Med-FoT: Boosting Real-World Diagnostic Accuracy and Reasoning in LLMs via Structured Medical Flow-of-Thought

Anonymous Author(s)
Affiliation
Address
email

Abstract

Recent advancements in large language models (LLMs) have improved performance on standardized medical benchmarks. However, existing medical benchmarks often rely on truncated context and idealized scenarios. In reality, clinical diagnosis involves analyzing extensive patient history, physical examination, laboratory test, and imaging reports under time constraints to reach a reliable conclusion. Consequently, LLMs often struggle with diagnostic accuracy and may generate inaccurate information, or "hallucinations," when faced with authentic patient cases. To address this challenge, we propose an agentic workflow named Flow-of-Thought (FoT), which breaks down clinical information into four stages: retrieval, preliminary diagnosis, final diagnosis, and recheck. Our pipeline mimics the thought processes of expert doctors and helps prevent random or misguided inferences that could lead to serious medical errors. We generate 2,000 cases covering 15 categories of abdominal diseases, complete with detailed FoT annotations to fine-tune our model. We also encourage the model to explore its reasoning paths using group relative policy optimization for reinforcement learning. Finally, we introduce **Med-FoT**, an LLM specifically designed for medical diagnosis. Experiments show that after both supervised fine-tuning and reinforcement learning, Med-FoT outperforms state-of-the-art medical reasoning models such as HuatuoGPT-o1 and MedReason, achieving accuracy comparable to closed-source models (e.g., OpenAI o4-mini). We also invite professional doctors to validate that our reasoning chains closely align with real-world clinical thought processes.

1 Introduction

The reasoning capacity of large language models (LLMs) has emerged as a primary metric for assessing progress toward artificial general intelligence (AGI) [**? ?**]. Recent breakthroughs, such as OpenAI o4 and Deepseek-R1, have achieved remarkable results on mathematical problem-solving and code-generation benchmarks [**? ?**]. However, the development of medical reasoning LLMs remains at an early stage. Recent research has investigated approaches for constructing medical chain-of-thought (CoT) datasets and used them to train reasoning models that achieve state-of-the-art performance on standard clinical benchmarks [**? ? ?**]. Yet these approaches concentrate on relatively simplified, human-curated scenarios that differ markedly from real-world clinical practice [**?**], where clinicians must assemble a patient's full spectrum of diagnostic data rather than work from pre-filtered excerpts.

For the real-world application, a common cost-efficient strategy is to distill knowledge from open-source LLMs that can natively generate extended CoT trajectories, e.g., Qwen-QwQ and Deepseek-R1. These models generate extended reasoning chains that systematically decompose complex

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

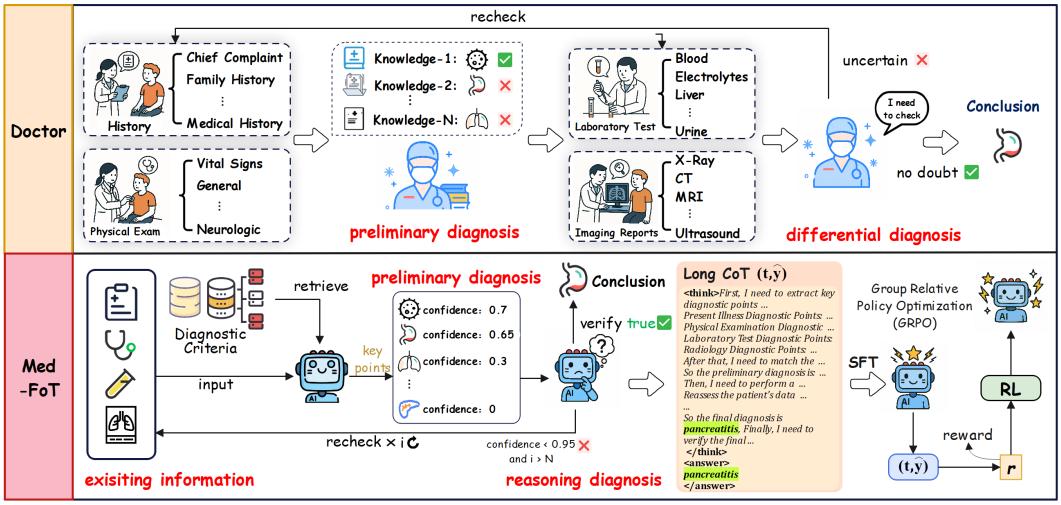

Figure 1: **Med-FoT performs similar cognitive pathways as medical experts.** The top part illustrates the typical diagnostic process, which combines different data sources and requires clinicians to keep their reasoning consistent. The bottom part shows the overview of Med-FoT.

problems and, when necessary, backtrack to correct errors before producing the final conclusion. Recent studies—most notably S1[**?**] and LIMO[**?**]—corroborate that high-quality, long-form CoT supervision is pivotal for robust LLM reasoning. **?**] demonstrate that a corpus of merely 17k CoT samples can enable a 32B model to perform approaching that of GPT-4o. Recent findings highlight that CoT reasoning, when coupled with large-scale reinforcement learning, can significantly enhance model performance [**?** **?** **?**]. They further contend that the structural integrity of the reasoning chain outweighs its lexical content. Yet, when transferred to clinical contexts, these trajectories often introduce domain-specific medical hallucinations. For instance, fabricated patient histories or erroneous medication dosages—thereby posing significant safety risks[**?** **?**].

Taking these challenges into account, we collaborate with board-certified physicians to introduce a novel flow-of-thought (**FoT**) framework. The FoT paradigm orchestrates heterogeneous LLMs and elevates zero-shot diagnostic performance by explicitly mimicking the reasoning steps of experts.. Building upon the FoT paradigm, we structure our methodology into three sequential phases that strengthen clinical reasoning: (1) A closed-loop data pipeline that, for each case, iterates through guideline retrieval, key-point extraction, preliminary diagnosis, definitive confirmation, and retrospective audit. (2) This pipeline yields 2,000 extended CoT samples spanning 14 abdominal pathologies, which we employ for supervised fine-tuning. (3) Lastly, we further improve long-context coherence via online gradient-regret policy optimization.

Leveraging LLaMA-3.1-8B-Instruct [**?**] and Qwen-2.5-7B-Instruct [**?**], we construct a domain-specialized medical language model, Med-FoT. Fine-tuned on curated MIMIC-IV clinical records [**?**] covering 15 abdominal pathologies [**?**], Med-FoT outperforms peer baselines (with 7-8B parameters) and even eclipses several 70B models on a demanding multi-step diagnostic benchmark. Empirical results demonstrate that our FoT framework concurrently optimizes diagnostic accuracy, reasoning depth, and interpretability. Our primary contributions are as follows:

1. We introduce the complex clinical diagnosis problem, a challenging multi-step test across 15 abdominal diseases that requires LLMs to integrate comprehensive patient history, examination findings, and test results into a diagnostic process.

2. With complex clinical problems, we propose an **agentic Flow of Thought (FoT) workflow**, which orchestrates LLMs with retrieval-augmented generation (RAG) to yield temporally and clinically coherent reasoning trajectories. Moreover, FoT supports training-free inference by emulating expert reasoning to bolster LLM performance in diagnosis.

3. Using our FoT framework, we design an end-to-end diagnostic pipeline that mimics expert clinician workflow and constructed 2K dataset for supervised fine-tuning, and further refine reasoning via GRPO-based reinforcement learning.

4. Through a two-stage training approach, we develop the medical reasoning LLM **Med-FoT** for real-world diagnosis . Two physicians judged our FoT rationale consistant with authentic clinical reasoning.

2 Related works

Medical LLMs Driven by the shortcomings of generic LLMs on real clinical questions, research has moved toward purpose-built medical models. Open-weight Med42-v2[**?**] now tops MedQA[**?**] while providing a fully reproducible recipe , and Llama-3 Meditron[**?**] matches these gains on MedQA[**?**] and PubMedQA[**?**] with an equally open pipeline. Compute-efficient variants-BioMistral[**?**] and OpenBioLLM-Llama-3-close[**?**] much of the gap using 2 B additional tokens, while Me-LLaMA[**?**] demonstrates similar benefits from moderate continual pre-training. Beyond benchmark scores, current work emphasises robustness, safety, and broader domain coverage. Med-PaLM-2/3 introduce specialised safety tuning and expert feedback loops to reduce hazardous recommendations at scale[**?** **?**]; Baichuan-M1[**?**] and PMC-LLaMA[**?**] extend coverage to under-represented specialties via 20 T-token from-scratch training; multilingual initiatives such as HuatuoGPT-o1[**?**] and BioMistral-Multilingual[**?**] broaden access to non-English clinical practice; and long-context adaptations (e.g., DeepSeek-R1[**?**]) show that efficient attention plus prompt design can retain accuracy across thousands-token notes. Prompt-only routes remain attractive: AutoMedPrompt[**?**] automatically tunes system prompts via text-gradients for zero-compute adaptation. In contrast, our approach emphasizes enabling LLMs to excel in medical reasoning, offering a distinct solution.

Reasoning in LLMs Chain-of-Thought (CoT) prompting boosts clinical text reasoning accu-rac[], yet collecting large expert-annotated chains is costly for complex case[]. Model-generated rationales screened by medical verifiers ease this burden but still degrade as case complexity rise. Reinforcement-learning variants-e.g. RL with verifiable rewards and preference optimisation-can align reasoning without explicit labels, though they incur high compute and sparse-reward hurdle[]. Recent token-efficient prompts such as Chain-of-Draft and Chain-of-Preference Optimisation retain gains while halving onte cost[]. Reflective techniques-including self-reflection, self-consistency voting, and internalised self-correction-lower hallucination rates in dagnostic QA[], but depend on reliable automatic scorers. Self-training pipelines that distil verified chains into smaller models show promise for low-resource specialties[], while retrieval-augmented reflection further stabilises long-note inerence[]. Despite these advances, fully domain-aligned pipelines that mirror physician workflows reman scarce[]. To bridge this gap, we propose a physician-inspired Flow-of-Thought workflow combined with GRPO-based alignment, providing clinically grounded reasoning without expensive expert supervision.

3 Methodology

In this section, we first formalize the complex diagnostic problem and describe how we assemble and process our clinical dataset. We then introduce our agentic workflow and the reasoning data generation pipeline. Finally, we use Group Relative Policy Optimization to refine model reasoning through reinforcement learning.

3.1 Preliminaries: Real-World Clinical Diagnosis Problems

Problem Formalization In contrast to normal closed-set medical Q&A benchmarks, we craft diagnostic tasks that include comprehensive patient-level clinical information, requiring the model to identify the most probable disease y within the candidate set $\mathcal{D}_{dis}$. Where $\mathcal{D}_{dis}$ denote the set of 15 abdominal disease labels $\mathcal{D}_{dis} = \{d_1, \ldots, d_{15}\}$. As illustrated in Figure **??**, we first retrieve standardized diagnostic guidelines, then integrate the patient's history (H), physical examination (PE), laboratory test (Lab), and radiology reports (Rad) to predict the most probable disease $y \in \mathcal{D}_{dis}$. Each patient case is represented as follows:

$$x \;=\; (S,\; I) \quad \text{with} \quad I = \big\{H,\; PE,\; Lab,\; Rad\big\}, \tag{1}$$

where $S = \{S_k\}_{k=1}^{15}$ constitutes the diagnostic-criteria library. Each criterion S_k for disease d_k comprises the four components of information I.

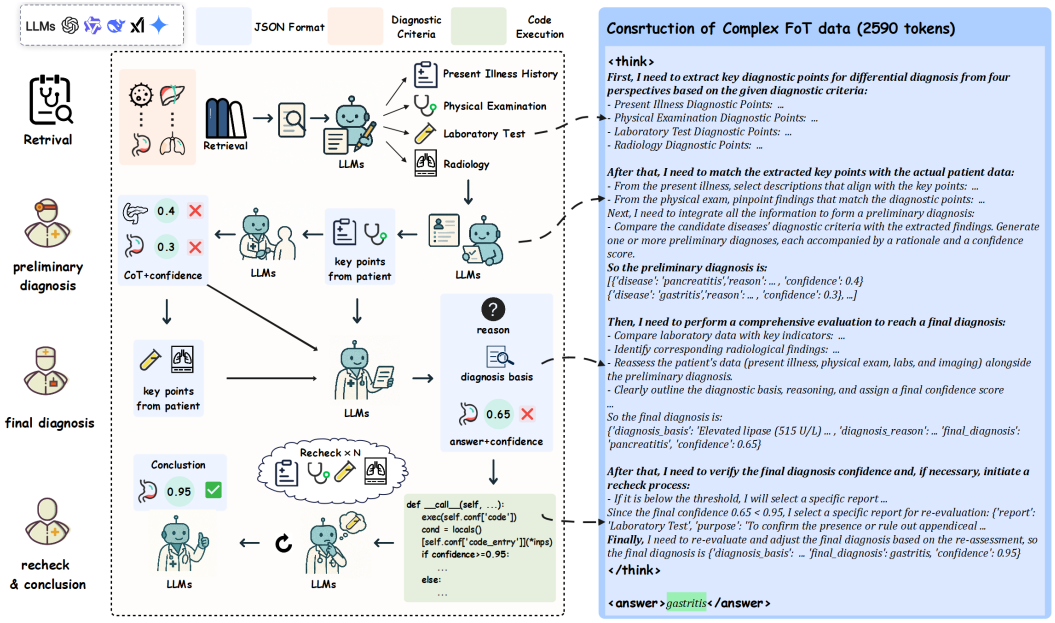

Figure 2: **Overview of FoT Data Generation Pipeline. Left:** We present a structured pipeline comprising the retrieval module, the initial diagnosis module, the final diagnosis module, and the recheck and conclusion module. **Right:** The instantiated diagnostic example.

Data Collection & Process To achieve this, we leverage de-identified, real-world clinical data to ensure both authenticity and scalability. Specifically, we collect 5K patient cases from the MIMIC-IV database[**?**], each labeled with one of 15 abdominal pathologies from the electronic health record (EHR) system. Some of these cases have incomplete information (e.g., lack of patient medical history). In addition, some questions are not suitable due to they may lack a unique correct answer or are too simple to require reasoning with comprehensive information. To address this, we first aggregate all ICD-9/10 codes and retain only digestive-system codes with at least 500 occurrences. Next, We harvest hospital admissions by matching these codes via regex and extract the full EHR records for each admission. In addition, we keep cases whose primary discharge diagnosis matches the target pathology and remove any cases annotated with multiple different disease labels.

The code used for filtering and processing can be found in Appendix. After this filtering and processing, we ultimately construct a dataset of 4K real-world clinical diagnostic cases denoted as $\mathcal{D} = \{(x, \hat{y})\}_{i=1}^{N}$, where $\hat{y}$ is the ground-truth disease from $\mathcal{D}_{dis}$.

3.2 FoT Generation and SFT

3.2.1 FoT Overview

The core of FoT is an agentic workflow that exploits LLMs to simulate tailored, high-efficiency medical reasoning pipelines. We formalize FoT as a directed graph comprising three categories of workflow nodes and seven distinct operations. The complete node set is denoted by $\mathcal{F} = \{N_1, N_2, \ldots, N_s\}$, where each node N_i belongs to one of the operations in the set $\mathcal{N}$:

- Model M: The model node (such as LLM) generates a response using a given prompt, connecting input data to the outputs.
- Tools T: Including RAG node R, code node C and web serach node W, each responsible for a specific function.
- Logic L: Logic node controls the transitions of workflow, including branch node B (directs flow based on conditions), for loop node F (iterates through data individually).

FoT supports simulation of tree, graph, network and other structures by defining node relationships with edges.

3.2.2 FoT generation Pipeline

This section details how the FoT framework instantiates reasoning trajectories aligned with canonical clinical decision pathways. The existing methods for generating long chains of thought (LongCoT) - including knowledge distillation and multi-agent coordination - remain unstable and clinically unreliable. Inspired by the routine diagnostic practice of physicians, we instead cast the procedure as a flexible workflow that coordinates multiple LLMs, thus producing reliable LongCoT demonstrations while preserving the agent's reasoning autonomy.

As shown in the Figure **??**, we divide the data generation pipeline into four parts, namely retrieval, initial diagnosis, final diagnosis, recheck and conclusion. All LLM nodes output in JSON format.

Retrieval In this stage, we employ Retrieval-Augmented Generation (RAG) to retrieve the diagnostic criteria for 15 disease categories. Two physicians independently prepared diagnostic guidelines for each disease in four modalities: medical history, physical examination, laboratory tests, and imaging studies. The aggregated retrieval results, denoted as S, are then encoded as key–value pairs according to the diseases. thereby constructing a "working memory" for inference. In this way, we hope the model can lay a traceable data foundation for the long-term thinking chain.

Preliminary Diagnosis Since emergency physicians rapidly form a preliminary diagnosis based on physical examination (PE) and patient history (H), we similarly cue a LLM to perform an initial assessment. We prompt the LLM to summarize key diagnostic points from the retrieved guidelines: *"Diagnostic criteria... Based on the diagnostic criteria given above, list key diagnostic points..."*, yielding a set of Diagnostic Points P. Then LLM maps diagnostic criteria to observed clinical features. From this alignment, we extract the patient's salient diagnostic information F_k for each part $k \in \{H, PE, Lab, Rad\}$. The LLM uses F_k to conduct a coarse-grained screening over the disease set $\mathcal{D}_{dis}$, producing a set of preliminary candidate diagnoses y_{pre}. Each candidate is accompanied by an explanatory rationale and a confidence score p_i. This enables rapid localization of the most likely etiology under limited information and guiding subsequent diagnostic testing.

Final Diagnosis In the final diagnosis phase, After the patient obtains further laboratory test (Lab) and imaging (Rad) results, we instruct the LLM to make a diagnosis based on comprehensive evidence. Specifically, we prompt LLM with:*"Diagnostic criteria:... please think carefully and give the final diagnosis results."*We define the final diagnosis using the following formula:

$$(y_{final}, R_f, B_f, C_f) = \text{LLM}(S, H, PE, Lab, Rad, , y_{pre}) \tag{2}$$

where LLM integrates all evidence chains$H$, PE, Lab, Rad to get a refined diagnosis y_{final}, simultaneously gives the diagnosis basis C_f, rationale R_f and global confidence C_f.

Recheck and Conclusion Before writing the final conclusion, physicians drill down into any uncertain details,triggering a back-verification loop. Specifically, if the final diagnosis confidence C_f (evaluated by the code execution node) falls below the threshold of 0.95, the loop node enables the LLM to select and recheck one of the four information sources (e.g., radiology) with the diagnostic criteria.The process iterates until C_f is verified as correct or is given up to $N = 3$ attempts. If all attempts fail or the eventual diagnosis remains incorrect, the case is discarded and moved to the test set. Once a successful flow is found, it is reformatted into a coherent language reasoning process.

3.3 Enhance Reasoning with RL

Following the development of foundational diagnostic reasoning abilities, we apply reinforcement learning (RL) to advance complex inference capabilities. After supervised fine-tuning (SFT), the large language model successfully analyzes four patient information components using diagnostic evidence, though the resulting reasoning paths may be suboptimal. We adopt the Group Relative Policy Optimization (GRPO[**?**]) algorithm for strategy optimization, employing a reward function to enhance the accuracy and generalization of diagnostic reasoning.

GRPO objective The Group Relative Policy Optimization (GRPO) objective can be written as:

$$J(\theta) = \mathbb{E}_{q\sim P(Q),\{o_i\}}\left[\frac{1}{G}\sum_{i=1}^{G}\min\big(r_i A_i,\ \text{clip}(r_i, 1-\epsilon, 1+\epsilon)A_i\big)\ - \beta\, D_{\text{KL}}\big(\pi_\theta \| \pi_{\text{ref}}\big)\right]. \tag{3}$$

Algorithm 1 Applying Large Language Models to Advanced Medical Reasoning

Definition: Complex diagnostic task $\mathcal{D} = (x, \hat{y})$, LLM module, RAG module R, branch node B, loop node F, code execution node C, medical input $I = \{H,\ PE,\ Lab,\ Rad\}$, candidate diseases $\mathcal{D}_{dis}$, diagnostic criteria S, max retries N, policy π_θ

$D_{\mathrm{SFT}} \leftarrow \emptyset$
for all $(x, \hat{y}) \in \mathcal{D}$ **do**
  $S \leftarrow F(\mathcal{D}_{dis}, R)$
  $P \leftarrow \mathrm{LLM}(S)$ ▷ Get Diagnostic Points
  for all $k \in [H, PE, Lab, Rad]$ **do**
    $F_k \leftarrow \mathrm{LLM}(k, P)$ ▷ Extract Findings
  end for
  $y_{pre} \leftarrow \mathrm{LLM}(S, H, PE, F_H, F_{PE})$ ▷ Preliminary Diagnosis
  $y_{final} \leftarrow \mathrm{LLM}(S, Lab, Rad, F_{Lab}, F_{Rad}, y_{pre})$ ▷ Final Diagnosis
  while $i < N$ **do**
    if $C(y_{final}[\text{confidence}] \geq 0.95)$ **then**
      $y \leftarrow y_{final}[\text{diagnosis}]$ ▷ Exit with high-confidence diagnosis
      if $y = \hat{y}$ **then**
        $D_{\mathrm{SFT}} \leftarrow (x, y_{pre}, y_{final}, S, H, PE, Lab, Rad, F_H, F_{PE}, F_{Lab}, F_{Rad})$
      end if
      break
    else
      $r_i \leftarrow \mathrm{LLM}(S, y_{final})$ ▷ Select Recheck information
      $y_{final} \leftarrow \mathrm{LLM}(S, y_{final}, r_i)$ ▷ Rediagnose
    end if
  end while
end for

Where $r_i = \frac{\pi_\theta(o_i|q)}{\pi_{\theta_{\mathrm{old}}}(o_i|q)}$, and its corresponding advantage estimate A_i. The clipping mechanism limits extreme updates by capping r_i within $[1-\epsilon, 1+\epsilon]$. A KL-divergence term with weight β penalizes large deviations from the reference policy π_{ref}.

Rewards Definition Let c be a generated completion and $\hat{y}$ its ground truth answer. We follow [**?** **?**] and define two binary reward functions. First, we prompt model to output the think process in "<think>...</think>" tags and its final answer in "<answer>...</answer>" tags. We verify that both tags appear and are used correctly. If so, the model earns a reward of 1.0. This encourages clear, well-structured outputs. Let $\hat{a}(c)$ denote the answer inside the "<answer>...</answer>"). We define the accuracy reward as follows:

$$R_{\mathrm{acc}}(c, \hat{y}) = \mathbb{I}\big[\hat{a}(c) = y^*\big] = \begin{cases} 1, & \text{if } \hat{a}(c) = \hat{y} \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

4 Experiments

Datasets According to **?**], we extracted 4000 diagnostic cases covering 15 abdominal disease categories from MIMIC-IV [**?**]. Using our FoT framework, we selected 2331 cases with correct reasoning trajectories for supervised fine-tuning (SFT) from QwQ-32B and reserved 3678 cases for reinforcement learning (RL). We strictly separate the wrong cases with reasoning errors (i.e., highly challenging examples including 13 diseases) from the training set and designate ten percent of the total as the test sets . Furthermore, consistent with prior work [**?**], we incorporated the original closed-question dataset (providing only disease labels) to enhance generalization.

Implementation Details We use the data constructed by FoT pipeline to train our model Med-FoT-LLaMA-8B and Med-FoT-Qwen-7B based on *LLaMA-3.1-8B-Instruct* and *Qwen-2.5-7B-Instruct* respectively.All experiments are run on a single node with $8x$A100(80GB) GPUs. In SFT stage, we down-weight the loss on all tokens preceding the <think> tag by a factor of 0.02, thereby mitigating the optimization challenges inherent in learning long chain-of-thought sequences. In addition, we use LoRA to train 4 epochs with a learning rate of 1.0e-4 and per-GPU batch size 1(2-step gradient

Table 1: **Main Results on Medical diagnosis.** LLMs with 🩺 are fine-tuned for the medical domain, and ⚡ means LLMs are trained for long chain-of-thought reasoning. Meanwhile, **bold** highlights the best performance.

Model	Appendicitis	Cholecystitis	Diverticulitis	Pancreatitis	Hepatitis	Pyelonephritis	Cholangitis	Peritonitis	Gastritis	Esophagitis	Duodenitis	Cystitis	Enteritis	Mean
Closed-Source Model														
gpt-o4-mini	**97.6**	85.0	64.5	72.1	54.5	80.0	32.5	79.3	44.0	27.3	14.3	50.0	68.8	67.1
Claude-3.5-sonnet-20241022	92.7	**83.8**	58.1	60.5	88.6	70.0	55.0	48.3	40.0	9.1	14.3	75.0	43.8	66.2
Grok-3-Beta	95.1	91.3	**61.3**	62.8	72.7	93.3	62.5	75.9	48.0	9.1	14.3	62.5	43.8	71.6
Deepseek-R1	90.2	82.5	58.1	69.8	79.5	76.7	**70.0**	79.3	48.0	9.1	14.3	25.0	62.5	70.3
∼8B Open-Source Model														
🩺 BioMistral-7B	46.3	13.8	16.1	9.3	15.9	3.3	7.5	3.4	0.0	0.0	0.0	12.5	0.0	12.7
🩺 OpenBioLLM-8B	34.1	11.3	29.0	2.3	6.8	0.0	2.5	6.9	0.0	0.0	0.0	0.0	6.3	9.8
🩺 UltraMedical-8B	78.0	45.0	41.9	53.5	45.5	30.0	15.0	27.6	24.0	9.1	0.0	0.0	6.3	38.0
🩺 ⚡ HuatuoGPT-o1-8B	90.2	71.3	54.8	62.8	56.8	53.3	52.5	51.7	28.0	0.0	14.3	12.5	25.0	55.9
🩺 ⚡ MedReason-8B	82.9	76.3	58.1	34.9	50.0	43.3	62.5	31.0	32.0	**18.2**	14.3	25.0	12.5	52.2
Mistral-7B-Instruct	82.9	80.0	41.9	51.2	50.0	36.7	17.5	20.7	12.0	18.2	0.0	25.0	31.3	46.8
Yi-1.5-9B	92.7	50.0	48.4	41.9	36.4	40.0	17.5	37.9	24.0	0.0	14.3	25.0	31.3	41.9
InternLM2.5-7B	82.9	80.0	41.9	51.2	50.0	36.7	17.5	20.7	12.0	18.2	0.0	25.0	31.3	46.8
LLaMA-3.1-8B-Instruct	85.4	73.8	61.3	53.5	40.9	40.0	12.5	27.6	48.0	0.0	0.0	37.5	12.5	48.0
Qwen2.5-7B-Instruct	97.6	73.8	41.9	48.8	59.1	30.0	22.5	48.3	28.0	0.0	28.6	12.5	12.5	50.0
DeepSeek-R1-Distill-LLaMA-8B	85.4	66.3	74.2	27.9	50.0	43.3	42.5	34.5	36.0	0.0	0.0	37.5	12.5	48.8
Qwen3-8B	90.2	73.8	71.0	74.4	63.6	66.7	37.5	51.7	64.0	0.0	14.3	37.5	25.0	62.0
>30B Open-Source Model														
🩺 ⚡ Citrus1.0-llama-70B	97.6	88.8	54.8	65.1	56.8	73.3	40.0	51.7	40.0	9.1	14.3	37.5	25.0	62.3
🩺 ⚡ HuatuoGPT-o1-70B	95.1	82.5	54.8	60.5	54.5	46.7	52.5	62.1	44.0	27.3	14.3	50.0	37.5	61.5
LLaMA-3.1-70B-Instruct	82.9	73.8	58.1	55.8	47.7	56.7	27.5	31.0	40.0	9.1	14.3	50.0	25.0	52.5
Qwen3-32B	92.7	82.5	67.7	55.8	65.9	76.7	50.0	62.1	48.0	0.0	14.3	62.5	43.8	65.0
Qwen2.5-72B-Instruct	95.1	76.3	61.3	46.5	72.7	70.0	35.0	34.5	52.0	18.2	14.3	37.5	6.3	58.1
QwQ-32B	92.7	78.8	64.5	60.5	63.6	73.3	42.5	69.0	56.0	18.2	14.3	62.5	62.5	65.2
DeepSeek-R1-Distill-LLaMA-70B	92.7	70.0	64.5	55.8	61.4	53.3	35.0	44.8	48.0	9.1	14.3	37.5	25.0	56.4
🩺 ⚡ Med-FoT-LLaMA-7B	97.6	88.8	61.3	76.7	65.9	93.3	57.5	65.5	28.0	0.0	0.0	25.0	43.8	68.4
🩺 ⚡ Med-FoT-Qwen-7B	95.1	87.5	64.5	65.1	65.9	73.3	55.0	79.3	32.0	0.0	0.0	12.5	37.5	66.2

accumulation). In RL stage, We use GRPO to train 2 epoch with full parameter tuning. We follow the configuration of [**?**]. We set the learning rate to 5e-7 for LLaMA and 1e-6 for Qwen. Following [**?**], the KL divergence coefficient β is set to 0.04 by default. The GRPO policy generates eight candidate rationales per sample with a maximum completion length of 8192 tokens.

Baselines & Evaluatoin Metric We compare our Med-FoT models with three types of baselines: **(1) Closed-Source Models:** GPT-o4-Mini [**?**], Claude-3.5-Sonnet [**?**], Grok-3-Beta [**?**], DeepSeek-R1 [**?**]; **(2) General Open-Source Models:** Mistral-Instruct [**?**], InternLM [**?**], Yi [**?**], LLaMA-3.1-Instruct [**?**], Qwen-2.5-Instruct [**?**], Qwen3, Citrus-llama; **(3) Medical-Specific Open-Source Models:** BioMistral [**?**], OpenBioLLM [**?**], UltraMedical [**?**], HuatuoGPT-o1 [**?**], and MedReason [**?**]. We evaluate model performance by diagnostic accuracy on each of the 15 abdominal disease categories and by the macro-average across all categories (Mean).

4.1 Experimental Results

Main Results As shown in Table **??**. We evaluated Med-FoT on a challenging 13-category abdominal disease diagnostic test set and compared it against state-of-the-art closed-source and open-source models.

Table 2: Zero-shot performance of various LLM under Original and FoT frameworks, "w/" means "with"

Model	Average		Appendicitis		Cholecystitis		Diverticulitis		Pancreatitis	
	Original	w/ FoT	Original	w/ FoT	Original	w/ FoT	Original	w/ FoT	Original	w/ FoT
DeepSeek-R1-Distill-LLaMA-70B	66.1	93.9 (+27.8)	96.2	99.2 (+3.0)	50.2	93.2 (+43.0)	49.8	90.7 (+40.9)	68.2	86.8 (+18.6)
LLaMA-3.3-70B-Instruct	83.9	93.0 (+9.1)	97.9	98.9 (+1.0)	82.6	92.7 (+10.1)	72.0	89.9 (+17.9)	82.9	84.4 (+1.5)
LLaMA-3.3-70B-Instruct (4-bit)	77.7	92.5 (+14.8)	97.5	98.9 (+1.4)	76.4	92.1 (+15.7)	68.9	88.7 (+19.8)	78.1	83.3 (+5.2)
Gemma-2-27B	81.0	86.7 (+5.7)	98.1	95.9 (-2.2)	86.7	88.0 (+1.3)	69.6	78.6 (+9.0)	82.9	72.5 (-10.4)

We first compare Med-FoT with medical-specialized open-source models. We cam observe that Citrus1.0-llama-70B (62.3%) and UltraMedical-8B (38.0%) lag by 4.6 and 28.9 percentage points, respectively. Even medical reasoning models such as MedReason-8B (52.2%) and HuatuoGPT-o1-8B (55.9%) are outperformed by 14.7 and 11.0 points. These discrepancies indicate that domain-specific pretraining or naive CoT fine-tuning alone is insufficient for complex multi-disease diagnostics. In addition, in comparison with general open-source reasoning models, Med-FoT exceeds DeepSeek-R1-Distill-LLaMA-70B (56.4%), QwQ-32B (65.0%), and Qwen3-32B (65.0%) by 10.5, 1.9, and 1.9 percentage points, respectively. This performance underscores the parameter efficiency and robustness of our approach, validating the effectiveness of structured Flow-of-Thought alignment and RL fine-tuning for small-scale models tackling complex medical reasoning tasks. Additionally, we compare Med-FoT-LLaMA-7B with closed-source baselines including gpt-o4-mini (67.1%), Claude-3.5-sonnet-20241022 (66.2%), and Grok-3-Beta (71.6%), achieves a mean accuracy of 66.9%, trailing only gpt-o4-mini by 0.2%. It demonstrates that our model can approximate the reasoning capacity of much larger. For example, on Appendicitis, Med-FoT-LLaMA-7B attains 100.0% accuracy, matching the top closed-source benchmark.

Training-free adoption in LLMs Following our Main Results, we evaluate two distinct training-free inference protocols on the abdominal disease test sets: (1) the standard zero-shot prompt-based approach, where each model directly generates a diagnosis from the input prompt; and (2) our agentic FoT pipeline, which orchestrates the model through a sequence of intermediate reasoning steps before arriving at a final decision.

We restrict this comparison to the four diseases tracked by the MIMIC-CDM leaderboard[**?**] to maintain consistency with prior evaluations. As shown in Table **??**, the prompt-based strategy (Original) yields an average accuracy of 66.1% for DeepSeek-R1-Distill-LLaMA-70B, 83.9% for LLaMA-3.3-70B-Instruct, 77.7% for its 4-bit , and 81.0% for Gemma-2-27B. When applying our FoT framework (w/ FoT), each model's accuracy improves substantially—for example, DeepSeek-R1 jumps from 66.1% to 93.9%, and LLaMA-3.3-Instruct from 83.9% to 93.0%.

These results demonstrate that the FoT pipeline effectively compensates for the limitations of direct prompt-based inference by structuring the model's reasoning into discrete, self-verifiable steps. Unlike naive prompt engineering, our agentic workflow supplies intermediate diagnostics and checks, leading to more accurate and robust predictions. Consequently, the FoT framework offers a general, model-agnostic mechanism to elevate zero-shot performance on complex, multi-disease diagnostic tasks without additional per-disease fine-tuning.

4.2 Ablation Study and Expert evaluation

Effect of FoT & RL We perform ablation experiments on *LLaMA-3.1-8B-Instruct* and *Qwen2.5-7B-Instruct* as baselines. In Table **??**, we first study the effect of our CoT data. The baseline LLaMA and Qwen models achieve mean accuracies of only 48.0% and 50.0%, respectively. Fine-tuning with CoT demonstrations increases accuracy by 17.2 and 12.3 percentage points respectively and demonstrates that explicit flow-of-thought supervision significantly improves diagnostic performance. In addition, we compare the impact of RL without FoT. Applying RL alone yields a comparable improvement but reduces the average number of generated tokens. When combining FoT data with RL, accuracy rises further by 1.7 and 4.9 points on top of RL alone. This observation suggests that RL refines the FoT-driven reasoning trajectories. These findings emphasize that while RL can independently shape effective reasoning, longer reasoning chains offer richer deliberations that allow the model to discover higher-reward solutions.

Table 3: The results of ablation experiments on Med-FoT. **Bold** highlights the best performance, "w" and "w/o" denote "with" and "without".

Model	Appendicitis	Cholecystitis	Diverticulitis	Pancreatitis	Hepatitis	Pyelonephritis	Cholangitis	Peritonitis	Gastritis	Esophagitis	Duodenitis	Cystitis	Enteritis	Mean	Avg Tokens
LLaMA Series															
LLaMA-3.1-8B-Instruct	85.4	73.8	61.3	53.5	40.9	40.0	12.5	27.6	48.0	0.0	0.0	37.5	12.5	48.0	824
SFT w/CoT	100.0	86.3	61.3	67.4	77.3	90.0	32.5	48.3	44.0	0.0	0.0	37.5	37.5	65.2 (+17.2)	3111
RL	97.6	85.0	74.2	93.0	50.0	86.7	52.5	62.1	28.0	9.1	33.3	0.0	31.3	66.7 (+18.7)	131
SFT w/CoT + RL	97.6	88.8	61.3	76.7	65.9	93.3	57.5	65.5	28.0	0.0	0.0	25.0	43.8	68.4 (+20.4)	962
Qwen Series															
Qwen2.5-7B-Instruct	97.6	73.8	41.9	48.8	59.1	30.0	22.5	48.3	28.0	0.0	33.3	12.5	12.5	50.0	386
SFT w/CoT	95.1	83.8	64.5	62.8	75.0	60.0	52.5	55.2	20.0	0.0	33.3	12.5	37.5	62.3 (+12.3)	2339
RL	95.1	82.5	51.6	74.4	68.2	63.3	45.0	55.2	16.0	0.0	33.3	12.5	18.8	60.3 (+10.3)	288
SFT w/CoT + RL	97.6	85.0	61.3	74.4	70.5	70.0	37.5	51.7	44.0	18.2	66.7	25.0	37.5	65.2 (+15.2)	804

Effect of RAG We evaluate the impact of Retrieval-Augmented Generation (RAG) within our Flow-of-Thought framework and observe a pronounced, model-size-dependent improvement. Models with fewer than 8 billion parameters benefit most: for instance, LLaMA-3.2-3B improves from 40 % to 48 %, and Qwen-2.5-7B from 55 % to 62 %. This demonstrates that RAG effectively supplements the limited internal knowledge of smaller models. By contrast, large models, such as DeepSeek-R1-Distill-Llama-70B and Llama-3.3-70B-Instruct, exhibit only marginal gains, as their scale already provides substantial medical reasoning capability. These findings confirm that RAG serves as a lightweight, generalizable augmentation strategy that narrows the performance gap on complex diagnostic tasks for compact LLMs.

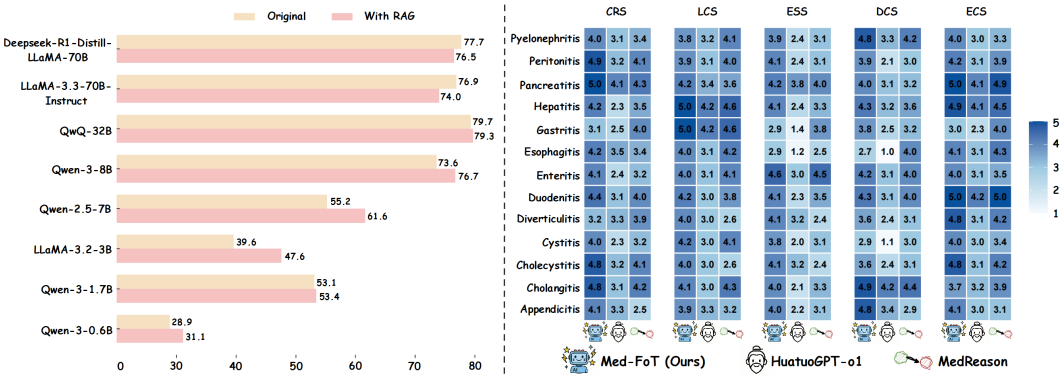

Figure 3: **Left:** The performance of different models with and without RAG. **Right:** Expert assessment on CoT data quality, comparing Med-FoT with MedReason and HuatuoGPT-o1.

Expert Assessment To evaluate Med-FoT's reasoning accuracy and soundness, we invite a gastroenterologist with 20 years of diagnostic experience to conduct a blind review of 65 cases (five cases for each of 13 disease categories). The expert rates Med-FoT, HuatuoGPT-o1, and MedReason on five dimensions: Clinical Relevance Score (CRS), Logical Coherence Score (LCS), Evidence Support Score (ESS), Differential Coverage Score (DCS), and Explanation Clarity Score (ECS), using integer values from 1 (poor) to 5 (excellent). As shown in Figure **??**, Med-FoT achieves average scores of 4.2, 4.2, 4.0, 4.0, and 4.3, outperforming HuatuoGPT-o1 (2.9, 3.1, 2.2, 2.6, 3.2) and MedReason (3.4, 3.5, 3.1, 3.2, 4.2). These results underscore Med-FoT's superior trustworthiness and clinical utility.

5 Conclusion

This paper proposes Flow-of-Thought (FoT), a structured agentic framework that automatically generates high-quality medical chain-of-thought data. We then apply a two-stage training pro-

cess—supervised fine-tuning (SFT) followed by GRPO-based reinforcement learning—to produce Med-FoT. Med-FoT tackles real-world clinical diagnostic challenges by simulating authentic diagnostic pathways. Experiments show that Med-FoT performs well on real-world clinical tests, rivaling much larger LLMs, and experts confirm the high quality of its reasoning. We hope our work will inspire further exploration of real-world medical problems and drive the advancement of medical AI.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction accurately reflect the paper's scope and contributions, clearly presenting both theoretical and experimental innovations in line with the results detailed later.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the limitations in Appendix.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: Our paper is based on solid experimental results and does not involve theoretical outcomes.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The information needed to reproduce the main experimental results is introduced in Experiments and Appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code and data will be publicly available.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: training and test details can be seen in Appendix.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: The paper indicates that the results are derived from multiple experimental runs.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Sufficient information on the computer resources can be seen in Section **??**.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Authors have reviewed the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: The paper discusses both the potential positive societal impacts and negative societal impacts of the work in Appendix.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: Our paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the original publication that introduced the code package, dataset, or model.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not employ crowdsourcing methods nor involve research with human participants.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: Our paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.

- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: The paper describes the use of LLMs as a core component for data construction, detailing their implementation and corresponding prompts in the Section **??** and Appendix.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.