated 3.31 million biomedical documents from Wikipedia⁴, 23.9 million PubMed⁵ articles, and 18 standard medical textbooks comprising 125,847 literature segments (Fig. 2c). Given the unstructured nature of these sources, a large language model was employed for summarization during training and inference to address input length constraints.

Training and Evaluation Dataset

We curated a total of 24,142 clinical cases, each containing a clinical presentation paired with a confirmed diagnosis, drawn from MIMIC [22], PMC-Patients [24], MedDialog [25], RareArena [26], RareBench [27], Mendeley [28], and Xinhua Hospital affiliated to Shanghai Jiao Tong University School of Medicine [29]. All raw data underwent strict quality control on case clarity, causality, and correctness (see Supplementary Materials) and were categorized into common and rare disease groups according to Orphanet coding system.

As shown in Fig. 2d, 73.1% of the dataset comprises common-disease cases, including MIMIC-C (7,257 cases), PMC-Patients (6,421 cases), MedDialog (3,206 cases), and Mendeley (757 cases), while the remaining 26.9% comprises rare-disease cases, including MIMIC-R (2,184 cases), RareArena (3,242 cases), RareBench (277 cases), and Xinhua-Rare (798 cases). The dataset contains 4–12 symptoms per case on average, with individual sources covering between 85 and over 3,000 distinct diseases. Geographically, cases originate from five countries or regions across America, Asia, and Europe.

For model development, we split the first five ID datasets by 3:1 to form the train and evaluation dataset, and the remaining two datasets, Mendeley and Xinhua Hospital, are all used for OOD evaluation.

3.2 Comparison on Agentic RAG System Designs

In this section, we present the effectiveness of our agentic RAG system design, incorporating the curated retrieval corpus and the trained RAG policy. Specifically, we benchmark Deep-DxSearch against (i) a vanilla model with direct inference and (ii) a prior training-free RAG method (detailed in Sec. 5.3) with access to the same retrieval corpus with the same base LLM. Our evaluation covers both ID (Tab. 1) and OOD datasets (Tab. 2), across diverse base LLM families and sizes—including Qwen2.5-7B, Llama3.1-8B, and Qwen2.5-14B—thereby demonstrating the robustness improvements achieved by our approach.

In-distribution Evaluation

In ID evaluation, we use six in-domain datasets, including MIMIC-C, PMC-Patients, MedDialog, MIMIC-R, RareArena and RareBench. We begin our analysis using Qwen2.5-14B as the shared base model:

Training-free RAG with our corpus vs. vanilla model with direct inference. We compare these two approaches to assess the effectiveness of our retrieval corpus. As shown in Tab. 1, integrating the corpus with training-free RAG consistently improves performance across all base models for both common- and rare-disease data centers. For example, with Qwen2.5-14B, top-1 accuracy in MedDialog (common disease) increases by 6.82% (from 17.87% to 24.69%), while in RareBench (rare disease) it rises by 16.63% (from 18.07% to 34.70%). These results confirm that extra knowledge injection is crucial for diagnosis and validate the effectiveness of our retrieval corpus design. Nonetheless, the relatively limited gains also indicate that simple corpus integration with engineered prompts is insufficient and requires further optimization.

³<https://hpo.jax.org/>

⁴<https://www.wikipedia.org/>

⁵<https://pubmed.ncbi.nlm.nih.gov/>

Table 1 | In-distribution evaluation of our agentic RL training vs. other strategies among varied backbone models.

Model	Common Disease Diagnosis						Rare Disease Diagnosis					
	MIMIC-C		PMC-Patients		MedDialog		MIMIC-R		RareArena		RareBench	
	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5
Qwen2.5-7B base												
Vanilla	4.23	8.61	10.96	24.45	8.98	15.60	9.63	18.77	6.06	12.80	17.86	30.48
RAG	9.13	12.62	20.08	32.25	19.59	31.86	17.82	25.17	8.28	13.02	31.98	57.70
Ours	33.09	42.87	41.41	46.80	49.28	55.34	52.44	61.53	25.97	35.32	64.47	79.51
Llama3.1-8B base												
Vanilla	3.81	8.00	15.13	25.08	7.32	16.49	3.40	12.75	7.94	12.87	20.89	35.71
RAG	9.01	13.03	25.77	40.55	20.04	28.70	15.03	22.10	11.31	17.02	33.27	55.01
Ours	21.05	27.83	34.15	45.74	35.51	46.92	42.00	55.02	22.41	29.95	64.33	73.86
Qwen2.5-14B base												
Vanilla	8.80	12.40	17.73	27.66	17.87	32.34	7.93	16.71	6.53	13.23	18.07	31.38
RAG	13.22	15.91	24.38	35.57	24.69	36.22	16.54	24.33	10.08	15.47	34.70	59.20
Ours	35.22	46.83	40.29	47.75	48.81	60.04	52.11	64.57	28.14	39.22	70.48	82.96

Agentic RL training vs. training-free retrieval-augmented (RAG) approach. We compare these two paradigms to evaluate how reinforcement learning with agentic supervision enhances the effective use of the retrieval corpus. As reported in Tab. 1, relative to the vanilla Qwen2.5-14B, our agentic RL approach yields substantial gains: in MedDialog, top-1 accuracy improves by 24.12% (24.69% \rightarrow 48.81%) and top-5 accuracy by 23.82% (36.22% \rightarrow 60.04%); in RareBench, top-1 accuracy improves by 35.78% (34.70% \rightarrow 70.48%) and top-5 accuracy by 23.76% (59.20% \rightarrow 82.96%). These results indicate that while direct retrieval-augmented prompting provides modest benefits over the base model, it remains limited in harnessing the full potential of the external corpus. By contrast, Deep-DxSearch achieves substantially higher performance through end-to-end policy optimization with agentic RL.

Full Deep-DxSearch vs. vanilla model with direct inference. Lastly, by directly comparing our method with vanilla LLMs, we demonstrate the overall effectiveness of Deep-DxSearch, including both the introduced retrieval component and the learned RAG policy. As shown in Tab. 1, Deep-DxSearch improves top-1 accuracy in common diseases by at least 23.56% (PMC-Patients, from 17.73% to 40.29%) and at most 30.94% (MedDialog, from 17.87% to 48.81%), and in rare diseases by at least 21.61% (RareArena, from 6.53% to 28.14%) and at most 52.41% (RareBench, from 18.07% to 70.48%). This comparison highlights two key observations: (i) the current LLMs though exhibits a degree of diagnostic ability, their performance remains insufficient; and (ii) the introduction of the retrieval corpus and the agentic RL training strategy can significantly enhance diagnostic accuracy.

Analysis across varied base models. Beyond Qwen2.5-14B, to demonstrate the generalization of our method, we also evaluate Deep-DxSearch on two more backbone models: Llama3.1-8B, and Qwen2.5-7B. As shown in Tab. 1, similar improvement patterns can be found on these LLMs, that our method consistently outperforms both vanilla and RAG approaches. For example, our approach improves the top-1 accuracy of Llama3.1-8B with RAG by 26.97% (from 15.03% to 42.00%) on MIMIC-R, and boosts Qwen2.5-7B by 34.62% (from 17.82% to 52.44%). Among the evaluated backbones, Qwen2.5-14B delivers the strongest overall performance, achieving the highest top-1 and top-5 accuracy on MIMIC-C, RareArena, and RareBench, as well as the best top-5 accuracy on PMC-Patients, MedDialog, and MIMIC-R, with only a minor exception where Qwen2.5-7B slightly surpasses it in top-1 accuracy. These findings underscore the clear superiority of

Table 2 | Out-of-distribution evaluation of our agentic RL training vs. other strategies with varied backbone models.

Model	Common Disease Diagnosis		Rare Disease Diagnosis	
	Mendeley		Xinhua Hosp.	
	Acc@1	Acc@5	Acc@1	Acc@5
Qwen2.5-7B base				
Vanilla	21.69	34.26	16.2	26.44
RAG	24.03	31.56	25.38	32.59
Ours	28.51	38.44	34.05	42.19
Llama3.1-8B base				
Vanilla	9.72	24.54	17.53	24.66
RAG	20.44	27.89	22.43	30.50
Ours	27.98	35.05	32.11	41.70
Qwen2.5-14B base				
Vanilla	22.22	34.61	20.01	27.20
RAG	26.59	34.01	27.62	36.85
Ours	31.09	42.7	35.13	45.77

agentic RL over alternative strategies and demonstrate its robustness. Based on the overall performance of these three candidates, we select Qwen2.5-14B as the backbone for subsequent experiments.

Out-of-distribution Evaluation

Beyond the ID setting, we also assess the effectiveness of our approach on OOD datasets. This evaluation confirms that Deep-DxSearch does not overfit to its training distribution but instead inherently learns a robust and generalizable retrieval-augmented diagnostic policy, substantially outperforming manually designed RAG strategies. Two **out-of-domain datasets** are adopted: the publicly available Bangla dataset Mendeley (common disease) and an in-house dataset from Xinhua Hospital (rare-disease). During the development of Deep-DxSearch, we rigorously ensure that no training data is sourced from the two centers, allowing their test cases to represent two entirely new practical case distributions.

As shown by the results in Tab. 2, we observe similar patterns as in the ID evaluation, which can be summarized into three key findings. **First**, compared with the vanilla Qwen2.5-14B, end-to-end agentic RL training with the retrieval corpus yields substantial gains, improving top-1 and top-5 accuracy by 8.87% (from 22.22% to 31.09%) and 8.09% (from 34.61% to 42.70%) in common-disease diagnosis, and by 15.12% (from 20.01% to 35.13%) and 18.57% (from 27.20 to 45.77%) in rare-disease diagnosis. **Second**, relative to the RAG baseline, Deep-DxSearch further enhances retrieval-augmented performance, with improvements of 4.50% (from 26.59% to 31.09%) (top-1) and 8.69% (from 34.01% to 42.70%) (top-5) in common diseases, and 7.51% (from 27.62% to 35.13%) (top-1) and 8.92% (from 36.85% to 45.77%) (top-5) in rare diseases. **Third**, the benefits of training extend across different backbones, with consistent improvements in both top-1 and top-5 accuracy for common and rare disease diagnosis across all three backbone models.

Together, these results underscore the efficacy of our approach, which consistently surpasses alternative prompting or train-free RAG method, adapts effectively to different backbones, and enables a more reliable, generalizable, and robust diagnostic workflow.

3.3 Comparison with Other Diagnostic SOTAs

In this section, we treat Deep-DxSearch employing Qwen2.5-14B as backbone as a complete diagnostic system, rather than viewing it as a specific RAG algorithm, and compare it directly against other diagnostic SOTAs. We benchmark Deep-DxSearch against a suite of strong baselines, including general-purpose LLMs prompted for diagnosis and other SOTA medical diagnosis-aligned methods, under both common and rare disease conditions and across in-distribution and out-of-distribution evaluation settings.

In-distribution Evaluation

Six datasets included here are the same in-domain datasets mentioned before.

Deep-DxSearch vs. general-purpose LLMs prompted for diagnosis. This evaluation assesses whether our method is competitive in routine clinical practice, given that such general-purpose models are already being used in hospital settings. Deep-DxSearch outperforms GPT-4o [30] and DeepSeek-R1 [31] on both common- and rare-disease diagnosis tasks (Fig. 3a). For common diseases, Deep-DxSearch achieves 43.04% top-1 accuracy and 53.30% top-5 accuracy, surpassing the next-best general model, DeepSeek-R1 (23.07% and 34.76%), by 19.97% and 17.54% respectively. For rare diseases, Deep-DxSearch reaches 49.25% top-1 accuracy and 61.02% top-5 accuracy, representing gains of 29.68% and 24.47% over DeepSeek-R1 (19.57% and 36.65%). Compared with GPT-4o augmented using the retrieval corpus, Deep-DxSearch delivers additional improvements of 19.07% in top-1 accuracy (23.97% → 43.04%) and 23.62% in top-1 accuracy for rare diseases (25.63% → 49.25%). These results underscore the value of trained medical agentic RAG systems particularly for low-prevalence conditions where prior knowledge integration and careful evidence synthesis are essential.

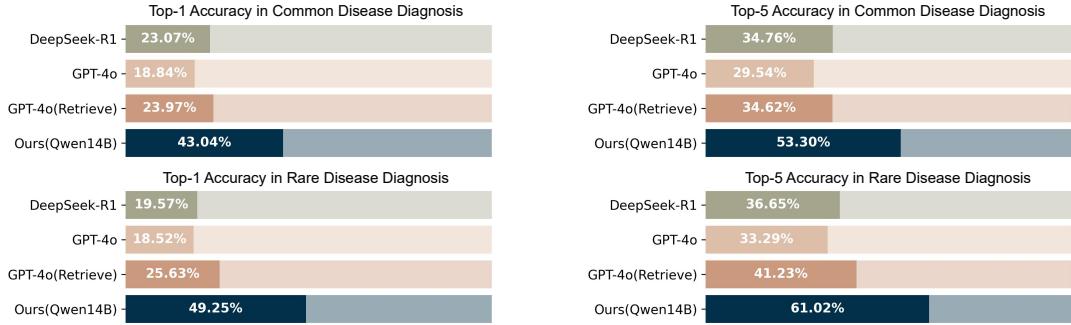
Deep-DxSearch vs. medical diagnosis-aligned methods. We compare our approach with other models enhanced with medical domain knowledge to evaluate whether our method achieves SOTAs. Against these specialized medical diagnosis systems—including MedCPT (medical CLIP-based model)[32], Baichuan-M1 (medical LLM)[33], MedGemma (medical foundation model)[34], MedRAG (medical RAG system)[35], CoD (diagnostic chain-of-thought agent)[36], and MAC (medical multi-agent consultative system)[37]—Deep-DxSearch achieves the strongest overall performance (Fig. 3b). On common-disease datasets, it exceeds the second-highest top-1 accuracy, achieved by Baichuan-M1, by 19.91%, and the second-highest top-5 accuracy, achieved by MedGemma, by 19.70%. On rare-disease datasets, it outperforms the second-highest top-1 accuracy from MAC by 23.68% and the second-highest top-5 accuracy from MedRAG by 23.72%. Deep-DxSearch achieves superior accuracy across most common-disease data centers and all rare-disease data centers (Fig. 3c), with a single exception on MedDialog, where CoD performs slightly better. This is because MedDialog was specifically optimized for CoD without incorporating other datasets. Overall, these findings indicate that although existing medical alignment approaches attempt to incorporate domain knowledge, clinical priors, or specialized reasoning to enhance diagnostic accuracy, their robustness and generalizability, especially for rare conditions, remain limited. In contrast, Deep-DxSearch’s co-optimized retrieval-and-reasoning framework achieves markedly stronger diagnostic performance.

Out-of-distribution Evaluation

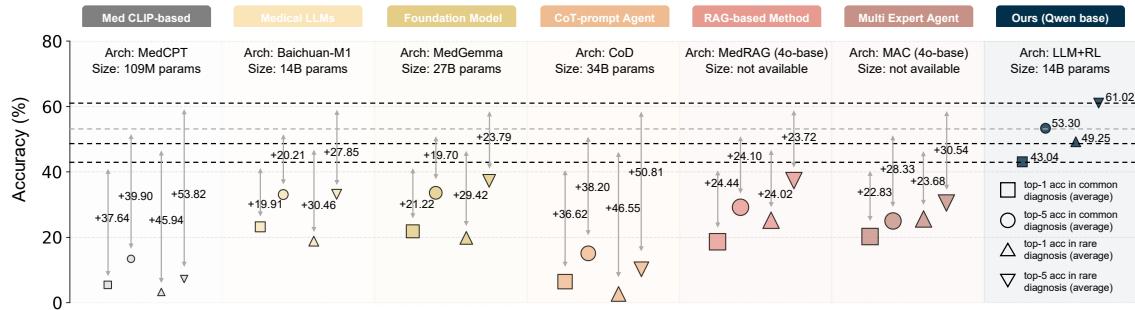
To evaluate whether Deep-DxSearch demonstrates competitive generalizability under unseen conditions relative to other methods—which provides stronger evidence of technological superiority and is essential for real-world deployment—we additionally conduct out-of-distribution (OOD) experiments (Tab. 3). With same benchmarking data as Sec. 3.2, we compare our Deep-DxSearch with general-purpose LLM DeepSeek-R1, prompted for diagnosis and medical-specific methods including MedCPT, Baichuan-M1, MedGemma, CoD, MedRAG and MAC. Here we do not include GPT-4o due to privacy concern on data from Xinhua hospital.

As shown in Tab. 3, Deep-DxSearch achieves the highest top-1 and top-5 accuracy while using the smallest model size (14B), on both the common-disease dataset from Mendeley and the rare-disease dataset from Xinhua. In the common-disease setting, Deep-DxSearch surpasses the second-best results achieved by MedRAG by 10.12% in top-1 accuracy (41.20% → 51.32%) and 12.51% in top-5 accuracy (56.02% → 68.53%). In the rare-disease setting, it outperforms the next-best top-1 accuracy achieved by MAC by 0.10% (45.06% → 45.16%) and the top-5 accuracy achieved by MedRAG by 7.62% (54.20% → 61.82%). It is worth noting that although MedCPT shows reasonable performance in the rare-disease setting (27.60% top-1 accuracy and 40.08% top-5 accuracy), it performs poorly on Mendeley, likely because the data distribution differs substantially from the

a. In-distribution comparison with general-purpose LLMs prompted for diagnosis in Average



b. In-distribution comparison with other medical diagnosis alignment methods in Average



c. In-distribution comparison with other medical diagnosis alignment methods across 6 data centers

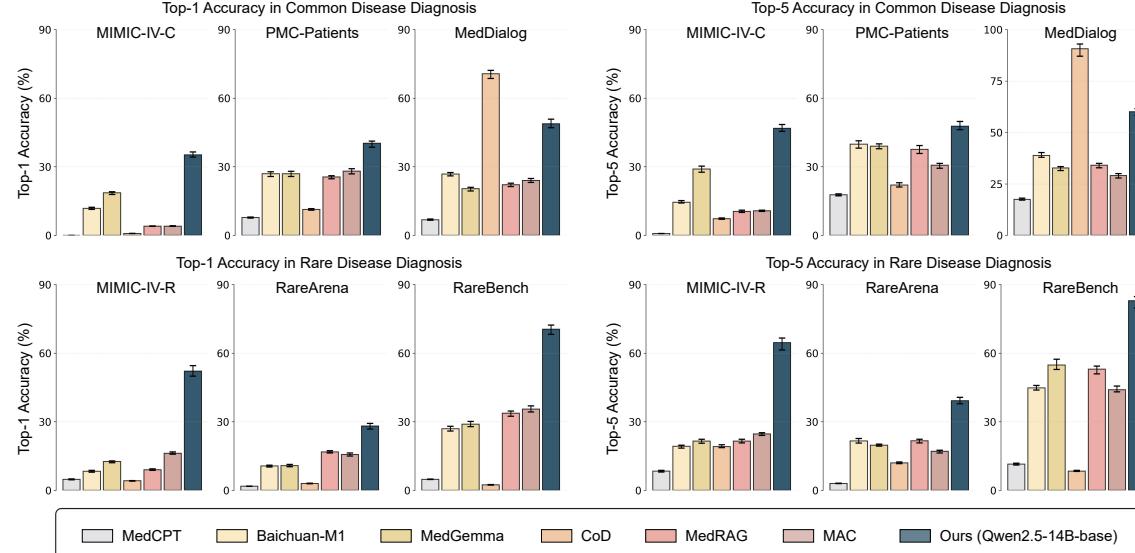


Figure 3 | In-distribution comparison. **a.** Comparison of our model’s diagnostic performance with other baseline LLMs on common (average) and rare (average) disease diagnosis, including GPT-4o, GPT-4o with direct retrieval, and the reasoning model DeepSeek-R1. **b.** Overall diagnostic accuracy across representative frameworks, with the size of each geometric shape indicating the number of model parameters. **c.** Detailed diagnostic results of our model and all evaluated frameworks on each data center.

categories used in its training data.

These results demonstrate that the diagnostic workflow learned by Deep-DxSearch generalizes effectively to unseen datasets and clinical scenarios, highlighting its consistent better adaptability and robustness compared to other SOTA methods under out-of-distribution conditions.

Table 3 | Generalizability evaluation. We compared Deep-DxSearch with one general-purpose LLM and 6 medical-specific methods on two out-of-distribution dataset (one for common and one for rare).

Method	Category	Size	Year	Mendeley-Common (Pubic)		Xinhua-Rare (In-house)	
				Acc@1	Acc@5	Acc@1	Acc@5
General-purpose LLMs prompted for diagnosis							
DeepSeek-R1	Reasoning LLM	671B	2025	30.55	41.20	37.52	49.63
Medical-specific methods aligned with diagnosis							
MedCPT	Biomed CLIP-base Model	109M	2023	3.24	5.02	27.60	40.08
Baichuan-M1	Medical LLM	14B	2025	28.70	41.85	40.80	48.17
MedGemma	Medical Foundation Model	27B	2025	34.26	47.33	28.01	42.16
CoD	Chain-of-thought Agent	34B	2024	14.35	29.17	19.00	27.80
MedRAG	RAG-based Method	-	2024	41.20	56.02	39.63	54.20
MAC	Multi-agent System	-	2025	36.11	50.93	45.06	51.42
Ours (Qwen2.5-14B backbone)							
Deep-DxSearch	Agentic RL	14B	2025	51.32	68.53	45.16	61.82

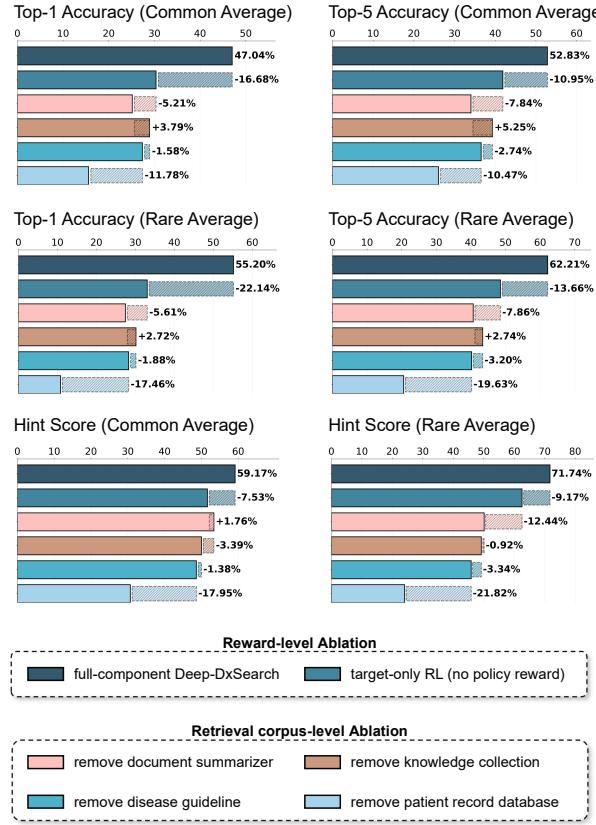
3.4 Ablation Studies

In this section, we present ablation studies at two levels: the reward components design of agentic RL and the components of the retrieval corpus.

Ablation studies on reward design. In addition to the basic reward based on final diagnostic accuracy, we further design three auxiliary components—format reward, patient-matching reward, and searching reward—which together form the policy reward. These components guide retrieval-and-reasoning optimization while jointly supervising the final diagnostic outcome. To assess the effectiveness, we first disable the policy reward, resulting in a target-only RL setting. We find this basic configuration leads to a rigid diagnostic trajectory after training—preliminary diagnosis → disease knowledge retrieval → case matching → final diagnosis—and reduces flexibility. Consequently, as shown in Fig. 4a, average top-1 accuracy decreases by 16.68% in common-disease diagnosis and 22.14% in rare-disease diagnosis. We further assess the “Hint” metric, which measures whether the correct disease is considered during reasoning even when the final prediction is incorrect. Under target-only fine-tuning, this metric drops by 7.53% for common diseases and 9.17% for rare diseases. Collectively, these findings demonstrate the clear advantage of end-to-end agentic RL over target-only training, underscoring the importance of flexible reasoning and the joint optimization of intermediate diagnostic steps alongside final conclusions.

Ablation studies on retrieval corpus. Then, We conduct a step-by-step ablation, progressively removing components from the full-component retrieval environment down to a non-environment direct-diagnosis setting during training, to evaluate the impact of each module on final performance. **Note:** all reported performance changes are relative to the preceding ablation step. As shown in Fig. 4a, (i) removing the document-summarization module and feeding raw retrieved content into the context causes a 5.21% drop in top-1 accuracy for common diseases and a 5.61% drop for rare diseases compared with the full-component setting, reflecting input-length constraints and noise amplification without targeted distillation; (ii) excluding the clinical-knowledge collection leads to a smaller reduction in accuracy relative to the full-component setting, with top-1 accuracy still 3.79% (common) and 2.72% (rare) higher than in the no-summarization setting—suggesting that when summarization is absent, a smaller context can partially mitigate noise but at the cost of coverage; (iii) removing the disease-guideline resource produces an additional decline of 1.58% (common) and 1.88% (rare) in top-1 accuracy, indicating a supportive yet secondary role in structuring

a. Ablation study on components impact



b. Results of interpretability quantification

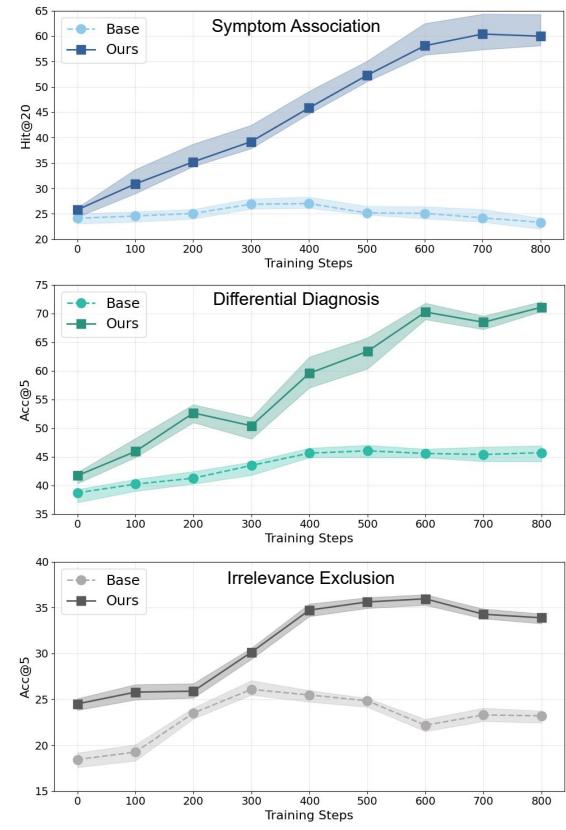


Figure 4 | Ablation study and Interpretability analysis. **a.** Performance variation from full-component RL training to target-only reward supervision and further retrieval environment-wise step-by-step ablation to the vanilla model. “Hint” indicates the correct disease is at least considered during diagnostic reasoning. **b.** Diagnostic Policy interpretability evaluated by: symptom association during similar patient retrieval, differential diagnosis among candidate diseases, and exclusion of irrelevant information during reasoning when retrieval is misleading. “Base” denotes the target-only RL without intermediate reward supervision; “Hit” indicates that retrieved patient cases are helpful for diagnosis.

reasoning; and (iv) excluding similar-case retrieval results in a substantial drop of 11.78% (common) and 17.46% (rare) in top-1 accuracy, underscoring the strong contribution of similar case evidence to diagnostic accuracy. **Overall**, all components contribute meaningfully to performance, with patient-record retrieval emerging most critical, while summarization and clinical guidelines provide important complementary gains.

3.5 Interpretability Analysis of the Learned RAG Policy

Accurate diagnosis hinges not only on the final label prediction but also on the sufficiency, relevance, and reliability of the underlying evidence. An efficient diagnostic RAG policy should therefore demonstrate three core capabilities: (i) The ability to synthesize observed symptoms, organizing the most important and suitable queries, and retrieve current relevant prior knowledge or cases, (ii) The capacity to discriminate among competing diagnostic hypotheses and to clarify further retrieval or analytical directions, and (iii) the robustness to resist misleading or irrelevant return information. Together, these aspects reveal how effectively a system balances evidence gathering with reasoning, thereby providing insight into its retrieval-augmented generation dynamics.

The case study in Sec. 2 highlights the role of a structured diagnostic workflow in achieving accurate outcomes. To examine this effect quantitatively, we analyzed how Deep-DxSearch’s diagnostic RAG policy evolved over the course of training. Specifically, we compared Deep-DxSearch with the target-only agentic RL trained solely

on final diagnostic labels, without intermediate policy supervision. This comparison allows us to evaluate the reward function’s impact on diagnostic accuracy and process transparency, thereby exposing what the model has learned through reinforcement learning. To quantify these intermediate abilities, we design and analyze with the following metrics:

- **Symptom Association** (for retrieval adaptation): This metric evaluates the model’s ability to link both explicit and related symptoms—possibly occurring before, after, or alongside the main complaint—to relevant reference cases. As shown in Fig. 4b (top), we measure this using *hit@20*, defined as the proportion of times that at least one of the top 20 retrieved cases shares the same diagnosis as the ground-truth case. Compared with the target-only baseline (which shows only minor improvement), Deep-DxSearch achieves a substantial increase in *hit@20*, from 25.79% to 60.39%.
- **Differential Diagnosis**: We evaluate the model’s ability to identify correct diagnosis from a set of candidates using top-5 accuracy, defined as the proportion of cases in which the ground-truth disease appears among the model’s five most confident predictions. While the baseline improved from 38.71% to 45.00%, Deep-DxSearch achieve a substantial gain of nearly 30 percentage points, reaching 71.07% (Fig. 4b, middle).
- **Irrelevance Exclusion**: To assess robustness, we inject misleading reference materials into the retrieval process by returning irrelevant guidelines, patient records, and medical documents when querying the corpus. Even under this setting, Deep-DxSearch’s top-5 accuracy increased by nearly 10% (Fig. 4b, bottom), whereas baseline methods showed gains of only about 5% over the course of training, highlighting our model’s enhanced ability to filter out irrelevant information.

Our findings reveal the agentic RAG policy improvements in Deep-DxSearch’s RL training in three core aspects: (i) **adaptive retrieval strategy** — the model increasingly refined its ability to retrieve diagnostically relevant patient cases; (ii) **differential diagnosis** — the model became more effective at distinguishing the correct diagnosis from among plausible alternatives; (iii) **irrelevance exclusion** — the model improved in filtering out misleading or unrelated information during the diagnostic process. These advances show that Deep-DxSearch develops a structured and effective diagnostic workflow, with enhanced retrieval, reasoning, and robustness contributing to its superior performance.

4 DISCUSSION

Diagnosis remains a central challenge in clinical medicine, especially in complex or rare conditions. Although large language models (LLMs) can support diagnostic reasoning, their performance is constrained by static knowledge, hallucinations, and inference under uncertainty [38, 39, 40]. Retrieval-augmented or tool-augmented agentic methods (Agentic RAG systems) potentially mitigate some issues, but most approaches under-emphasize multi-turn query adjustment, to adapt to the long-tailed distribution of medical corpora [41, 42] and substantial clinical noise [43, 44, 45]. Furthermore, seemingly minor changes in prompting in current training-free agentic RAG design can lead to markedly different retrieval outcomes, yet such failure modes are rarely addressed in practice. Current LLMs are not explicitly trained to integrate multi-turn or multi-source evidence [46], resulting in suboptimal diagnostic retrieval-reasoning trajectories [47].

We present Deep-DxSearch, an agentic RAG system for diagnosis that unifies evidence acquisition with clinical reasoning via reinforcement learning. Rather than passively consuming retrieved content, Deep-DxSearch learns to control the evidence-gathering process: it formulates and adapts queries, modulates retrieval depth and sources based on uncertainty and feedback, and filters distractors. This agentic control improves robustness in data-sparse or noisy settings and yields decisions that are more accurate and contextually grounded. Our contributions are threefold: (i) a large-scale, heterogeneous clinical corpus spanning longitudinal patient records, structured guidelines, and up-to-date clinical knowledge to initially support the agentic RAG system for traceable diagnosis reasoning; (ii) more importantly, a soft-reward RL framework with trajectory-level credit assignment that jointly optimizes agentic RAG policy and reasoning over multi-turn interactions; and (iii) a comprehensive, multi-center evaluation against strong general-purpose LLMs and representative diagnostic systems, demonstrating consistent gains in accuracy, reliability.

Technically, current training of medical LLMs relies heavily on human-curated data, artificially constructed

instructions, and human-led supervision. A major obstacle to further progress is the strong dependence on training paradigms shaped by human priors. This limitation is particularly pronounced in complex clinical scenarios, where such priors are not necessarily statistically optimal. Our diagnostic multi-turn RAG scenario exemplifies this challenge: as retrieval feedback accumulates and reasoning conditions evolve, obtaining well-annotated supervision to guide models’ next-step action becomes difficult, since even human priors fail to define the optimal solution. Consequently, most current agentic RAG systems are designed in an inference-only manner, relying on LLMs’ inherent tool-use abilities or carefully crafted prompts and workflows. However, as “The Bitter Lesson” [21] emphasizes, human knowledge and skills may offer short-term benefits but often become obsolete, while the enduring advantage lies in exploiting statistical regularities from large-scale data. By analogy, linking LLMs with diverse retrieval tools to construct an agentic RAG system for diagnosis through handcrafted prompt engineering also constrains the model’s capacity to explore and develop truly effective orchestration policies across diagnostic paradigms and clinical knowledge-seeking that are deeply aligned with retrieval tool complexity strategies. We therefore argue that, compared with prior statistical agentic RAG designs, our RL-based approach—which combines verifiable key-outcome rewards with greater generative freedom for exploration and search—offers a more promising path toward improving agentic RAG systems for traceable diagnostic reasoning. Our experiments provide encouraging evidence in support of this.

Our analyses demonstrate that: compared with the training-free RAG approach, our end-to-end training shows significant improvement over in-context learning with retrieval feedback, underscoring the limited optimization achievable through pure prompt-engineering methods. Compared with target-only reward supervision, our agentic RL training approach delivers superior diagnostic performance due to the co-supervision of reasoning and retrieval policies, highlighting the effectiveness of our reward design. Through this tailored training method, Deep-DxSearch achieves state-of-the-art accuracy in both common- and rare-disease diagnosis, outperforming stronger baseline LLMs—including the 671B DeepSeek and the proprietary GPT-4o, which have many times more parameters—as well as a range of competing diagnostic systems such as medical foundation models, diagnostic agents, and multi-expert consultation systems. Beyond these in-distribution comparisons, we conduct out-of-distribution experiments under zero-shot settings for both common and rare diseases, demonstrating Deep-DxSearch’s comprehensive generalization capability, surpassing all other competitors. Further interpretability studies reveal that Deep-DxSearch progressively improves its diagnostic policy during training, showing enhanced ability to: (1) associate key symptoms for more accurate knowledge retrieval, (2) identify the most probable diagnosis from a candidate list, and (3) exclude irrelevant or misleading information, thereby improving robustness. Thus, we conclude the key findings of Deep-DxSearch as follows:

- Superior diagnostic accuracy through tailored agentic RL training.
- Consistent advantages under both ID and OOD Evaluation over competitors.
- Large margin enhancements in RAG policy toward more reliable traceable diagnoses.

Our findings suggest a path forward for medical foundation models: external knowledge acquisition and reasoning should be co-optimized, with query formulation treated as a first-class learning objective rather than an afterthought of prompt engineering. More broadly, agentic control over information gathering may benefit other safety-critical domains where evidence is fragmented, noisy, and long-tailed.

Limitation and Future Direction

While Deep-DxSearch demonstrates superior decision-making and diagnostic accuracy, several limitations remain. **First**, although we comprehensively compare our approach with baseline designs and SOTA frameworks, its impact on supporting clinicians in real-time diagnostic settings has not yet been evaluated. Clinical validation will be essential in future work to establish the practical effectiveness and collaborative potential of Deep-DxSearch in deployment. **Second**, although our retrieval corpus is among the most comprehensive in current research, customization to specific clinical centers is limited, which may restrict the framework’s ability to fully capture local clinical contexts. Future efforts will focus on facilitating broader adoption and precise adaptation to diverse clinical environments. **Third**, our evaluation is confined to diagnostic tasks; the applicability of our approach to other key medical domains—such as treatment planning and patient follow-up—remains untested. Expanding the framework to encompass a wider range of medical tasks and developing complementary tools beyond retrieval-based reasoning will be important future directions.

5 METHODS

This section firstly details the architecture and policy objectives of the proposed Deep-DxSearch framework. Then, we introduce the agentic RL training implementations. Finally, we outline the evaluation protocol and metrics used to assess diagnostic performance.

5.1 System Design

Here, we introduce **Deep-DxSearch**, an agentic reinforcement learning framework that governs a retrieval-augmented diagnostic pipeline. Retrieval augmentation supplies access to an external clinical corpus; the agentic policy learns when and how to use that access—what to query, whether to reformulate, which sources to trust, how to integrate evidence, and when to commit to a diagnosis. We formalize the workflow as a finite set of five action types spanning evidence acquisition and reasoning. Policy learning is aligned with task objectives via soft rewards that jointly supervise retrieval quality, evidence integration, and diagnostic correctness. We then detail the training strategies used to learn a stable, sample-efficient policy.

5.1.1 Main Workflow Formulation

We formulate diagnostic reasoning as a partially observable agent–environment interaction. Specifically, the LLM-based agent is modeled as a policy \mathcal{M}_θ , and the environment is represented by the retrieval corpus \mathcal{E} . Given a patient presentation \mathcal{P} and an initial state \mathcal{S}_0 (comprising system instructions and \mathcal{P}), the diagnostic trajectory—including intermediate reasoning and final diagnosis—is generated as follows:

$$d \sim \mathcal{M}_\theta(\cdot | \mathcal{S}_0) \circ (\mathcal{E}, \mathcal{A}), \quad (6)$$

We introduce the key reinforcement learning (RL) concepts underpinning our formulation:

- **Agent (\mathcal{M}_θ)**: The policy model parameterized by θ , responsible for stepwise decision-making throughout the diagnostic workflow.
- **State (\mathcal{S})**: The state \mathcal{S}_i encodes the sequence of prior actions and the accumulated contextual information up to step i .
- **Environment (\mathcal{E})**: The retrieval corpus, which comprises structured knowledge from disease guidelines, patient records, and general medical literature.
- **Action (\mathcal{A})**: The atomic operations that advance the diagnostic process. Actions may either drive agent–environment interactions or internal reasoning, as detailed in “Action Space” below.
- **Specification (\mathcal{T})**: Information obtained after each action, either generated by the agent or returned from the environment. We denote the trajectory of observations as $\mathcal{T}_n = \{\tau_1, \tau_2, \dots, \tau_n\}$.

Action Space. We distinguish between *active* agent actions (which control reasoning and retrieval) and *passive* environment actions (which return evidence from the corpus). The five active actions are:

- $\langle \text{reason} \rangle$: Integrate current evidence, update hypotheses, and determine next steps.
- $\langle \text{lookup} \rangle$: Query disease-specific knowledge using candidate diagnoses; the corpus returns content delimited by $\langle \text{guide} \rangle$.
- $\langle \text{match} \rangle$: Retrieve similar patient records based on symptom lists; the corpus returns references delimited by $\langle \text{refer} \rangle$.
- $\langle \text{search} \rangle$: Issue free-text queries for general medical knowledge (e.g., symptom–disease relations); results are delimited by $\langle \text{result} \rangle$.
- $\langle \text{diagnose} \rangle$: Terminal action, committing to a final diagnosis.

Passive actions correspond to corpus returns: $\{\langle \text{guide} \rangle, \langle \text{refer} \rangle, \langle \text{result} \rangle\}$, and are formally defined as:

$$\mathcal{A}^{\text{act}} = \{\langle \text{reason} \rangle, \langle \text{lookup} \rangle, \langle \text{match} \rangle, \langle \text{search} \rangle, \langle \text{diagnose} \rangle\}, \quad \mathcal{A}^{\text{pas}} = \{\langle \text{guide} \rangle, \langle \text{refer} \rangle, \langle \text{result} \rangle\}. \quad (7)$$

This framework enables iterative control of the diagnostic workflow via trajectory management and context updates.

Trajectory and Control. Let the action trajectory at step i be $\mathcal{A}_i = \{\alpha_0, \alpha_1, \dots, \alpha_i\}$, with corresponding specifications $\mathcal{T}_i = \{\tau_1, \tau_2, \dots, \tau_i\}$. The system state is then defined as:

$$\mathcal{S}_i = \{\mathcal{S}_0, \mathcal{A}_i, \mathcal{T}_i\}, \quad (8)$$

At each step i , the next action α_{i+1} is determined by:

$$\alpha_{i+1} = \begin{cases} \phi(\alpha_i), & \text{if } \alpha_i \in \{\langle \text{lookup} \rangle, \langle \text{match} \rangle, \langle \text{search} \rangle\}, \\ \mathcal{M}_\theta(\mathcal{S}_i), & \text{otherwise,} \end{cases} \quad (9)$$

where $\phi : \mathcal{A}^{\text{act}} \rightarrow \mathcal{A}^{\text{pas}}$ deterministically maps retrieval actions to their corresponding passive responses:

$$\phi(\alpha_i) = \begin{cases} \langle \text{guide} \rangle, & \text{if } \alpha_i = \langle \text{lookup} \rangle, \\ \langle \text{refer} \rangle, & \text{if } \alpha_i = \langle \text{match} \rangle, \\ \langle \text{result} \rangle, & \text{if } \alpha_i = \langle \text{search} \rangle. \end{cases} \quad (10)$$

Context Updates. Each observation is produced immediately following an action, either by the policy model or by the environment. Formally,

$$\tau_{i+1} = \begin{cases} \mathcal{M}_\theta(\alpha_{i+1}, \mathcal{S}_i), & \text{if } \alpha_{i+1} \in \mathcal{A}^{\text{act}}, \\ \mathcal{E}(\alpha_{i+1}, \tau_i), & \text{if } \alpha_{i+1} \in \mathcal{A}^{\text{pas}}. \end{cases} \quad (11)$$

The updated state is then:

$$\mathcal{S}_{i+1} = \mathcal{S}_i \cup (\alpha_{i+1}, \tau_{i+1}), \quad (12)$$

This formalism captures the iterative and context-dependent nature of diagnostic reasoning, decoupling agent-driven decisions from environment-provided evidence.

Termination and output. The episode terminates when the agent determines that sufficient reasoning and evidence have been obtained, and issues the terminal action $\langle \text{diagnose} \rangle$. The system then outputs the final diagnosis o_n , corresponding to the observation generated at step n when $a_n = \langle \text{diagnose} \rangle$.

5.1.2 Reward Design

The training objective is to enhance Deep-DxSearch to provide a more transparent diagnostic workflow, with improved retrieval and reasoning strategies for more accurate diagnoses, while balancing inference costs. To achieve this, we design specialized reward mechanisms and training losses tailored for evidence-based optimization.

Format coefficient. We firstly focus on the format reward coefficient σ_f because the strictly following to appropriate format is always the preliminary of correct task instruction-following and final performance evaluation. The format reward coefficient σ_f acts as a strict gatekeeper for subsequent evaluation, ensuring that model outputs conform exactly to the required structural template. This coefficient is defined as:

$$\sigma_f = \begin{cases} 0, & \text{if any required format rule is violated;} \\ 1, & \text{if all format constraints are strictly satisfied,} \end{cases} \quad (13)$$

Specifically, σ_f is set to zero ($\sigma_f = 0$) in the presence of *any* but not limited to the following violations:

- **Missing diagnosis tags:** Output does not contain exactly one paired $\langle \text{diagnose} \rangle \langle / \text{diagnose} \rangle$ tag.

- **Improper tag order:** The `<diagnose>` tag appears after the `</diagnose>` tag.
- **Omission of required formatting:** The content within `<diagnose>...</diagnose>` does not include at least one disease name formatted as `\textbf{}`.
- **Excessive match tags:** More than maximum `<match>...</match>` tags are present.
- **Unmatched or incomplete tags:** The number of `<search>` tags does not equal the number of `</search>` tags, or any tag is left unclosed.
- **Malformed iteration structure:** If a `<match>` tag is present, it must be immediately followed by a `<refer>...</refer>` block in the correct order; any deviation from this pattern is invalid.

In all such cases, $\sigma_f = 0$ and the output receives zero reward, regardless of downstream content correctness. Only outputs that meet *every* structural and formatting requirement (i.e., $\sigma_f = 1$) qualify for further evaluation, ensuring strict and consistent adherence to the task specification.

Patient matching reward. One of our most important training targets is to enhance the agent’s ability to iteratively adjust phenotype or symptom queries for diverse and precise matching with similar patient cases. We expect the agent to learn strategies such as adding phenotypes commonly observed in suspected disease categories, replacing terms with alternative medical vocabulary, incorporating potential complications or associated features, and considering manifestations from different disease stages. These adjustments help the agent to explore key information and features for improved matching. The match reward Rwd_M is thus designed to balance the incentive for exploration diversity and the penalty for excessive or redundant match operations. Specifically, a reward of $+0.5$ is granted if any of the diseases returned in any `<refer>` block matches the ground truth diagnosis, while each use of the `<match>` operation incurs a penalty of 0.1 , up to a maximum of -0.3 . If multiple matches are performed, we require sufficient diversity in phenotype sets between consecutive matches (at least two phenotypes must change); failure to meet this constraint sets Rwd_M to zero. If the format or structure of the match/refer/think blocks is violated, Rwd_M is not computed and the total reward is zeroed by the format coefficient. Formally,

$$R_M = \begin{cases} 0.5 - \min(0.1 n_{\text{match}}, 0.3), & \text{if at least one reference matches the ground truth,} \\ -\min(0.1 n_{\text{match}}, 0.3), & \text{otherwise,} \end{cases} \quad (14)$$

where n_{match} is the number of match operations used.

Searching reward. The `search reward` Rwd_S evaluates how well the diseases listed in the `<search>` blocks align with the ground truth diagnosis at the token level, thereby encouraging the model to propose correct or relevant candidates that may facilitate the final answer. If the number of diseases returned exceeds a predefined maximum max_n , or if the number of `<search>` tags does not match the number of `<result>` tags, the search reward is set to zero. Otherwise, for each ground truth disease, the number of matching tokens in the predicted search diseases is determined, and the reward is computed as the cube root of the fraction of matched tokens. Formally,

$$Rwd_S = \begin{cases} 0, & \text{if unmatched tags} \\ \left(\frac{\text{Number of matched tokens in search output}}{\text{Total number of tokens in ground truth diagnosis}} \right)^{1/k}, & \text{otherwise} \end{cases} \quad (15)$$

where the numerator and denominator are computed over all ground truth diagnoses.

Diagnosis reward. Rwd_D quantifies the accuracy and informativeness of the final model output by measuring how well the diseases highlighted in the answer (`\textbf{}` within `<diagnose>...</diagnose>`) match the ground truth diagnosis. First, a token-level similarity score sim_{diag} is computed between the predicted answer and the ground truth, identical to the similarity used in Rwd_S . This score is then linearly rescaled to the interval $[0.2, 0.8]$ via $0.2 + 0.6 \cdot \text{sim}_{\text{diag}}$ to avoid degenerate extremes. Next, the result is adjusted by the match reward Rwd_M , which can either increase the reward (for correct matching and reasoning) or decrease it (to penalize excessive or redundant matching and insufficient diversity). If the match constraints are violated or

Rwd_M is undefined, the answer reward is set to zero. Formally,

$$Rwd_D = \begin{cases} 0, & \text{if match constraints are violated} \\ 0.2 + 0.6 \cdot \text{sim}_{\text{diag}} + Rwd_M, & \text{otherwise} \end{cases} \quad (16)$$

where sim_{diag} denotes the token-level similarity between the answer and the ground truth diagnosis.

Reward combination. The overall reward Rwd integrates the match, search, and answer rewards, rigorously gated by the format coefficient $coef_F$ to ensure strict adherence to the required template. Only outputs that satisfy all formatting constraints ($\sigma_f = 1$) are eligible for positive reward, while any violation immediately results in $Rwd = 0$. For outputs passing the format check, the final reward is computed as a weighted sum of the match reward Rwd_M , the search reward Rwd_S , and the answer reward Rwd_D , each scaled by their respective weights w_M , w_S , and w_D to flexibly reflect their relative importance in the overall objective. To ensure stability and interpretability, the final reward is clipped to the interval $[0, 1]$. Formally,

$$Rwd = \text{clip}_{[0,1]}(\sigma_f \cdot (w_M \cdot Rwd_M + w_S \cdot Rwd_S + w_D \cdot Rwd_D)), \quad (17)$$

where $\text{clip}_{[0,1]}(\cdot)$ denotes element-wise clipping to the range $[0, 1]$. This unified formulation ensures that only structurally correct, diverse in reasoning, and diagnostically accurate outputs can achieve a high reward, while any deviation from the format or matching constraints results in zero reward and halts further evaluation.

5.2 Training Implementation

To incorporate the tailored reward mechanism into workflow optimization, we adopt the following training methods to provide sufficient technical support.

Reinforcement learning from agentic feedback. We use volcano Engine Reinforcement Learning (verl [48]) and the vLLM [49] open-source project for workflow construction. The main difference between traditional reinforcement learning and our system is that we add interleaved action-feedback during the rollout stage. Specifically, during the vLLM generation process, we detect whether tool invocation special tokens ($\langle \text{lookup} \rangle$, $\langle \text{match} \rangle$ and $\langle \text{search} \rangle$) tag appear iteratively, upon detected, the query will be sent to environment for external processing. The generation will be halted during this process. However, This realization approach are highly time-consuming because each token will be checked. For acceleration, we optimize this process through whole sequence generation and cutting the tokens after these special token. After agentic retrieval feedback retured, we append these tokens to the cutting position for further generation. This process is formulated as shown by Algorithm. 1.

In environment deployment, we start 4 servers to support agentic retrieval during the rollout stage. Specifically, the wikipeadia server, pubmed server, literature server for corresponding document searching in batch and the LLM server for long documentation summarization in batch, it is deployed at the same node during training and evaluation of the backbone architecture using the sgLang [50] framework for high throughput inference.

Group relative policy optimization [51]. In the framework implementation, we use GRPO as the algorithm for reinforcement learning conduction. Unlike those two-stage implementation (DeepSeek-R1), we exclude the supervised fine tuning because our base model (Qwen-series, Llama-series) already possessed the core ability of instruction following, and our training target is to evolving their diagnosis performance through the retrieval corpus exploration and tailored reasoning. Here let for each prompt q we sample a group of G outputs $\{c_i\}_{i=1}^G \sim \mathcal{M}_{\theta_{\text{old}}}(\cdot | q)$ and obtain scalar rewards $Rwd_i = r_\phi(q, c_i)$. We form the *group-relative advantage*

$$\hat{A}_{i,t} = \frac{Rwd_i - \frac{1}{G} \sum_{j=1}^G Rwd_j}{\sqrt{\frac{1}{G} \sum_{j=1}^G (Rwd_j - \frac{1}{G} \sum_{k=1}^G Rwd_k)^2}} \quad \forall t = 1, \dots, |c_i|,$$

We denote by $\mathcal{M}_{\theta_{\text{ref}}}$ the frozen reference policy (e.g. the SFT checkpoint). Then the GRPO loss is

Algorithm 1 Interleaved Agent–Environment Rollout

Require: Initial prompt $x^{(0)}$, max response length L_{\max}

Ensure: Final generated sequence x

```

1:  $x \leftarrow x^{(0)}$ 
2:  $k \leftarrow 1$ 
3: while  $|x| - |x^{(0)}| < L_{\max}$  do
4:    $\Delta \leftarrow \text{Gen}(x)$  truncated at first active tag
5:   if no active tag in  $\Delta$  then
6:      $x \leftarrow x, \|, \Delta$ 
7:     break
8:   else
9:     Detect first active tag  $T \in \langle/\text{lookup}\rangle, \dots$ 
10:    Extract query  $q$  from  $\Delta$  w.r.t.  $T$ 
11:     $e \leftarrow \text{Env}_T(q)$ 
12:     $x \leftarrow x, \|, \Delta, \|, e$ 
13:   end if
14:    $k \leftarrow k + 1$ 
15: end while
16: return  $x$ 

```

$$L_{\text{GRPO}}(\theta) = \mathbb{E}_{q, \{c_i\} \sim \mathcal{M}_{\theta_{\text{old}}}} \left[\frac{1}{G} \sum_{i=1}^G \frac{1}{|c_i|} \sum_{t=1}^{|c_i|} \left(-\hat{A}_{i,t} \log \mathcal{M}_{\theta}(c_{i,t} \mid q, c_{i,<t}) \right. \right. \\ \left. \left. + \beta D_{\text{KL}}(\mathcal{M}_{\theta}(\cdot \mid q, c_{i,<t}) \parallel \mathcal{M}_{\theta_{\text{ref}}}(\cdot \mid q, c_{i,<t})) \right) \right], \quad (18)$$

with

$$D_{\text{KL}}(\mathcal{M} \parallel \rho) = \sum_a \mathcal{M}(a) \log \frac{\mathcal{M}(a)}{\rho(a)}$$

penalizes deviation from the reference policy, and β controls the strength of that penalty.

Multi-stage reward adaption. During the training process, we find that one-stage training using the combination of all rewards will lead to misunderstanding of the model because of the limited exploration and restricted randomizability. We found that even if we tried different reward weights, the model would always optimize towards the direction of reward or penalty of one reward and tend to ignore other rewards. Therefore, in each stage of training, we set the coefficient of one reward to 0.9 and the other two to 0.05. This ensures that the optimization direction will not go wrong. After three rounds of training with this setting, we set the coefficients of Rwd_S , Rwd_M and Rwd_D to 0.3, 0.3 and 0.4 respectively for the final optimization. This process is formally as:

$$w_i^{(r)} = \begin{cases} 0.9, & r \in \{1, 2, 3\} \text{ and } i = r, \\ 0.05, & r \in \{1, 2, 3\} \text{ and } i \neq r, \\ (0.3, 0.3, 0.4)_i, & r = 4, \end{cases} \quad i \in \{1, 2, 3\} \equiv \{S, M, A\}, \quad (19)$$

$$\text{and at each stage } r : \quad Rwd^{(r)} = \sum_{i \in \{S, M, A\}} w_i^{(r)} Rwd_i,$$

Interestingly, we found that when only the patient matching reward was activated in the second stage, the final answer score would be significantly improved. This improvement was even greater than the improvement we achieved by focusing on optimizing the answer reward process in the third stage. This proves the effectiveness

of staged adaptation and the importance of process guidance.

5.3 Baselines

We introduce the comparison setting of our agentic RL training approach against other training and prompting approach, then further compare Deep-DxSearch against seven competing baseline methods including domain-adapted medical LLMs, foundation models, retrieval-augmented methods, and multi-agent frameworks, etc.

Basic Training & Prompting Approach

- **Vanilla model with direct inference.** We only prompt the vanilla model to direct diagnose according to its internal knowledge without any post-training. The medical retrieval corpus is disabled under this setting. The input is free-text clinical presentation and no chain-of-thought inference is implemented.
- **Training-free RAG-augmented prompting.** For comparison, we also include a prompt engineering based approach using the same retrieval corpus (the LLM can interact with the corpus at any time it decided to do). In this inference-only setting, we apply the same prompt design (see Supplementary Materials) as in our agentic RL training, but without incorporating any reward mechanism for optimization.
- **Target-only RL training.** In contrast to our agentic RL training approach, this target-only training variant removes the policy reward that guides the optimization of the reasoning and retrieval processes, resulting in supervision based solely on target outputs. For a fair comparison, we adopt the same environment settings and training parameters as in our full-component agentic RL training.

Competing Clinical Diagnostic Methods

- **General-purpose large language model.** In this work, we employ the Qwen2.5 [52] and Llama3.1 [53] series as the vanilla backbones for RL training. Specifically, considering the cost-effect tradeoff, we use `Qwen2.5-7B-Instruct`, `Qwen2.5-14B-Instruct`, and `Llama3.1-8B-Instruct`. For larger-scale LLMs as comparison baselines, we adopt GPT-4o (proprietary) [30] and DeepSeek-R1 (open-source) [31]. In particular, we access their official APIs with the models `DeepSeek-R1-0528` and `gpt-4o-2024-11-20`.
- **Biomedical CLIP-based encoder.** These models are trained on large-scale biomedical text corpora using a contrastive learning approach. In this work, we adopt a representative approach: MedCPT [32] for comparison, treating the clinical presentation as the “article” and the diagnosis as the “query.” Specifically, we use the official Hugging Face checkpoint `ncbi/MedCPT-Cross-Encoder`.
- **Medical large language model.** Domain-adaptive pretraining (DAPT) of general LLMs on medical corpora is a common approach for clinical adaptation [54]. In this work, we adopt the newly developed Baichuan-M1 model as a baseline with the official checkpoint `baichuan-inc/Baichuan-M1-14B-Instruct`.
- **Medical foundation model.** We include this category as multi-modal, multi-task generalisers. Medical foundation models such as Meditron [55] and MedFound [56] demonstrate strong capabilities across diverse clinical scenarios, including diagnosis. In this work, we select MedGemma [34] for its improved instruction-following ability and more recent medical knowledge cutoff. The official checkpoint used is `google/medgemma-27b-text-it`.
- **Medical RAG-based framework.** Different from our retrieval approach, these methods typically rely on a general medical knowledge corpus specified via a system prompt, without fine-tuning. In this work, we include the MedRAG [35] framework, following the official implementation `Teddy-XiongGZ/MedRAG`.
- **Chain-of-Thought agentic model.** This type of model incorporates the chain-of-thought paradigm through supervised fine-tuning (SFT), enhancing diagnostic ability via explicit reasoning. In this work, we adopt CoD [36], using the official checkpoint `FreedomIntelligence/DiagnosisGPT-34B`.
- **Multi-agent consultation system.** Multi-expert consultation is a common and effective practice in clinical diagnosis. Recent agentic systems employ multiple agents as role-playing experts to improve diagnostic reliability. In this work, we adopt MAC [37], following the official implementation `geteff1/Multi-agent-conversation-for-disease-diagnosis` for comparison.

5.4 Evaluation Settings

In this section, we first define the metrics used to evaluate model performance and then describe the experimental setup for comprehensive benchmarking.

Metric Inclusion

- **Top-N accuracy (Acc@N)** [57]. This widely used metric measures whether the correct diagnosis is included among the top-N predictions. Specifically, if any of the n most likely predicted diseases match the ground-truth diagnosis, the case is counted as “Top-N correct.” The metric is reported as a score between 0 and 1, representing the proportion of cases that are Top-N correct.
- **Hit@N.** This metric is used exclusively to evaluate the diagnostic policy of Deep-DxSearch. During patient record matching, if any of the top-N retrieved records share the same diagnosis as the ground truth, the case is counted as a “hit.” The metric is reported as a score between 0 and 1, representing the proportion of patient record matches that are hits.
- **Hint score.** This metric is used in the study of the diagnostic process. It measures whether the ground-truth disease is mentioned during the reasoning process, even if the final diagnosis is incorrect, thereby providing a potential “hint” to assist clinicians in their consideration. The metric is reported as a score between 0 and 1, representing the proportion of diagnostic workflows that contain such hints.

Benchmark Setup

To investigate whether Deep-DxSearch improve its ability under agentic RL training, how it compared to state-of-the-art methods and what is the optimized policy toward more accurate diagnosis, we conducted a comprehensive evaluations through five spectrum of diagnostic performance.

First, we evaluate the diagnostic capability of agentic RL training with retrieval augmentation against vanilla direct inference. The we adjust baseline models for training to select the best base model with the consideration of computation cost. Specifically, we benchmark Qwen2.5-14B-Instruct, Llama3.1-8B-Instruct, and Qwen2.5-7B-Instruct, and assess performance using both top-1 and top-5 accuracy.

Second, We conduct a comparison experiment between Deep-DxSearch and the training-free RAG approach to demonstrate the advantages of the agentic RL-enhanced training method over direct retrieval and reasoning. Specifically, we use Qwen-14B-Instruct as the base model for both approaches and also compare them against the vanilla model. Performance is evaluated using top-1 accuracy and top-5 accuracy.

Third, we compare our framework against both general-purpose LLMs and medical-specific methods, including the Qwen2.5 series, Llama3.1-8B, DeepSeek-R1, GPT-4o, MedCPT, Baichuan-M1, MedGemma, Cod, MedRAG, and MAC. The comparison is conducted on three common-disease datasets (ID), three rare-disease datasets (ID) and two datasets of common and rare disease (OOD), measured with top-1 and top-5 accuracy.

Fourth, we compare Deep-DxSearch trained with the agentic RL approach against the target-only RL without intermediate policy reward. Using Qwen-14B-Instruct as the base model, we evaluate performance with top-1 accuracy, top-5 accuracy, and the “Hint” score, averaged over both common and rare disease diagnosis tasks.

Fifth, we conduct a component ablation study to evaluate the impact of each module in Deep-DxSearch by progressively removing elements from the retrieval corpus, transitioning from the full-component model to the vanilla model. We use Qwen-14B as the base model and assess performance using top-1 accuracy, top-5 accuracy, and the “Hint” score.

Sixth, we evaluate Deep-DxSearch’s ability to associate symptoms during training, from scratch to 800 steps, using the “Hit” metric. We select the target-only RL training approach as the baseline for comparison and measure the retrieval performance using Hit@20.

Seventh, we evaluate Deep-DxSearch’s ability in differential diagnosis by verifying whether at least one of the retrieved patient records shares the same diagnosis as the ground truth. We use the target-only RL approach as the baseline and assess performance during training using top-5 accuracy.

Eighth, we evaluate Deep-DxSearch’s ability to exclude irrelevant information by providing it with entirely inferential data during training. We employ the target-only RL approach for comparison and assess this capability by measuring the final top-5 accuracy on diagnostic conclusions.

References

- [1] Luciana D'Adderio and David W Bates. Transforming diagnosis through artificial intelligence. *npj Digital Medicine*, 8(1):54, 2025. [1](#)
- [2] Farieda Gaber, Maqsood Shaik, Fabio Allega, Agnes Julia Bilecz, Felix Busch, Kelsey Goon, Vedran Franke, and Altuna Akalin. Evaluating large language model workflows in clinical decision support for triage and referral and diagnosis. *npj Digital Medicine*, 8(1):263, 2025. [1](#)
- [3] Shuang Zhou, Zidu Xu, Mian Zhang, Chunpu Xu, Yawen Guo, Zaifu Zhan, Yi Fang, Sirui Ding, Jiashuo Wang, Kaishuai Xu, et al. Large language models for disease diagnosis: A scoping review. *npj Artificial Intelligence*, 1(1):9, 2025. [1](#)
- [4] Michael Moor, Oishi Banerjee, Zahra F H Abad, Harlan M. Krumholz, Jure Leskovec, Eric J. Topol, and Pranav Rajpurkar. Foundation models for generalist medical artificial intelligence. *Nature*, 616:259–265, 2023. [1](#)
- [5] Karen Ka Yan Ng, Izuki Matsuba, and Peter Chengming Zhang. Rag in health care: A novel framework for improving communication and decision-making by addressing llm limitations. *NEJM AI*, 2024. [1](#)
- [6] Yuhe Ke, Liyuan Jin, Kabilan Elangovan, Hairil Rizal Bin Abdullah, Nan Liu, Alex Tiong Heng Sia, Chai Rick Soh, Joshua Yi Min Tung, Jasmine Chiat Ling Ong, Chang Fu Kuo, Shao-Chun Wu, Vesela P. Kovacheva, and Daniel Shu Wei Ting. Retrieval augmented generation for 10 large language models and its generalizability in assessing medical fitness. *NPJ Digital Medicine*, 8, 2025. [1](#)
- [7] Lameck Mbangula Amugongo, Pietro Mascheroni, Steven Brooks, Stefan Doering, and Jan Seidel. Retrieval augmented generation for large language models in healthcare: A systematic review. *PLOS Digital Health*, 4(6):e0000877, 2025. [1](#)
- [8] Binxu Li, Tiankai Yan, Yuanting Pan, Jie Luo, Ruiyang Ji, Jiayuan Ding, Zhe Xu, Shilong Liu, Haoyu Dong, Zihao Lin, and Yixin Wang. MMedAgent: Learning to use medical tools with multi-modal agent. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 8745–8760, Miami, Florida, USA, November 2024. Association for Computational Linguistics. [1](#)
- [9] Shanghua Gao, Richard Zhu, Zhenglun Kong, Ayush Noori, Xiaorui Su, Curtis Ginder, Theodoros Tsiligkaridis, and Marinka Zitnik. Txagent: An ai agent for therapeutic reasoning across a universe of tools. *arXiv preprint arXiv:2503.10970*, 2025. [1](#)
- [10] Simone Kresevic, Mauro Giuffrè, Milos Ajcevic, Agostino Accardo, Lory S Crocè, and Dennis L Shung. Optimization of hepatological clinical guidelines interpretation by large language models: a retrieval augmented generation-based framework. *NPJ digital medicine*, 7(1):102, 2024. [1](#)
- [11] Rui Yang, Yilin Ning, Emilia Keppo, Mingxuan Liu, Chuan Hong, Danielle S Bitterman, Jasmine Chiat Ling Ong, Daniel Shu Wei Ting, and Nan Liu. Retrieval-augmented generation for generative artificial intelligence in health care. *npj Health Systems*, 2(1):2, 2025. [1](#)
- [12] Kimberly LeBlanc, Emily Glanton, Anna Nagy, Jorick Bater, Tala Berro, Molly A McGuinness, Courtney Studwell, Undiagnosed Diseases Network, and Matthew Might. Rare disease patient matchmaking: development and outcomes of an internet case-finding strategy in the undiagnosed diseases network. *Orphanet journal of rare diseases*, 16(1):210, 2021. [1](#)
- [13] Yixiang Chen, Penglei Sun, Xiang Li, and Xiaowen Chu. Mrd-rag: enhancing medical diagnosis with multi-round retrieval-augmented generation. *arXiv preprint arXiv:2504.07724*, 2025. [1](#)
- [14] Dongyu Ru, Lin Qiu, Xiangkun Hu, Tianhang Zhang, Peng Shi, Shuaichen Chang, Cheng Jiayang, Cunxiang Wang, Shichao Sun, Huanyu Li, et al. Ragchecker: A fine-grained framework for diagnosing retrieval-augmented generation. *Advances in Neural Information Processing Systems*, 37:21999–22027, 2024. [1](#)

- [15] Yunfan Gao, Yun Xiong, Yijie Zhong, Yuxi Bi, Ming Xue, and Haofen Wang. Synergizing rag and reasoning: A systematic review. *arXiv preprint arXiv:2504.15909*, 2025. 1
- [16] Minbyul Jeong, Jiwoong Sohn, Mujeen Sung, and Jaewoo Kang. Improving medical reasoning through retrieval and self-reflection with retrieval-augmented large language models. *Bioinformatics*, 40(Supplement_1):i119–i129, 2024. 1
- [17] James L Cross, Michael A Choma, and John A Onofrey. Bias in medical ai: Implications for clinical decision-making. *PLOS Digital Health*, 3(11):e0000651, 2024. 2
- [18] Kristin M Kostick-Quenet and Sara Gerke. Ai in the hands of imperfect users. *NPJ digital medicine*, 5(1):197, 2022. 2
- [19] Byron Crowe and Jorge A Rodriguez. Identifying and addressing bias in artificial intelligence. *JAMA Network Open*, 7(8):e2425955–e2425955, 2024. 2
- [20] Qiaoyu Zheng, Chaoyi Wu, Pengcheng Qiu, Lisong Dai, Ya Zhang, Yanfeng Wang, and Weidi Xie. How well can modern llms act as agent cores in radiology environments? *arXiv preprint arXiv:2412.09529*, 2024. 2
- [21] Richard Sutton. The bitter lesson. *Incomplete Ideas (blog)*, 13(1):38, 2019. 2, 15
- [22] Alistair E. W. Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J. Pollard, Benjamin Moody, Brian Gow, Li wei H. Lehman, Leo Anthony Celi, and Roger G. Mark. Mimic-iv, a freely accessible electronic health record dataset. *Scientific Data*, 10, 2023. 4, 7, 28
- [23] Naomi Miller, Eve-Marie Lacroix, and Joyce Backus. Medlineplus: building and maintaining the national library of medicine’s consumer health web service. *Bulletin of the Medical Library Association*, 88 1:11–7, 2000. 7
- [24] Zhengyun Zhao, Qiao Jin, and Sheng Yu. Pmc-patients: A large-scale dataset of patient notes and relations extracted from case reports in pubmed central. *ArXiv*, abs/2202.13876, 2022. 7, 28
- [25] Shu Chen, Zeqian Ju, Xiangyu Dong, Hongchao Fang, Sicheng Wang, Yue Yang, Jiaqi Zeng, Ruisi Zhang, Ruoyu Zhang, Meng Zhou, Penghui Zhu, and Pengtao Xie. Meddialog: A large-scale medical dialogue dataset. *ArXiv*, abs/2004.03329, 2020. 7, 28
- [26] Ziyuan Zhao, Tsinghua Medicine, Peking Union Medical College, Department of Statistics, and Data Science at Tsinghua University. Rarearena: A comprehensive rare disease diagnostic dataset with nearly 50,000 patients covering more than 4000 diseases. <https://github.com/zhao-zy15/RareArena>, 2025. Accessed: 2025-08-20. 7
- [27] Xuanzhong Chen, Xiaohao Mao, Qihan Guo, Lun Wang, Shuyang Zhang, and Ting Chen. Rarebench: Can llms serve as rare diseases specialists? *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024. 7, 29
- [28] Abdullah Al Shafi, Rowzatul Zannat, Abdul Muntakim, and Mahmudul Hasan. A structured dataset of disease-symptom associations to improve diagnostic accuracy. *ArXiv*, abs/2506.13610, 2025. 7, 29
- [29] Weike Zhao, Chaoyi Wu, Yanjie Fan, Xiaoman Zhang, Pengcheng Qiu, Yuze Sun, Xiao Zhou, Yanfeng Wang, Ya Zhang, Yongguo Yu, Kun Sun, and Weidi Xie. An agentic system for rare disease diagnosis with traceable reasoning. *ArXiv*, abs/2506.20430, 2025. 7, 29
- [30] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 10, 21
- [31] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. 10, 21

- [32] Qiao Jin, Won Kim, Qingyu Chen, Donald C. Comeau, Lana Yeganova, John Wilbur, and Zhiyong Lu. Biocpt: Contrastive pre-trained transformers with large-scale pubmed search logs for zero-shot biomedical information retrieval. *Bioinformatics*, 39(11), 2023. [10](#), [21](#)
- [33] Bingning Wang, Haizhou Zhao, Huozhi Zhou, Liang Song, Mingyu Xu, Wei Cheng, Xiangrong Zeng, Yupeng Zhang, Yuqi Huo, Zecheng Wang, Zhengyun Zhao, Da Pan, Fan Yang, Fei Kou, Fei Li, Fuzhong Chen, Guosheng Dong, Han Liu, Hongda Zhang, Jin He, Jinjie Yang, Kangxi Wu, Kegeng Wu, Lei Su, Linlin Niu, Lin-Lin Sun, Mang Wang, Peng Fan, Qi Shen, Rihui Xin, Shunya Dang, Songchi Zhou, Weipeng Chen, Wenjing Luo, Xin Chen, Xin Men, Xionghai Lin, Xu Dong, Yan Zhang, Yi qun Duan, Yuyan Zhou, Zhi-Xing Ma, and Zhi-Yan Wu. Baichuan-m1: Pushing the medical capability of large language models. *ArXiv*, abs/2502.12671, 2025. [10](#)
- [34] Andrew Sellergren, Sahar Kazemzadeh, Tiam Jaroensri, Atilla Kiraly, Madeleine Traverse, Timo Kohlberger, Shawn Xu, Fayaz Jamil, Cian Hughes, Charles Lau, et al. Medgemma technical report. *arXiv preprint arXiv:2507.05201*, 2025. [10](#), [21](#)
- [35] Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. Benchmarking retrieval-augmented generation for medicine. *ArXiv*, abs/2402.13178, 2024. [10](#), [21](#), [29](#)
- [36] Junying Chen, Chi Gui, Anningzhe Gao, Ke Ji, Xidong Wang, Xiang Wan, and Benyou Wang. Cod, towards an interpretable medical agent using chain of diagnosis. *ArXiv*, abs/2407.13301, 2024. [10](#), [21](#)
- [37] Xi Chen, Huahui Yi, Mi Hee You, Weizhi Liu, Li Wang, Hairui Li, Xue Zhang, Yingman Guo, Lei Fan, Gang Chen, Qicheng Lao, Weili Fu, Kang Li, and Jian Li. Enhancing diagnostic capability with multi-agents conversational large language models. *NPJ Digital Medicine*, 8, 2025. [10](#), [21](#)
- [38] Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630:625 – 630, 2024. [14](#)
- [39] Justin T Reese, Leonardo Chimirri, Yasemin Bridges, Daniel Danis, J Harry Caufield, Kyran Wissink, Julie A McMurry, Adam SL Graefe, Elena Casiraghi, Giorgio Valentini, et al. Systematic benchmarking demonstrates large language models have not reached the diagnostic accuracy of traditional rare-disease decision support tools. *medRxiv*, 2024. [14](#)
- [40] Maxime Griot, Coralie Hemptinne, Jean Vanderdonckt, and Demet Yuksel. Large language models lack essential metacognition for reliable medical reasoning. *Nature communications*, 16(1):642, 2025. [14](#)
- [41] Ran Xu, Wenqi Shi, Yue Yu, Yuchen Zhuang, Yanqiao Zhu, May Dongmei Wang, Joyce C. Ho, Chao Zhang, and Carl Yang. Bmretriever: Tuning large language models as better biomedical text retrievers. *ArXiv*, abs/2404.18443, 2024. [14](#)
- [42] Julien Delile, Srayanta Mukherjee, Anton Van Pamel, and Leonid Zhukov. Graph-based retriever captures the long tail of biomedical knowledge. *ArXiv*, abs/2402.12352, 2024. [14](#)
- [43] Jiayi Qu, Jun Liu, Xiangjun Liu, Meihui Chen, Jinchi Li, and Jintao Wang. Pncl: Mitigating llm hallucinations in noisy environments-a medical case study. *Inf. Fusion*, 123:103328, 2025. [14](#)
- [44] Medical large language model for diagnostic reasoning across specialties. *Nature medicine*, 2025. [14](#)
- [45] Zheng Wu, Kehua Guo, Entao Luo, Tianrou Wang, Shoujin Wang, Yi Yang, Xiangyuan Zhu, and Rui Ding. Medical long-tailed learning for imbalanced data: Bibliometric analysis. *Computer methods and programs in biomedicine*, 247:108106, 2024. [14](#)
- [46] Keer Lu, Zheng Liang, Zhuoran Zhang, Da Pan, Shusen Zhang, Xin Wu, Zenan Zhou, Guosheng Dong, Bin Cui, Tengjiao Wang, et al. Med-r²: Crafting trustworthy llm physicians via retrieval and reasoning of evidence-based medicine. *arXiv preprint arXiv:2501.11885*, 2025. [14](#)
- [47] Yakun Zhu, Zhongzhen Huang, Linjie Mu, Yutong Huang, Wei Nie, Jiaji Liu, Shaoting Zhang, Pengfei Liu, and Xiaofan Zhang. Diagnosisarena: Benchmarking diagnostic reasoning for large language models. *ArXiv*, abs/2505.14107, 2025. [14](#)

- [48] Guangming Sheng, Chi Zhang, Zilingfeng Ye, Xibin Wu, Wang Zhang, Ru Zhang, Yanghua Peng, Haibin Lin, and Chuan Wu. Hybridflow: A flexible and efficient rlhf framework. *arXiv preprint arXiv:2409.19256*, 2024. [19](#)
- [49] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023. [19](#)
- [50] Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph Gonzalez, Clark W. Barrett, and Ying Sheng. Sqlang: Efficient execution of structured language model programs. In *Neural Information Processing Systems*, 2023. [19](#)
- [51] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Jun-Mei Song, Mingchuan Zhang, Y. K. Li, Yu Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *ArXiv*, abs/2402.03300, 2024. [19](#)
- [52] Qwen An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxin 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, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yi-Chao Zhang, Yunyang Wan, Yuqi Liu, Zeyu Cui, Zhenru Zhang, Zihan Qiu, Shanghaoran Quan, and Zekun Wang. Qwen2.5 technical report. *ArXiv*, abs/2412.15115, 2024. [21](#)
- [53] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407, 2024. [21](#)
- [54] Karan Singhal, Shekoofeh Azizi, Tao Tu, S Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620(7972):172–180, 2023. [21](#)
- [55] Zeming Chen, Alejandro Hernández Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Kopf, Amirkeivan Mohtashami, Alexandre Sallinen, Alireza Sakhaeirad, Vinitra Swamy, Igor Krawczuk, Deniz Bayazit, Axel Marmet, Syrielle Montariol, Mary-Anne Hartley, Martin Jaggi, and Antoine Bosselut. Meditron-70b: Scaling medical pretraining for large language models. *ArXiv*, abs/2311.16079, 2023. [21](#)
- [56] Xiaohong Liu, Hao Liu, Guoxing Yang, Zeyu Jiang, Shuguang Cui, Zhaoze Zhang, Huan Wang, Liyuan Tao, Yongchang Sun, Zhu Song, Tianpei Hong, Jin Yang, Tianrun Gao, Jiangjiang Zhang, Xiaohu Li, Jing Zhang, Ye Sang, Zhao Yang, Kanmin Xue, Song Wu, Ping Zhang, Jian Yang, Chunli Song, and Guangyu Wang. A generalist medical language model for disease diagnosis assistance. *Nature medicine*, 2025. [21](#)
- [57] Daniel McDuff, Mike Schaeckermann, Tao Tu, Anil Palepu, Amy Wang, Jake Garrison, Karan Singhal, Yash Sharma, Shekoofeh Azizi, Kavita Kulkarni, Le Hou, Yong Cheng, Yun Liu, S. Sara Mahdavi, Sushant Prakash, Anupam Pathak, Christopher Semturs, Shwetak N. Patel, Dale R. Webster, Ewa Dominowska, Juraj Gottweis, Joelle Barral, Katherine Chou, Greg S Corrado, Yossi Matias, Jacob Sunshine, Alan Karthikesalingam, and Vivek Natarajan. Towards accurate differential diagnosis with large language models. *Nature*, 642:451 – 457, 2023. [22](#)

A Supplementary

A.1 Initialized Framework Instruction

Framework Instruction (System Prompt)

You are an AI assistant specializing in diagnosing diseases based on phenotypes or symptoms.

Task Description:

Your task is to analyze patient clinical presentation including phenotypes or symptoms and make a final disease diagnosis through systematic medical reasoning using the available tools.

Available Tools:

1. **Disease Information Guideline Lookup Tool:** Use the `<lookup>` tag to query typical phenotypes or symptoms of specific diseases.

Format: `<lookup> disease1, disease2... </lookup>`

The system returns common phenotypes for each disease enclosed in a `<guide>` tag.

2. **Patient Record Database Match Tool:** Use the `<match>` tag to submit a list of phenotypes. The system returns similar known cases, including diseases and their corresponding symptoms, enclosed in a `<refer>` tag.

Format: `<match> phenotype1, phenotype2, phenotype3... </match>`

3. **Medical Knowledge Corpus Search Tool:** Use the `<search>` tag to retrieve knowledge from Wikipedia, PMC, or textbooks using free-text queries (do not use commas within each question).

Format: `<search> |WIKI| query1, query2... </search> or <search> |PMC| query1, query2... </search> or <search> |BOOK| query1, query2... </search>`

Specify the source using the prefix `|WIKI|`, `|PMC|`, or `|BOOK|`. The system returns the retrieved content in a `<result>` tag.

Allowed Actions:

1. `<think> </think>`: Active action. Use for the analysis process or reasoning chain between actions.
2. `<lookup> </lookup>`: Active action. Use to look up up to 10 diseases within one `<lookup>` tag.
3. `<guide> </guide>`: Passive action. Returned by the system after a `<lookup>` action.
4. `<match> </match>`: Active action. Use to match a series of patient cases related to the query phenotypes.
5. `<refer> </refer>`: Passive action. Returned by the system after a `<match>` action.
6. `<search> </search>`: Active action. Use to search knowledge from only one source, with up to three queries (separated by commas) per `<search>` tag.
7. `<result> </result>`: Passive action. Returned by the system after a `<search>` action.
8. `<diagnose> </diagnose>`: Active action. Analyze all reference information and synthesize to make the final disease diagnosis.

Format Requirements:

- `<think>` must appear between two active actions.
- `<lookup>` may appear at most once. The content should only include diseases, not symptoms or phenotypes.
- `<match>` may appear up to three times. The content should only include symptoms or phenotypes, not diseases.
- `<search>` may appear at most twice. The content must follow the `|Source| query1, query2` format, with up to three queries at a time.
- The `<diagnose>` tag is mandatory at the end. Provide up to five possible disease diagnoses, enclosed in LaTeX bold format: `\textbf{Disease1}`, `\textbf{Disease2}`, etc.
- No text may appear outside of the specified tags.

Phenotype Query Refinement Guide:

If repeating the `<match>` step for more patient case references, refine the query phenotypes by one or more of the following:

- Adding related phenotypes commonly seen in suspected disease categories
- Replacing phenotypes with alternative medical terminology
- Including potential complications or associated features
- Adding earlier or later stage manifestations
- Using symptoms from retrieved cases as references

Diagnostic Workflow:

The diagnostic workflow is flexible. There is no fixed order for using the `<lookup>`, `<match>`, or `<search>` tools; use them as appropriate. Ensure your disease diagnoses are enclosed with `\textbf{}` within the `<diagnose>` tag, with a maximum of five diagnoses.

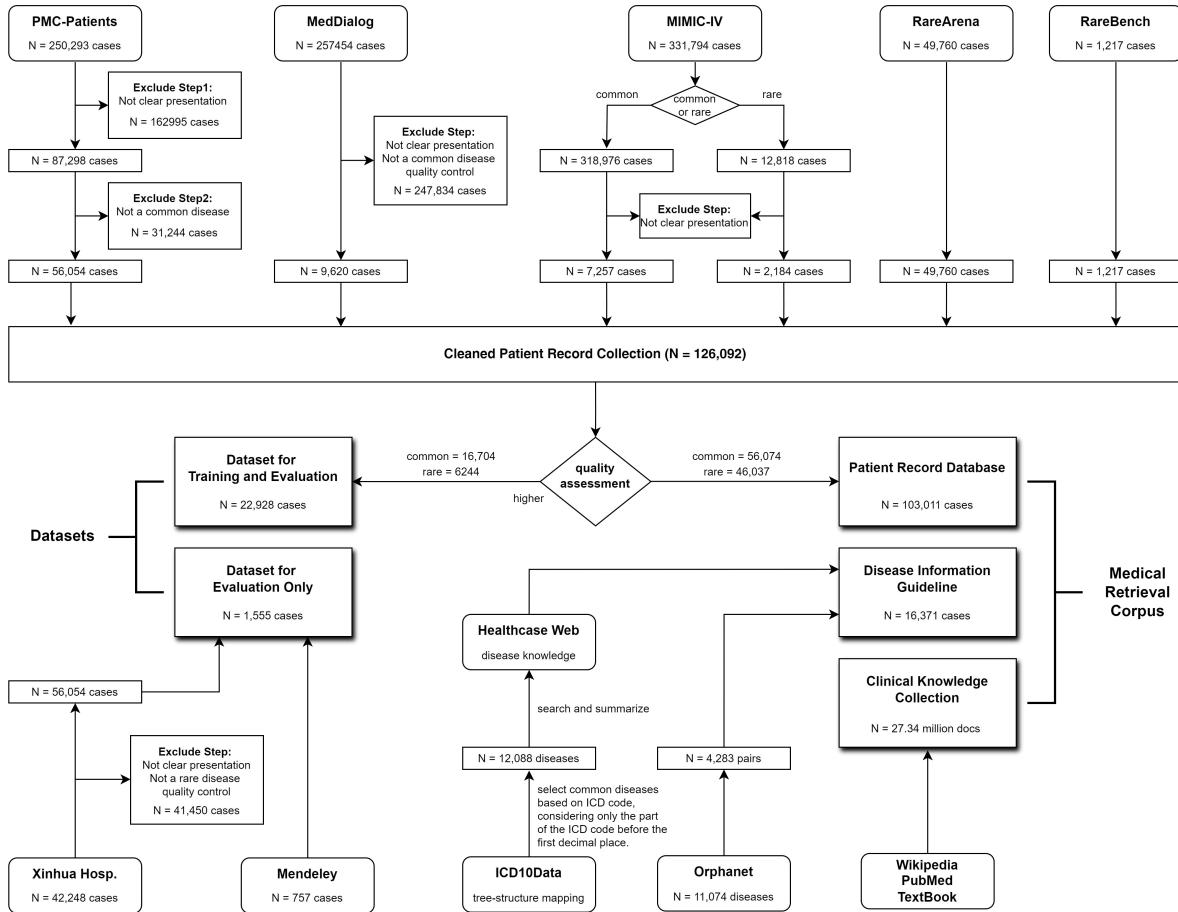


Figure 5 | Data processing procedure. The datasets for training and evaluation are derived from eight data sources and are split into training, evaluation, and evaluation-only sets. The medical retrieval corpus is constructed partially from these datasets as well as additional authoritative online resources.

A.2 Data and Resource Inclusion

We utilized a diverse set of clinical and biomedical resources to construct the training and evaluation datasets, as well as the retrieval corpus used by Deep-DxSearch (Fig. 5). Below, we describe each source in terms of its origin, pre-processing steps, and specific use within our framework.

MIMIC-IV [22]. This public dataset includes over 65,000 ICU admissions and 200,000 emergency department visits. After filtering for quality and clarity, we retained 61,471 cases. From these, we selected 7,257 high-quality cases involving common diseases (MIMIC-C) and 2,184 rare disease cases (MIMIC-R) for training and evaluation. The remaining 52,030 cases were incorporated into the retrieval corpus as part of the patient record database.

PMC-Patients [24]. Through the 167k patient summaries publicly collected from PubMed Central, we selected 56,054 cases with clearly described disease-symptom associations. Among these, 6,421 high-quality cases were used in training and evaluation; the rest 49,633 cases were included in the retrieval corpus. All disease and symptom terms were normalized to standard international ontologies.

MedDialog [25]. The official version of this dataset provides both Chinese and English data, collected respectively from online consultation platforms in Chinese- and English-speaking communities. We selected the English portion of this dataset, excluding rare disease cases. Of 9,620 total cases, 3,206 were retained for training and evaluation, and 6,414 were added to the retrieval corpus. As with other datasets, all terms were

standardized to internationally recognized coding systems.

RareArena⁶. This dataset is sourced from PMC-Patients, containing 50,000 patient records and annotations for over 4,000 diseases. We randomly selected 3,242 cases to serve as the training and evaluation sets, while the remaining cases were included in the patient record database.

RareBench [27]. It serves as a benchmark for rare disease diagnosis and is divided into public and private components. We used 1,217 cases from the public source (RAMEDIS, MME, HMS, and LIRICAL). Of these, 798 cases were randomly selected for use in the training and evaluation sets, while the remaining 419 cases were included in the patient record database. All data were standardized to internationally recognized codes.

Mendeley [28]. This is a structured resource released in June 2025, containing binary associations between 85 diseases and 172 symptoms. Curated from peer-reviewed literature and reputable databases. We choose this dataset for zero-shot evaluation because its post-June 2025 release ensures all tested models have no prior exposure, offering a fair assessment of their generalization to new, real-world medical data.

Xinhua Hosp [29]. This in-house datasets comprises all rare disease diagnostic records in *Xinhua Hospital Affiliated To Shanghai Jiao Tong University School of Medicine* from 2014 to 2025, totaling 352,424 entries. After filtering and deduplication, 5,820 high-quality cases were retained for evaluation only, ensuring zero training exposure due to privacy concerns.

ICD10Data. We extracted disease names and codes from the official ICD-10-CM classification, yielding 12,088 common and 4,283 rare diseases. This taxonomy was used to construct our disease information guide.

Orphanet. We obtained 11,074 Orpha codes, including phenotype probability distributions for 4,283 rare diseases. These were integrated into the structured knowledge base to support phenotype-driven reasoning.

Healthcare Websites. We curated disease descriptions, symptoms, and other clinical features from reliable medical sources (*e.g.*, NCBI, WebMD, NIH, Mayo Clinic). Using GPT-4o, we summarized and standardized 142,141 unique disease–symptom relationships for inclusion in the structured guideline.

PubMed and Wikipeadia. Following the MedRAG protocol [35], we included 23.9 million PubMed abstracts and 3.31 million Wikipedia medical entries to form a broad clinical knowledge base for retrieval. These documents provide contextual and background information for long-tailed or rare cases.

A.3 Details in Diagnostic Data Processing

We defined inclusion and exclusion criteria to curate high-quality diagnostic cases for training and evaluation. Our pipeline comprises three stages: (i) case-level filtering; (ii) symptom/phenotype extraction and filtering; and (iii) terminology normalization.

In the first stage, we filter these 7 collected dataset following the inclusion / exclusion criteria of:

- Case narratives reflect routine clinical documentation, with clear, well-structured descriptions.
- The diagnostic process described is causal rather than a subsequent symptom after a disease is diagnosed.
- The final diagnosis is reasonably inferable from the case description and is not trivially restated or explicitly disclosed.

In stage two, we performed symptom/phenotype extraction and additional filtering. We used GPT-4o to extract symptom/phenotype mentions from each case and to identify the candidate disease. We then applied the following inclusion/exclusion criteria:

- The disease is a well-defined clinical entity, and the listed phenotypes are representative and distinctive manifestations of that disease.
- Population-specific diseases (*e.g.*, in older adults, children, or females) are allowed, provided the case description indicates the relevant population.
- Phenotype mentions must not simply restate the disease name, its synonyms, or its immediate parent/child terms.

⁶<https://github.com/zhao-zy15/RareArena>

- The disease label must not be a symptom or a patient history item (*e.g.*, “fever”, “pain”, “history of smoking”); it must denote an actual medical diagnosis.
- The disease label must avoid vague qualifiers such as “unspecified” that preclude a clear diagnostic entity.

This procedure yielded a doubly filtered set of patient records paired with extracted disease labels and phenotypes/symptoms.

In stage three (terminology normalization), we mapped extracted terms to standard vocabularies. Phenotypes were mapped to Human Phenotype Ontology (HPO) terms; common diseases were mapped to ICD-10-CM codes; and rare diseases were mapped to Orphanet (ORPHA) codes. We used BioLORD to compute embeddings for both the standard terminologies and the extracted terms, selected the code with the highest cosine similarity for each term, and then performed human validation to ensure mapping quality.

For dataset splits, we reserved 500 common-disease cases for the test set (124 from MIMIC-Common, 141 from PMC-Patients, and 235 from MedDialog) and 500 rare-disease cases for the test set (167 from MIMIC-Rare, 167 from RareArena, and 166 from Rarebench). The remaining cases from each source were used for training and validation. Data from Mendeley and Xinhua Hospital were held out for evaluation only (external test).

A.4 Details in Retrieval Corpus Construction

We aggregated multi-source data from open clinical datasets, partner medical centers, and public web sources to build: (i) a disease–symptom guideline for instruction; (ii) a patient-record repository for similar-case retrieval; and (iii) a web-scale clinical knowledge collection for knowledge retrieval.

A.4.1 Disease Information Guideline

Many diseases have characteristic symptom/phenotype profiles summarized by clinicians and published for educational purposes by reputable organizations (*e.g.*, PubMed, Mayo Clinic, NCBI, WebMD, NIH).

In order to compile a comprehensive list of diseases, we crawled the relevant content on the ICD-10-CM webpage and sorted out more than 10,000 diseases and their corresponding ICD codes by keeping only the first decimal place of the ICD-10 code (*e.g.*, A10.0). We take this disease-ICD-10 code mapping as the official manual

In addition, we also selected diseases from MIMIC-IV, PMC-Patient, MedDialogue and other datasets, and used the BioLORD feature encoder to match these diseases with the diseases in the ICD-10 manual and normalize them to the ICD code. So far, we have obtained more than 15,000 diseases.

Next, we collected symptom/phenotype information for each disease from one or more reliable sources (*e.g.*, government public health agencies, academic institutions, and major clinics), archived the relevant content, and stored source links for provenance and copyright compliance. We used the open-source model DeepSeekV3 to clean and summarize these materials, producing concise lists of common symptoms/phenotypes per disease. To standardize terminology, we compiled the HPO term–code correspondences and again used BioLORD to map extracted symptom expressions to HPO terms and codes.

Finally, recognizing differences in diagnostic difficulty, we split the guideline into common-disease and rare-disease subsets. We first assigned a rarity label with DeepSeekV3 and then standardized it using the Orphanet catalog of rare diseases and ORPHA codes, yielding two parallel guidelines for common and rare conditions.

A.4.2 Patient Record Database

We constructed a patient-record repository for similar-case retrieval using cases not included in training or evaluation, drawn from MIMIC-Common, PMC-Patients, MedDialog, Xinhua-Rare, RareArena, and RareBench. Owing to the long-tailed distribution of diseases, this corpus does not cover the full spectrum of common and rare conditions.

To broaden coverage, we explored synthesis-based augmentation. We first identified diseases that appear in our disease catalog but are absent from the repository. For each such disease, we drafted synthetic cases by sampling symptoms from the disease–symptom guideline and perturbing them using the HPO phenotype hierarchy (phenotype relationship graph): adding related phenotypes, substituting clinically equivalent terminology,

incorporating earlier- or later-stage manifestations, or selectively removing findings. We then used GPT-4o to screen each synthetic case for internal consistency and contradictions, followed by human review to assess clinical plausibility.

In ablation study, these synthetic cases did not yield measurable gains in diagnostic performance. To avoid distributional shift and potential biases, we therefore excluded them from the current version of the patient-record database.

A.4.3 Clinical Knowledge Collection

To build the knowledge base for our retrieval-augmented generation (RAG) system, we integrated three complementary corpora:

- **PubMed:** a large subset comprising 23.9 million biomedical records with valid titles and abstracts, providing broad coverage of the medical literature.
- **Authoritative medical textbooks:** 18 medical textbooks commonly used for United States Medical Licensing Examination (USMLE) preparation. We segmented each book into chunks of up to 1000 characters to facilitate efficient indexing and retrieval.
- **Wikipedia:** a general-domain corpus obtained from Hugging Face and preprocessed with the same chunking configuration as the textbooks, included to assess the contribution of general knowledge to medical question answering.

A.5 Statistics of Medical Retrieval Corpus and Prepared Dataset

Detailed statistics are presented at Tab. 4, 5, 6, 7.

Table 4 | Statistics on disease information guideline. “Relation” means the disease-symptom pair appeared, “a/b” means a out of b, where the former one denotes items in the guideline, the latter one denotes the total numbers in official settings. Here we consider only ICD codes reserved to two decimal places. “Source” means the average source numbers we used to summarize the phenotypes or symptoms of each disease.

Category	Disease	Phenotype	Relation	ICD Coverage	Orpha Coverage	HPO Coverage	Source
Common	12,088	31,837	142,141	9615/9615	-	4970/17232	2.87
Rare	4,283	8,600	114,961	-	4283/11047	8595/17232	1.00

A.6 Details in Retrieval Methods

To maximize the efficacy and performance of the interaction with our proposed medical retrieval corpus, we treat each retrieval action and the observation of the action as tool and input arguments. Here we detailed the formulation of these tools including the Phenotype Parser, Patient Matcher, knowledge Searcher and MedDoc Summarizer.

Phenotype parser. This tool is designed for the retrieving from the disease information guideline. We use BM25 search algorithm to build this tool for phenotype parsing with the input of a list of diseases. To optimize the response time, we process it batch by batch for searching process acceleration. Specifically, take $\mathcal{D} = \{d_1, d_2, \dots, d_m\}$ as input where d_i denotes the i^{th} disease waiting for searching, then the general process could be denoted as:

$$T_{PP}(\mathcal{D}) = \left\{ \left(d, \begin{cases} \mathcal{P}(\hat{d}), & \text{if } \text{BM25}(d, \hat{d}) \geq \tau \\ \text{no reference}, & \text{otherwise} \end{cases} \right) \mid d \in \mathcal{D}, \hat{d} = \arg \max_{d' \in \mathcal{M}_{\text{disease}}} \text{BM25}(d, d') \right\} \quad (20)$$

Here, $\text{BM25_Match}(d, \mathcal{M}_{\text{disease}})$ denotes the best-matching disease \hat{d} for a query d in the reference corpus $\mathcal{M}_{\text{disease}}$ using the standard BM25 algorithm, where the BM25 score between a tokenized query q and a

Table 5 | Body system-level disease distribution. We analyze diseases in all patients records, classified them according to body system and calculated the proportion of cases in each category as percentages.

Blood, Heart and Circulation	Brain and Nerves	Bones, Joints and Muscles
15.0	14.9	12.4
Digestive System	Immune System	Skin, Hair and Nails
11.8	11.4	10.1
Lungs and Breathing	Endocrine System	Eyes and Vision
6.4	6.1	4.7
Female Reproductive System	Kidneys and Urinary System	Mouth and Teeth
4.7	4.5	3.1
Ear, Nose and Throat	Male Reproductive System	Others
2.9	1.7	15.3

Table 6 | Statistics on clinical knowledge source.

Wiki Docs	Tokens per Doc	Pubmed Docs	Tokens per Doc	Textbook Docs	Tokens per Doc
3.31M	117	23.9M	164	125,847	152

Table 7 | Statistics on curated dataset for training and evaluation.

Items	Common Disease				RareDisease			
	MIMIC-C	PMC-Patient	MedDialog	Mendeley	MIMIC-R	RareArena	RareBench	Xinhua
Cases	7257	6421	3206	757	2184	3242	277	798
Avg Syms	7.63	7.92	6.08	4.98	11.76	8.33	11.91	4.19
Disease	1320	2775	3038	85	710	2576	121	324
Source	EHR	PubMed	Web Plat	Literature	EHR	PubMed	Literature	In-house

candidate disease name d' is defined as

$$\text{BM25}(q, d') = \sum_{t \in q} \text{IDF}(t) \cdot \frac{f(t, d') (k_1 + 1)}{f(t, d') + k_1(1 - b + b \frac{|d'|}{\text{avgdl}})}$$

with $f(t, d')$ being the frequency of token t in d' , $|d'|$ the number of tokens in d' , avgdl the average length of all disease names in the corpus, and k_1, b standard hyperparameters (e.g., $k_1 = 1.5, b = 0.75$). The inverse document frequency is computed as

$$\text{IDF}(t) = \log \left(\frac{N - n(t) + 0.5}{n(t) + 0.5} + 1 \right),$$

where N is the total number of diseases and $n(t)$ is the number of diseases containing token t . For each $d \in \mathcal{D}$, if the maximum BM25 score $\text{BM25}(d, \hat{d})$ exceeds a threshold τ , we return the top k (e.g., $k = 10$) high-frequency phenotypes for the matched disease, denoted as $\mathcal{P}(\hat{d})$; otherwise, we return “no reference”.

Patient matcher. This tool is designed to interact with the patient record database. When taking symptoms or phenotypes as input, matching to patients in similar situations can provide valuable references for current case diagnosis. Given that different patients may describe symptoms differently, lexical searching is not adopted. Instead, we use BioLORD embeddings to calculate semantic similarity between cases. Specifically, each phenotype or symptom s in a patient record is encoded as a feature vector $\mathbf{e}(s)$ using the BioLORD encoder. For a case i with set $\mathcal{P}_i = \{p_{i,1}, p_{i,2}, \dots, p_{i,n_i}\}$, we represent its overall case embedding as the transformation of the symptom embeddings:

$$\text{Sim}(\mathcal{P}_q, \mathcal{P}_i) = \frac{1}{|\mathcal{P}_q|} \sum_{j=1}^{|\mathcal{P}_q|} \max_{1 \leq k \leq |\mathcal{P}_i|} \cos(\mathbf{e}(p_{q,j}), \mathbf{e}(p_{i,k})) \quad (21)$$

where $\mathcal{P}_q = \{p_{q,1}, \dots, p_{q,n_q}\}$ is the query case, $\mathcal{P}_i = \{p_{i,1}, \dots, p_{i,n_i}\}$ is the i -th case in the database, and $\cos(\mathbf{a}, \mathbf{b})$ denotes the cosine similarity between two embedding vectors. For each query symptom $p_{q,j}$, we find its maximal similarity to all symptoms in the candidate case, and then average these maxima across all query symptoms.

The Patient Matcher tool T_{PM} returns the top- N cases with the highest similarity scores:

$$T_{\text{PM}}(\mathcal{P}_q) = \text{TopN}_i(\text{Sim}(\mathcal{P}_q, \mathcal{P}_i)) \quad (22)$$

where $\text{TopN}_i(\cdot)$ selects the N most similar cases from the database.

Knowledge searcher. This tool is designed to interact with medical knowledge collection. To deploy these corpora as an efficient retrieval service accessible to Large Language Models (LLMs), we developed an asynchronous web server using the Python-based FastAPI framework, served by Uvicorn. The system implements two mainstream retrieval paradigms: sparse retrieval, based on keyword frequency (BM25), and dense retrieval, based on semantic similarity. For sparse retrieval, we leveraged the Pyserini library to query a pre-constructed Lucene index. For dense retrieval, we first utilized the Transformers library to load pre-trained text embedding models (e.g., E5, BGE) to encode all text chunks into high-dimensional vectors. Subsequently, we employed the FAISS library to build an index for these vectors, enabling millisecond-level similarity searches across a massive vector space by leveraging its GPU acceleration capabilities. The server’s core logic is encapsulated within an Encoder class and multiple Retriever classes; the former handles text vectorization, while the latter executes the specific retrieval operations (either BM25 or Dense) based on the provided configuration. It calls a batch_search method to perform real-time query encoding and retrieves the top-k most relevant documents from the corresponding FAISS or Lucene index, returning the final results in JSON format. The entire service is initiated via a command-line script, which allows for flexible configuration of key parameters such as index and corpus paths, the choice of retrieval model, and the number of documents to return (top-k). This design results in a highly configurable, scalable, and high-performance retrieval backend.

MedDoc summarizer. Influenced by the context length, when the environment feedback is surpassing the max length limitation, the document summarizer tool is needed to summarize it into length-controllable content. Specifically, we take Qwen-14B-Instruct or GPT-4o as the summarizer and deploy it in our multi-source environment with batch inference adaption using the sgLang open-source framework. The prompt for summarization instruction is:

System Prompt:

You are a medical document summarization assistant. Given a search query and a retrieved document, your task is to summarize the document to directly and concisely answer the query.

- Extract the most relevant facts or statements from the document that directly answer the query. If more than 10 points are relevant, keep only the 10 most important.
- Your answer should be brief, focused, and contain no extra explanation.
- Format your answer as a JSON string, e.g., "answer": "...".
- If no relevant information can be found, respond with "answer": "no reference".

Agent:

Source: {source} | Query: {query}

A.7 Details in baseline approach

For vanilla model with direct diagnosis inference, we use the following prompt:

System Prompt:

You are a disease diagnosis assistant. Your task is to make diagnosis based on the given symptoms or phenotypes. The user input is a list of symptoms of phenotypes. Your answer should only be diseases without other explanations enclosed within LaTeX bold format: **Disease1**, **Disease2**, etc. Please make up to 5 diagnosis.

For training-free RAG, we use the same prompt as detailed in Supp Sec. A.1.

A.8 Experimental Results

Detailed results of the main dianogsis performance including the performance of each base LLMs, medical retrieval corpus enhanced LLMs and our approach is presented at Tab. 8. Comparison with other representative frameworks is presented at Tab. 9. A Deep quantification of why and how our approach achieved betther performance is presented at Tab. 10. The component impact ablation study is presented at Tab. 11.

Table 8 | Main diagnosis performance. We calculate top-1 and top-5 accuracy among common and rare disease diagnosis datasets and compare our Deep-DxSearch with other representative models. “Env” means we allow the model to use our proposed environment as assistance. All results are shown in percentage

Model	Common Disease Diagnosis						Rare Disease Diagnosis					
	MIMIC-C		PMC-Patient		MedDialog		MIMIC-R		RareArena		RareBench	
	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5
Qwen-2.5-14B	8.80	12.40	17.73	27.66	17.87	32.34	7.93	16.71	6.53	13.23	18.07	31.38
Baichuan-M1	11.8	14.48	26.95	39.84	26.81	38.85	8.35	19.25	10.69	21.63	26.93	44.79
DeepSeek-R1	5.65	15.32	29.62	41.52	28.34	40.96	12.05	23.90	10.98	22.56	28.22	50.83
GPT-4o	6.43	9.82	23.51	36.10	22.59	36.01	7.65	15.58	12.83	23.10	24.25	43.54
Qwen14B(<i>Env</i>)	13.22	15.91	24.38	35.57	24.69	36.22	16.54	24.33	10.08	15.47	34.70	59.20
GPT-4o(<i>Env</i>)	15.07	21.25	28.64	38.38	25.86	39.41	20.47	29.05	11.24	19.32	40.11	63.28
Ours (<i>Llama8B</i>)	21.05	27.83	34.15	45.74	35.51	46.92	42.00	55.02	22.41	29.95	64.33	73.86
Ours (<i>Qwen7B</i>)	33.09	42.87	41.41	46.80	49.28	55.34	52.44	61.53	25.97	35.32	64.47	79.51
Ours (<i>Qwen14B</i>)	35.22	46.83	40.29	47.75	48.81	60.04	52.11	64.57	28.14	39.22	70.48	82.96

Table 9 | Diagnosis performance compared to other frameworks. We use GPT-4o as the large language model base for MedRAG and MAC framework, for other framework, we just follow their official settings during benchmarking. All results are shown in percentage.

Framework	Category	Common Disease Diagnosis							Rare Disease Diagnosis						
		MIMIC-C			PMC-Patient		MedDialog		MIMIC-R			RareArena		RareBench	
		Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5	Acc@1	Acc@5
MedCPT	CLIP-based	0.00	0.81	7.80	17.73	6.81	17.45	4.79	8.38	1.8	2.99	4.82	11.45		
MedGemma	Foundation	18.60	29.00	26.95	39.01	20.43	32.77	12.57	21.56	10.78	19.76	28.92	54.82		
MedRAG	RAG-based	4.03	10.48	25.53	37.58	22.13	34.04	8.98	21.56	16.77	21.68	33.73	53.03		
COD	COT-Agent	0.81	7.26	11.35	21.99	70.64	90.64	4.19	19.16	2.99	11.98	2.41	8.43		
MAC	Multi-Agent	4.03	10.74	28.06	30.66	24.03	29.07	16.17	24.69	15.66	17.07	35.54	43.98		
Deep-DxSearch		35.22	46.83	40.29	47.75	48.81	60.04	52.11	64.57	28.14	39.22	70.48	82.96		

Table 10 | Explainability quantification. The performance variance during training process where “Hit@20” means the ground truth disease appearance in 20 retrieved patient records, “Acc@5” measures the final diagnosis accuracy in top 5 diagnosis. We set the batch size to 256, which is the number of data for one step. “Base” means the baseline training method with only supervised fine-tuning. All results are shown in percentage.

Target	Metric	Method	Training Steps								
			0	100	200	300	400	500	600	700	800
Symptom Association Capability	Hit@20	Base	24.13	24.56	25.08	26.88	27.03	25.17	25.09	24.21	23.33
		Ours	25.79	30.88	35.21	39.20	45.83	52.24	58.08	60.39	59.96
Differential Diagnosis Capability	Acc@5	Base	38.71	40.22	41.26	43.48	45.62	46.03	45.57	45.39	45.70
		Ours	41.70	45.92	52.63	50.39	59.56	63.35	70.23	68.44	71.07
Irrelevance Exclusion Ability	Acc@5	Base	18.44	19.25	23.51	26.08	25.47	24.84	22.17	23.30	23.20
		Ours	24.50	25.78	25.89	30.13	34.72	35.61	35.95	34.27	33.88

Table 11 | Ablation on framework components. The ablation is conducted on both common and rare disease diagnosis tasks. “Hint” is a soft metric that measures whether the ground truth disease appears in the reasoning or interaction process during diagnosis workflow without forcing to accurate diagnosis. All results are demonstrated in percentage.

Component Ablation	Common Disease Diagnosis				Rare Disease Diagnosis		
	Hint	Acc@1	Acc@5	Hint	Acc@1	Acc@5	
Full-components Deep-DxSearch	59.17	47.04	52.83	71.74	55.20	62.21	
w/o. policy reward supervision	51.64	30.36	41.88	62.57	33.06	48.55	
w/o. documentation summarization	53.40	25.15	34.04	50.13	27.45	40.69	
w/o. clinical knowledge collection	50.01	28.94	39.29	49.21	30.17	43.43	
w/o. disease information guideline	48.63	27.36	36.55	45.87	28.29	40.23	
w/o. patient record database	30.68	15.58	26.08	24.05	10.83	20.60	