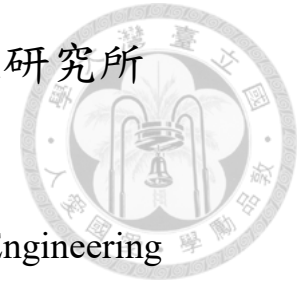


國立臺灣大學電機資訊學院資訊工程研究所

碩士論文

Department of Computer Science and Information Engineering

College of Electrical Engineering and Computer Science

National Taiwan University

Master's Thesis

大型語言模型進行醫療診斷推理

Large Language Models Perform Diagnostic Reasoning

吳承光

Cheng-Kuang Wu

指導教授: 陳信希 博士

Advisor: Hsin-Hsi Chen Ph.D.

中華民國 113 年 8 月

August, 2024

doi:10.6342/NTU202403801

國立臺灣大學碩士學位論文
口試委員會審定書
MASTER'S THESIS ACCEPTANCE CERTIFICATE
NATIONAL TAIWAN UNIVERSITY

大型語言模型進行醫療診斷推理

Large Language Models Perform Diagnostic Reasoning

本論文係吳承光君（學號 R10922186）在國立臺灣大學資訊工程學系完成之碩士學位論文，於民國 113 年 7 月 8 日承下列考試委員審查通過及口試及格，特此證明。

The undersigned, appointed by the Department of Computer Science and Information Engineering on 8 July 2024 have examined a Master's thesis entitled above presented by CHENG-KUANG WU (student ID: R10922186) candidate and hereby certify that it is worthy of acceptance.

口試委員 Oral examination committee:

陳信希　　張嘉惠　　[illegible]

（指導教授 Advisor）

[illegible]

系主任/所長 Director: 陳祝嵩

Acknowledgements

過去兩年半的碩士生活，是我目前人生中第一個，也是我認為相當重要的一個轉捩點。這是一段收穫豐盛的旅程，它讓我了解到，做研究與探索未知是多麼美好的一件事；它讓我意識到我真正擅長、有興趣的人事物，讓我更貼近自己的本質；最重要的是，它讓我認識了許多生命中的貴人，不論是教授、學長姐，或甚至一同在實驗室激盪點子、規劃實驗、寫程式到撰寫論文，最後飛越半個地球一起到非洲參加頂尖學術會議的好戰友。我除了感謝自己當初的勇氣，更想感謝自己的幸運，感謝這旅程中所有對我有過幫助的人們。

首先要感謝的是我的指導教授 ── 陳信希博士，老師給予了我做研究上極大的自由、支持與信任，讓我對自己的研究幾乎具有百分之百的掌控。印象最深的一次，是當我的好同伴陳韋霖和我討論到一個新研究點子，我們非常興奮的要開始寫程式、做實驗時，卻發現計算下來實驗費用不斐 ── 當時的大型語言模型收費還是相當昂貴的，和今日不可同日而語 ── 但老師您在聽了我們的點子後，卻完全信任我們，超級爽快的就給了我們做實驗的預算，完全沒有任何遲疑。您也是一位用心指導，同時卻又非常照顧學生感受的老師，不論是我們遭受實驗上的失敗、呈現表達上的失誤、甚至投稿的挫折後，除了給出建議之外，每次都不會忘了加上勉勵、安慰的話語。只能說，能遇到一位如此適合自己獨立個性的指導教授，是我的福氣。

doi:10.6342/NTU202403801

接著，我要感謝我的好戰友陳韋霖，你真的是一位非常有思想、才華、執行力，但同時又極度謙虛的朋友。沒有你，不會有人和我在禮拜五晚上，自願額外開線上會議討論研究；不會有人能和我一起在想到新點子後，一起泡在實驗室規劃實驗、寫程式、寫論文；不會有一個瘋狂的夥伴，在聽了我關於結合醫生診斷推理於大型語言模型的新點子後，就馬上找到一個新的論文投稿標的 (ICLR 2023 Tiny Papers Track)，然後一起密集衝刺，一個人寫程式、另一個人寫論文瘋狂趕稿，然後在半夜四點收到論文被接受的消息後從床上跳起來恭喜彼此。最後，一起在二十幾歲的時候就飛越半個地球到非洲的盧安達 ── 一個連長榮的地勤都不知道怎麼掛行李的地方 ── 一起參加第一個辦在非洲的頂尖人工智慧會議。總而言之，少了這位好戰友，我無法想像自己能有本論文的研究成果。

我也要感謝一路上幫助過我，許許多多其他的師長、學長姐與夥伴。感謝王暉智老師，要不是當初您願意讓我在台大醫院實習時，擔任急診部 AI 計畫的研究助理，我不會認識陳信希老師的團隊，非常可能就不會進到目前的實驗室，認識許許多多優秀的夥伴們；感謝林禹廷、魏聖倫學長，從我還在醫學系兼任研究助理時，就一起在台大急診部 AI 計畫奮鬥的研究夥伴，讓我第一次進入電腦科學的世界；也再次感謝陳信希、王暉智與呂理駿老師，當初二話不說就願意幫我寫研究所入學的推薦信，您們的推薦至關重要。感謝顏安孜學姊、陳重吉學長與黃瀚萱學長，每次被我用網路通訊軟體請教問題後，都非常用心、詳細的回答我；感謝陳建宏學長、劉又慈，你們為實驗室的付出，大家都看在眼裡，少了你們，我們不會有這麼舒適、便利的實驗室環境；感謝陳柏君學長，在知道關於大型語言模型或最新研究的消息，都會馬上與我交流分享；再次感謝魏聖倫，願意主動在我碩士論文口試前幾天，額外在你原本就極度忙碌的行程中，騰出時間聽我報告；感謝張庭維，實驗室裡難得同樣具備醫學系背景的夥伴，願意無償的幫我做需要醫療專業知識的標記。我也相當感謝修課時好夥伴們，感謝劉鎮霆、黃昱翔、黃

doi:10.6342/NTU202403801

哲韋、任恬儀、李艾霓、邢皓霆、簡宏曄，在軟體工程的課程一起沒日沒夜的衝刺出期末專案，最後成為這堂課最高分的組別，甚至後來有些成為一起打球的好夥伴；感謝王睿誼與盧思宇，我會一直記得那些想了好幾天都 TLE 後，一起在教室白板絞盡腦汁、擠出線性時間演算法的時光。感謝林佑恩，你早早就看出了陳韋霖與我對研究的熱情，在非常早期就給了我們鼓勵、推我們一把，這個火苗我認為相當重要。

感謝口試委員鄭卜壬教授、張嘉惠教授以及王釧茹教授，於口試期間提供許多寶貴的提問與建議，補足了我自己的許多盲點，使本論文能夠更加完備，在此獻上深深的謝意。

最後，我要感謝我親近的家人、女朋友、自己與許許多多未列出的貴人們。感謝我的父母吳柾杰、蔡靜如，你們同樣給予了我極大的信任與自由，讓我完全無後顧之憂的追逐自己的興趣。感謝我非常體貼的女朋友黃薇霓，我知道自己有數不清的夜晚都在忙課業、忙工作，甚至連出去玩的時候還會帶著電腦，抓到空擋就開始寫程式、寫論文，排擠到了我們一起相處時的時間、精力。非常感謝妳願意為了我追逐夢想而包容，承受著離開醫師這個穩定工作的不確定性。我也要感謝自己的勇氣、毅力與努力，願意踏出這未知的一步，願意在床上想到演算法後，直接在半夜跳上書桌一路寫程式到天亮。感謝所有所有沒有列在本文章中的貴人們，感謝的心情、要感謝的人實在太多太多了。

這兩年多的經驗如此寶貴，任世界上所有的金錢都無法與其交換。

doi:10.6342/NTU202403801

摘要

醫療診斷推理是臨床工作中的重要能力，它讓醫師能從病人身上「蒐集關鍵資訊」，並且利用蒐集到的資訊「預測診斷」。本論文以問診作為研究案例，探討大型語言模型之診斷推理能力，並提出進一步提升此推理能力之方法論。此項研究之主要貢獻有三：一、發展大型語言模型角色扮演評估框架，用以評估大型語言模型之問診能力；二、提出少樣本、零樣本之提示工程方法論，此方法論結合醫師診斷推理之思考過程，並以實驗佐證其能提升大型語言模型「蒐集關鍵資訊」以及「預測診斷」之表現。三、顯示大型語言模型能透過儲存及擷取自生成之診斷推理過程，持續增進其診斷預測之能力。

關鍵字：大型語言模型、醫療診斷推理、問診

doi:10.6342/NTU202403801

Abstract

Medical diagnostic reasoning is a crucial capability in clinical practice, enabling physicians to “collect key information” from patients and “predict diagnoses” based on the collected information. This thesis investigates the diagnostic reasoning abilities of large language models (LLMs) using history taking as a case study and proposes methodologies to further enhance these reasoning abilities. The main contributions of this research are threefold: (1) Development of an LLM Role-Playing Evaluation Framework to assess the history-taking abilities of LLMs. (2) Introduction of few-shot and zero-shot prompting methodologies that integrate the diagnostic reasoning processes of physicians, with experimental evidence demonstrating their effectiveness in improving LLMs’ performance in “collecting key information” and “predicting diagnoses”. (3) Showing that LLMs can continuously improve their diagnostic prediction capabilities through storing and retrieving self-generated diagnostic reasoning processes.

Keywords: Large Language Models, Medical Diagnostic Reasoning, History Taking

doi:10.6342/NTU202403801

Contents

doi:10.6342/NTU202403801

doi:10.6342/NTU202403801

List of Figures

doi:10.6342/NTU202403801

List of Tables

doi:10.6342/NTU202403801

Chapter 1 Introduction

1.1 Background

In this work, we investigate the medical diagnostic reasoning [1] capabilities of large language models (LLMs). Diagnostic reasoning, extensively used by medical doctors in daily clinical practice, involves two major aspects. One is the ability to collect diagnostic information from patients, and the other is the ability to predict diagnoses based on the collected information. This study focuses on the task of history taking to evaluate LLMs' diagnostic reasoning abilities in these two aspects.

A typical history taking process is a multi-turn conversation between a doctor and a patient, divided into three stages. Initially, the doctor is informed of the chief complaint, the primary reason the patient seeks medical attention, which serves as the initial diagnostic information. In the second stage, the doctor engages in an iterative question-answering (QA) session, asking additional questions to gather more diagnostic information, as the chief complaint alone is insufficient for predicting the diagnosis. In this work, we refer to the diagnostic information as the patient's clinical findings. Finally, the doctor predicts the diagnosis based on the collected clinical findings. Making an accurate diagnosis in this scenario relies heavily on the quality of the gathered information, which requires asking critical questions. The ability to ask such questions depends on performing diagnostic rea-

doi:10.6342/NTU202403801

soning with medical knowledge, as the doctor must consider all possible diagnoses from the previously collected information to ask proper questions in the current dialogue turn.

1.2 Motivation

In clinical practice, history taking is the crucial first step in collecting diagnostic information from patients. According to previous literature, history taking alone provides sufficient information to make a diagnosis in approximately 75% of patient encounters before further examinations [2]. Therefore, when done appropriately, effective history taking can significantly reduce the cost of unnecessary medical tests.

The field of developing dialogue systems for history taking has made significant progress in recent years. Researchers have developed various frameworks and model architectures to perform automatic diagnosis by simulating the history taking process. These advancements primarily fall into two research branches. One branch consists of reinforcement learning (RL)-based methods, which use deep Q-learning or policy gradients to train fully-connected neural networks for asking questions about clinical findings [3–7]. The other branch leverages Transformer-based architectures, where history taking is formulated either as a sequence generation problem or a multi-label classification task [8–10].

Despite their promising applicability, these methods share several limitations:

- Training data requirements: These works require training on medical conversational datasets, which are costly and time-consuming to construct.
- Fixed action space: These models assume a fixed set of clinical findings as their action space, which does not align with the unbounded nature of real-world questions.

doi:10.6342/NTU202403801

- Unrealistic patient simulators: They evaluate model performance by interacting with a lookup-table-like patient that responds in a non-natural language manner, deviating from real clinical scenarios.

On the other hand, LLMs can be used out-of-the-box to solve tasks in fluent natural language without fine-tuning on specific datasets, thanks to their strong few-shot in-context learning [11] and instruction-following [12] abilities. Furthermore, previous studies have shown that LLMs are capable of passing the United States Medical Licensing Examination (USMLE) through proper prompting techniques [13]. Given this evidence, we are curious about LLMs' potential diagnostic reasoning capabilities as-is, without any fine-tuning. This curiosity leads us to the following research questions: ***RQ1:*** *How well can LLMs perform history taking out-of-the-box?* ***RQ2:*** *How can we elicit LLMs' diagnostic reasoning abilities to further improve them?*

In summary, this study seeks to evaluate and enhance LLMs' diagnostic reasoning capabilities, focusing on two key aspects: the ability to *collect diagnostic information* and the ability to *predict the diagnosis*. Our contributions are summarized as follows:

- We develop an LLM-Role-Playing Evaluation Framework to evaluate LLMs' performance in history taking.
- We propose few-shot and zero-shot prompting methodologies that integrate medical doctors' diagnostic reasoning processes, demonstrating improvements in LLMs' ability to collect diagnostic information and predict diagnoses.
- We show that LLMs can continuously improve over time through a memory mechanism with self-generated diagnostic reasoning.

doi:10.6342/NTU202403801

1.3 Thesis Organization

This thesis is organized as follows: Chapter 1 introduces the task of history taking and the motivation of our study. Chapter 2 reviews related works on history taking and the reasoning abilities of large language models. Chapter 3 describes the dataset used in this work. Chapter 4 presents experiments on LLMs' abilities to collect diagnostic information. Chapter 5 discusses experiments on LLMs' abilities to predict diagnoses. Chapter 6 summarizes the major findings from the experiments and outlines future work.

doi:10.6342/NTU202403801

Chapter 2 Related Work

2.1 History Taking

Developing dialogue systems for history taking (i.e., automatic diagnosis) has recently gained significant interest in the research community. Tang et al. (2016) [3] formulates history taking as a Markov decision process (MDP) of an agent interacting with a patient, and describes the problem using RL terms. The action set of the agent is the union of predefined clinical findings and diagnoses, and they adopt deep Q-learning [14] to find the optimal policy. Kao et al. (2018) [4] make two enhancements on top of Tang et al. (2016) [3] to improve diagnosis accuracy. One is hierarchical reinforcement learning to make joint diagnostic decisions by introducing a latent layer using anatomical parts. The other is incorporating context-awareness into the model by considering patient demographics, temporal factors, and geographic location. Since the previously mentioned works [3, 4] use synthetic medical datasets, Liu et al. (2018) [5] construct the MZ dataset from real doctor-patient dialogues and train a deep Q-network (DQN) on MZ. Xu et al. (2019) [6] augment the DQN with prior knowledge by relation matrix and medical knowledge graph, and propose a new dataset called DX. Xia et al. (2020) [7] propose a new policy gradient framework and enhance the reward function with mutual information to encourage the model to ask the most discriminative clinical finding. Taking a step back,

doi:10.6342/NTU202403801

all these works [3–7] use the same problem formulation: they adopt an RL algorithm to learn the policy using fully-connected neural networks, where each neuron in the input layer encodes a clinical finding and each neuron in the output layer represents an action.

Due to the success of the Transformer [15], recent works have begun to approach history taking from perspectives other than RL by exploiting the strengths of this novel network architecture. Diaformer formulates history taking as a symptoms sequence generation problem [8]. MTDiag leverages the unordered nature of clinical findings and reformulates history taking as a multi-label classification task [9]. CoAD bridges the gap between training and inference by generating symptoms and diagnoses collaboratively [10].

Whether one chooses to adopt the RL-based methods [3–7] or methods that require fine-tuning Transformers [8–10], they all share the following limitations: (1) The models need to be trained on medical conversational datasets, which are costly to construct. (2) They assume a fixed number of clinical findings as the action space. However, in real-world scenarios, the actions are natural language questions and thus unbounded. (3) These works evaluate model performance by interacting with a lookup-table-like patient that responds in a non-natural language fashion, deviating from real clinical scenarios. We aim to address these limitations by exploring the possibility of directly leveraging the encoded clinical knowledge in LLMs [13] to conduct history taking in an end-to-end, natural language manner.

2.2 Reasoning Abilities of Large Language Models

Recent advancements in large-scale pretraining have led to several emergent abilities in language models. GPT-3 shows that scaling up language models enable them to perform

doi:10.6342/NTU202403801

in-context few-shot learning, which could achieve state-of-the-art performance without any gradient updates [11]. Wei et al. (2021) further shows that instruction tuning enhances LLMs' abilities to follow instructions, and substantially improves zero-shot performance on unseen tasks [12]. This leads to the paradigm shift where language models can be used out-of-the-box to solve problems without fine-tuning on the downstream tasks.

Although these works show that LLMs are able to achieve striking performance through prompting alone, they still struggle on tasks that require reasoning (e.g., arithmetic). For example, PaLM 540B only achieves the accuracy of 18% with standard few-shot prompting [16]. Therefore, several works try to elicit the reasoning capabilities in LLMs. Wei et al. (2022) show that augmenting the few-shot examples with chain of thought (CoT), which includes the intermediate reasoning steps for solving a problem, significantly improves LLMs' performance on arithmetic, commonsense, and symbolic reasoning tasks [16]. Kojima et al. (2022) extend the idea of CoT to the zero-shot setting, and show that such reasoning capabilities can be elicited simply by appending a trigger phrase (e.g., "Let's think step by step.") to the end of the prompt, discarding the need for constructing few-shot CoT examples [17]. However, these works only focus on arithmetic [18], commmonsense [19], and symbolic reasoning (e.g., last letter concatenation and coin flip) tasks.

In this work, we extend the CoT framework to medical diagnostic reasoning [1], and investigate whether LLMs can benefit from medical doctors' intermediate reasoning steps. Specifically, we explore the task of history taking [20], which requires the models to apply extensive medical knowledge in a multi-turn conversation.

doi:10.6342/NTU202403801

Chapter 3 Datasets

We adopt the DDXPlus dataset [21] for our experiments. DDXPlus is a large-scale dataset with extensive coverage of clinical findings and diagnoses for the task of history taking. Each instance in DDXPlus is a patient profile, which contains the chief complaint, clinical findings (i.e., evidence), and the ground truth diagnosis of the patient. Below, we introduce the contents of a patient profile and provide basic statistics of DDXPlus.

3.1 Patient Profile

Sex: M, Age: 49

Diagnosis: Allergic sinusitis

Clinical findings:

- My nose or the back of my throat is itchy.

- I have severe itching in one or both eyes.

- I have nasal congestion.

- I am coughing.

- Some family members suffer from allergies, hay fever or eczema.

- Some family members have asthma.

The block above shows the content of a simple patient profile in DDXPlus. The

doi:10.6342/NTU202403801

profile contains the chief complaint (highlighted in blue here), clinical findings, and diagnosis. The number of clinical findings in each patient profile varies.

3.2 Data Statistics

DDXPlus contains roughly 1.3 million patient profiles, each synthesized based on the distribution of age, sex, geographical region, clinical findings, and diagnosis according to a proprietary medical knowledge base (KB) [21]. The synthesis is verified by medical doctors to ensure the quality. The patient profiles are divided into training, validation, and test sets based on an 80%-10%-10% split. We use the test set for our experiments. Overall, DDXPlus encompass 223 different clinical findings and 49 diagnoses, and there are 13.56 clinical findings in a patient profile on average.

doi:10.6342/NTU202403801

Chapter 4 Collecting Diagnostic Information

In this chapter, we investigate LLMs' ability to collect diagnostic information from patients. To evaluate this ability, the following two aspects must be addressed:

1. A patient is needed to interact with the LLM to be evaluated.
2. A metric is required to assess the quality of the collected information.

To address these two points, we propose (1) an LLM-Role-Playing Evaluation Framework, and (2) the use of diagnostic accuracy at each dialogue turn as the assessment metric.

4.1 The LLM-Role-Playing Evaluation Framework

Inspired by the concept of "standardized patients" in medical education—where trained individuals read and memorize patient profiles to accurately answer medical students' questions during history taking—we propose a novel evaluation framework. In this framework, one LLM assumes the role of the patient, while another LLM takes on the role of the doctor. Figure 4.1 illustrates the overview of the proposed framework.

doi:10.6342/NTU202403801

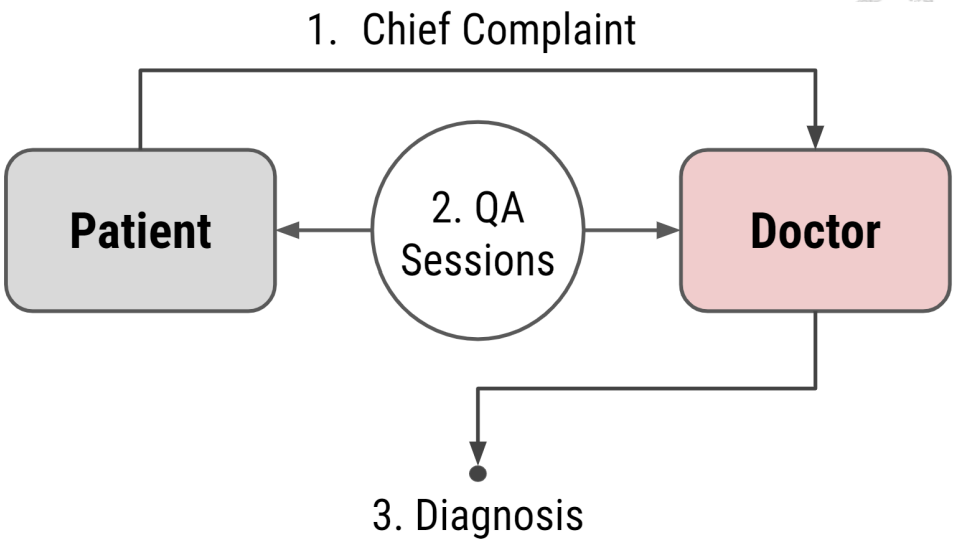


Figure 4.1: Overview of the LLM-Role-Playing Evaluation Framework.

Doctor. The doctor LLM is denoted as $f(\cdot \mid \theta)$, where θ is the LLM's network parameters. When the doctor LLM asks a question to collect diagnostic information, its output can be denoted as $q_i = f(p_{ask}(\cdot) \mid \theta)$, where q_i is the question asked at the i-th dialogue turn and p_{ask} is the prompting template for asking questions.

Patient. The patient LLM is denoted as $g(\cdot \mid \theta)$, where θ is the LLM's network parameters. During implementation, the LLM used for the doctor or the patient could be either the same or different. When the patient LLM answers the doctor LLM's question q_i, its output is represented as $a_i = g(p_{res}(q_i, P) \mid \theta)$, where p_{res} is the prompting template for responding to questions and P the patient profile. The instructions in p_{res} specify that the LLM should respond to q_i faithfully based on the content of P.

Dialogue history. We represent a dialogue of ℓ turns as $H = [(a_0), (q_1, a_1), (q_2, a_2)..., (q_\ell, a_\ell)]$, where a_i denotes the patient's utterance and q_i denotes the doctor's utterance. Specifically, a_0 is the chief complaint of the patient, q_i represents the doctor's question, and a_i denotes the patient's response to q_i. The maximum dialogue turn is set to 8 for all experiments.

doi:10.6342/NTU202403801

4.2 Evaluation Metric

To evaluate LLMs' ability to collect diagnostic information, we use diagnostic accuracy at each dialogue turn as the assessment metric. The rationale behind this metric is straightforward: the higher the quality of the collected information, the more accurately LLMs can predict the diagnosis.

In this framework, we denote the diagnosis set on DDXPlus as D and the dialogue history up to turn ℓ as H. The diagnostic prediction at turn ℓ is represented as $\hat{y}_\ell = f(p_{make}(D, H) \mid \theta)$, where p_{make} is the prompting template for making a diagnosis and θ represents the LLM's network parameters. As the dialogue progresses, the collected information in H grows. The higher the quality of the collected information, the more rapidly the diagnostic accuracy will increase as ℓ goes up. To see what the accuracy curves look like, please refer to Figure 4.2 and Figure 4.4. The experiment details are introduced in the section below, which are included in our previously published work [22].

4.3 Experiments

4.3.1 Few-Shot Setting

4.3.1.1 Methodology

Inspired by few-shot bot [23], we propose the following methods for the doctor LLM. We implement these methods with InstructGPT [24], using the *text-davinci-003* endpoint. The notations used below are consistent with the ones mentioned in section 4.1.

doi:10.6342/NTU202403801

Baseline. We demonstrate k few-shot examples of doctor-patient dialogues in the prompt. Formally, the output is $q_i = f(p_{ask}(H_1, ..., H_k, H_{cur}) \mid \theta)$, where $H_1, ..., H_k$ are k few-shot dialogue examples ($H = [(a_0), (q_1, a_1), (q_2, a_2)..., (q_\ell, a_\ell)]$) and H_{cur} is the current dialogue history. We set $k = 2$ in our experiments.

DR-CoT. Inspired by the evidence that CoT prompting elicits LLMs' reasoning in arithmetic, commonsense, and symbolic reasoning tasks [16, 17], we hypothesize that it is also possible to elicit LLMs' diagnostic reasoning abilities by demonstrating medical doctors' reasoning steps in the few-shot examples. We call this approach Diagnostic-Reasoning CoT (DR-CoT). Specifically, we augment the few-shot examples $H_1, ..., H_k$ with intermediate reasoning steps r_i as follows: $H = [(a_0), (r_1, q_1, a_1), (r_2, q_2, a_2)..., (r_\ell, q_\ell, a_\ell)]$. There are three intermediate reasoning steps in r_i: *positive clinical findings*, *negative clinical findings*, and *differential diagnosis*. Note that the only difference in $H_1, ..., H_k$ between the baseline and DR-CoT is r_i, with a_0 and (q_i, a_i) being completely the same.

4.3.1.2 Results

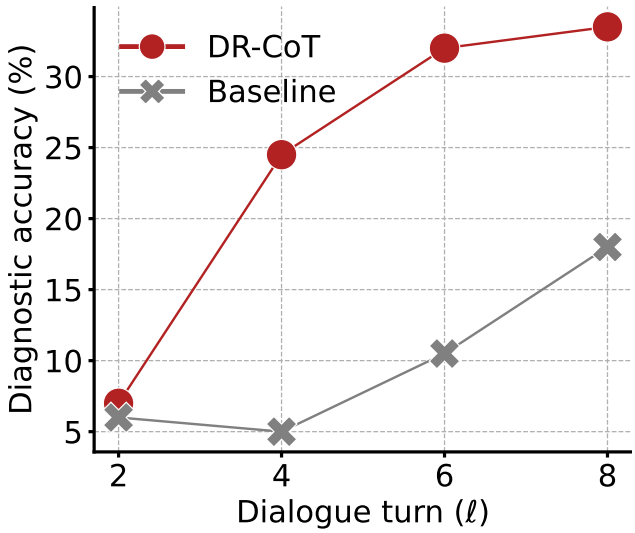


Figure 4.2: Diagnostic accuracy of *text-davinci-003* at each dialogue turn (ℓ = 2, 4, 6, 8).

doi:10.6342/NTU202403801

Diagnostic accuracy at each dialogue turn. Figure 4.2 shows the accuracy curves of baseline and DR-CoT. There are two findings in this figure. First, we can enable LLMs to collect useful diagnostic information simply by demonstrating them $k = 2$ dialogue examples, as shown by the accuracy curve of the baseline method. Secondly, the efficiency of collecting diagnostic information can be dramatically increased by incorporating medical doctors' diagnostic reasoning steps into dialogue examples, as shown by the accuracy curve of DR-CoT. The performance boost in diagnostic accuracy can be +15.5% to +21.5% depending on the dialogue turns.

Out-domain experiments. To investigate the robustness of DR-CoT, we conduct out-domain (OD) experiments. Specifically, we use different chief complaint a_0 in $H_1, ..., H_k$ and H_{cur} to ensure that the doctor LLM doesn't just use the memorized reasoning in few-shot examples for solving H_{cur}. The results are shown in Figure 4.3, where we show the diagnostic accuracies at the 8th dialogue turn. Although diagnostic accuracies drop for out-domain chief complaints, the performance boost of DR-CoT over the baseline remains, which are +18.0% and +14.0% in OD1 and OD2, respectively.

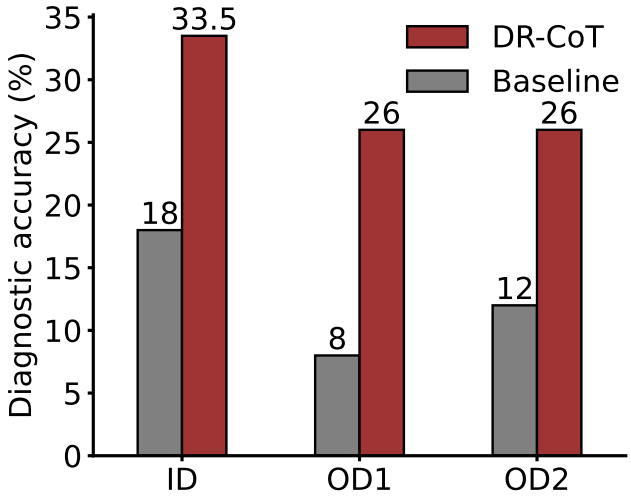
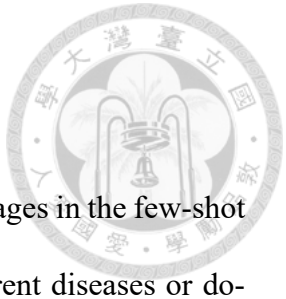


Figure 4.3: Diagnostic accuracy in out-domain experiments with *text-davinci-003*. Chief complaints: ID = cough; OD1 = shortness of breath; OD2 = nasal congestion.

doi:10.6342/NTU202403801

4.3.2 Zero-Shot Setting



Despite the effectiveness of DR-CoT, there are several disadvantages in the few-shot methods. Firstly, constructing few-shot dialogue examples for different diseases or domains is still costly, as the reasoning steps require the expertise of medical doctors. Additionally, the few-shot examples are not simply (x, y) pairs in the task of history taking, but rather full dialogue histories in the format of $H = [(a_0), (r_1, q_1, a_1), (r_2, q_2, a_2)..., (r_\ell, q_\ell, a_\ell)]$, making the construction process even more resource-intensive. Secondly, the inference cost increases as the number of shots scales up. This leads to higher computational expenses and longer processing times. Given these challenges, we explore whether it is possible to elicit diagnostic reasoning in LLMs in a zero-shot setting, which eliminates the need for few-shot examples.

4.3.2.1 Methodology

Baseline. The standard zero-shot baseline involves generating questions based solely on the current dialogue history without any examples. Formally, the output question at the i-th dialogue turn is given by: $q_i = f(p_{zeroshot}(H_{cur}) \mid \theta)$, where $p_{zeroshot}$ is the prompting template for zero-shot question generation and H_{cur} is the current dialogue history.

ZS-DRCoT. In the original DR-CoT method, the doctor LLM generates questions based on few-shot examples that include intermediate reasoning steps. Formally, this is represented as $q_i = f(p_{ask}(H_1, ..., H_k, H_{cur}) \mid \theta)$, where each example H is structured as $H = [(a_0), (r_1, q_1, a_1), (r_2, q_2, a_2)..., (r_\ell, q_\ell, a_\ell)]$. In the zero-shot DR-CoT (ZS-DRCoT) method, instead of using few-shot examples, the model generates questions by leveraging

doi:10.6342/NTU202403801

a zero-shot prompting template that incorporates the reasoning structure directly into the prompt. The output question at the i-th dialogue turn is given by $q_i = f(p_{zs-drcot}(H_{cur}) \mid \theta)$. Here, the reasoning steps are provided as a structured JSON format within $p_{zs-drcot}$ rather than through few-shot examples $H_1, ..., H_k$. The JSON format includes the three intermediate reasoning steps as:

```
{
  “positive_clinical_findings”: [],
  “negative_clinical_findings”: [],
  “ranked_differential_diagnosis”: [],
  “question_to_ask”: “”,
}
```

This approach offers several advantages over the few-shot DR-CoT:

- **Cost-effectiveness:** It eliminates the need for costly and time-consuming construction of few-shot dialogue examples.
- **Scalability:** The zero-shot method can be easily applied across various diseases and domains without additional example generation in that specific domain.

4.3.2.2 Results

Diagnostic accuracy at each dialogue turn. Figure 4.4 shows the diagnostic accuracy of *gpt-3.5-turbo-0125* at different dialogue turns $\ell = 0, 2, 4, 6, 8$. There are two findings from this figure:

1. The zero-shot baseline can already collect useful diagnostic information, as indicated by the increasing accuracy on the curve. This demonstrates that even without

doi:10.6342/NTU202403801

few-shot examples, the baseline model can effectively gather relevant clinical findings over successive dialogue turns.

2. The ZS-DRCoT method consistently outperforms the baseline across all dialogue turns. This indicates that incorporating the reasoning structure directly into the zero-shot prompt significantly enhances the model's ability to collect diagnostic information.

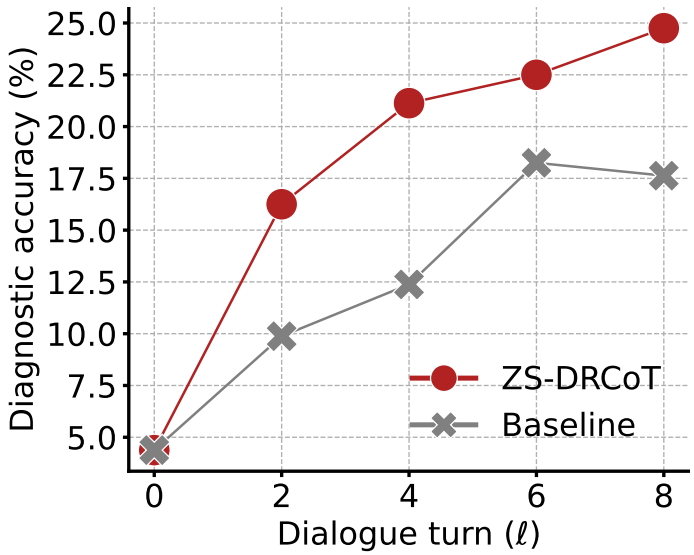


Figure 4.4: Diagnostic accuracy of *gpt-3.5-turbo-0125* at each dialogue turn (ℓ = 0, 2, 4, 6, 8) in the zero-shot setting.

Generalizability across different LLMs. To evaluate the generalizability of the ZS-DRCoT approach, we tested it across different LLM families, including GPT, Gemini, Claude, Mixtral, and Llama3. As shown in Table 4.1, the ZS-DRCoT method consistently outperforms the baseline across all tested models. Specifically, ZS-DRCoT achieved an average diagnostic accuracy boost of +5.48% in absolute terms and +28.39% in relative terms compared to the baseline. These results demonstrate that the ZS-DRCoT method is effective across a variety of LLMs, highlighting its applicability in diverse settings.

doi:10.6342/NTU202403801

Method/LLM	GPT	Gemini	Claude	Mixtral	Llama3	Average
Baseline	17.63	22.38	18.13	21.13	18.88	19.63
ZS-DRCoT	**24.75**	**28.25**	**25.63**	**25.63**	**21.25**	**25.10**

Table 4.1: Performance across different methods and LLMs (GPT: *gpt-3.5-turbo-0125*; Gemini: *gemini-1.0-pro-001*; Claude: *claude-3-haiku-20240307*; Mixtral: *mixtral-8x22b*; Llama3: *llama-3-70b*).

Robustness across chief complaints. To evaluate the robustness of ZS-DRCoT over the baseline, we decompose the performance boost in Table 4.1 with four different chief complaints: cough, fever, nasal congestion, and shortness of breath. The differences in performance between ZS-DRCoT and the baseline are shown in Table 4.2. The results show that ZS-DRCoT is robust for cough, fever, and nasal congestion across most LLMs, demonstrating improvements over the baseline. However, for the chief complaint of shortness of breath, ZS-DRCoT shows a decline in performance, particularly with GPT and Llama3. This decline suggests that the current zero-shot diagnostic reasoning methodology may not adequately address the complexity or variability associated with the chief complaint of shortness of breath. Further investigation is needed to understand the underlying reasons and to develop methods to enhance robustness across all types of chief complaints, which we leave for future work.

Chief complaint/LLM	GPT	Gemini	Claude	Llama3	Mixtral	Average
Cough	+20.00	+4.50	+10.50	+4.50	+11.00	+10.10
Fever	+14.50	+5.00	+8.50	+5.50	+6.00	+7.90
Nasal congestion	**-1.00**	+12.50	+13.00	+10.00	**-4.00**	+6.10
Shortness of breath	**-5.00**	+1.50	**-2.00**	**-10.50**	+5.00	**-2.20**
All	+7.13	+5.88	+7.50	+2.38	+4.50	+5.48

Table 4.2: Performance difference between ZS-DRCoT and standard zero-shot baseline.

doi:10.6342/NTU202403801

Ablations on Intermediate Reasoning Steps. In ZS-DRCoT, three intermediate reasoning steps are specified: *positive clinical findings*, *negative clinical findings*, and *differential diagnosis*. To understand the contributions of these steps to the performance boost of DR-CoT, we conducted ablation studies to evaluate the impact of removing each reasoning step. The results are shown in Table 4.3. On average, all intermediate reasoning steps contribute to the performance boost, but their contributions vary:

- Removing positive clinical findings leads to the largest drop in performance across all LLMs, highlighting its critical role in diagnostic reasoning.
- Differential diagnosis also shows a notable decrease in performance when removed.
- While the absence of negative clinical findings has a varied impact across different LLMs, its removal still results in a small performance decrease on average.

Ablation/LLM	GPT	Gemini	Claude	Average
ZS-DRCoT	0.00	0.00	0.00	0.00
- positive clinical findings	-6.00	-11.00	-6.00	-7.67
- negative clinical findings	+**2.50**	-8.50	0.00	-2.00
- differential diagnosis	-2.00	-6.50	-5.50	-4.67

Table 4.3: Ablations on ZS-DRCoT and change in diagnostic accuracy (%).

Scaling Trends on Number of Total Diagnoses. In real-world scenarios, the diagnosis set may vary in size depending on the application. Therefore, it is crucial to understand the scaling trends of performance improvement as he number of possible diagnoses increases. The full diagnosis set on DDXPlus contains 49 diagnoses. We evaluate LLMs' performance on diagnosis sets of 5, 10, 20, and 49 diagnoses.

Table 4.4 shows the absolute improvements in diagnostic accuracy achieved by ZS-DRCoT over the baseline for different numbers of diagnoses, while Figure 4.5 demon-

doi:10.6342/NTU202403801

strates the scaling curves of relative improvements in diagnostic accuracy. The absolute and relative improvement remains consistent or even increases as the label set scales up, especially for *gpt-3.5-turbo-0125*, demonstrating that ZS-DRCoT is effective across varying sizes of diagnosis sets. This scalability is particularly beneficial in real-world applications where the number of possible diagnoses can be large and diverse.

LLM/Number of Diagnoses	**5DDx**	**10DDx**	**20DDx**	**49DDx**
GPT				
Baseline	49.75	31.75	22.84	17.63
ZS-DRCoT	53.00	35.75	30.21	24.75
Absolute improvement	+3.25	+4.00	+7.37	+7.12
Gemini				
Baseline	54.50	37.25	23.53	22.38
ZS-DRCoT	62.00	46.75	29.78	28.25
Absolute improvement	+7.50	+9.50	+6.25	+5.87

Table 4.4: Absolute improvements of ZS-DRCoT over baseline in diagnostic accuracy (%) GPT: *gpt-3.5-turbo-0125*; Gemini: *gemini-1.0-pro-001*.

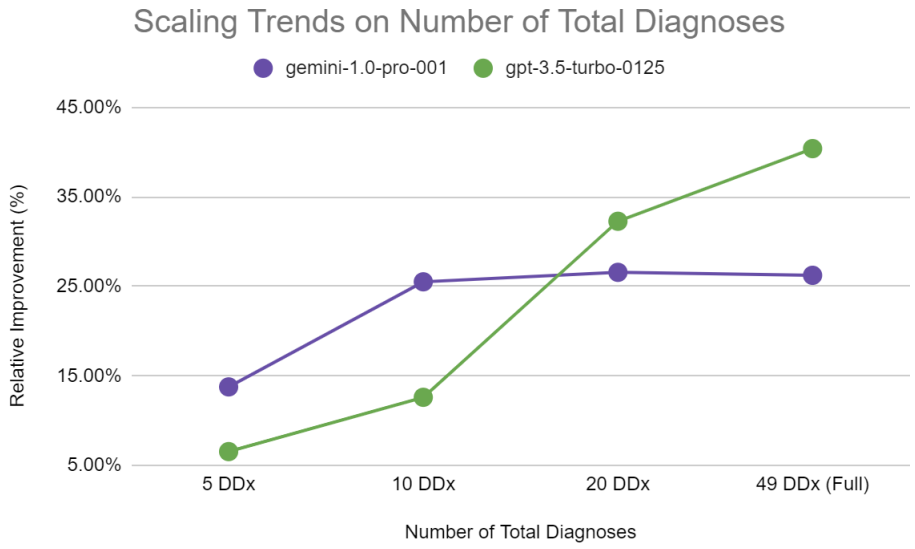


Figure 4.5: Relative improvements of diagnostic accuracy with different number of total diagnoses.

doi:10.6342/NTU202403801

Chapter 5 Diagnosis Prediction

In this chapter, we focus on LLMs' ability to predict the diagnosis based on collected information. Specifically, we evaluate LLMs' diagnostic accuracy on the patient profiles in DDXPlus [21].

5.1 General Setup

To ensure a fair comparison between LLMs and to focus solely on their diagnosis prediction ability, we adopt a slightly different problem setting from previous experiments, which primarily investigated LLMs' ability to collect diagnostic information.

In this new setup, we simulate an application scenario where the goal is to assist doctors in making diagnoses after the patient history has been taken. This approach disentangles the diagnosis prediction ability from the ability to collect information, allowing us to evaluate how well LLMs can infer the correct diagnosis from a given set of clinical findings. The process is as follows:

- The input x is the patient profile, which includes the patient's clinical findings without the ground truth diagnosis.
- The doctor LLM $f(\cdot \mid \theta)$ processes this information and predicts its diagnosis $\hat{y}$.

doi:10.6342/NTU202403801

5.2 Methodology

5.2.1 Non-streaming setting

In the non-streaming setting, the doctor LLM does not update itself in the clinical environment. The performance of the LLM solely depends on its inherent capabilities and the prompting methodology used. In this scenario, the LLM processes each patient profile independently without any mechanism for learning from previous encounters or feedback. This approach highlights the baseline diagnostic accuracy of the LLM based on its pre-trained knowledge and the prompts provided, such as the zero-shot baseline or our proposed ZS-DRCoT.

Baseline. In the zero-shot baseline setting, the LLM generates a diagnosis based on the current patient profile using zero-shot prompting. Formally, this can be represented as: $\hat{y} = f(p_{zeroshot}(x) \mid \theta)$, where x is the patient profile and $\hat{y}$ the predicted diagnosis.

ZS-DRCoT. In the ZS-DRCoT setting, the LLM generates the diagnostic reasoning before predicting the diagnosis. This approach aims to examine whether our previously proposed methodology is also beneficial for the LLMs' ability to predict the diagnosis. Formally, this can be represented as: $(\hat{r}, \hat{y}) = f(p_{zs-drcot}(x) \mid \theta)$, where $\hat{r}$ is the generated diagnostic reasoning and $\hat{y}$ is the predicted diagnosis.

doi:10.6342/NTU202403801

5.2.2 Streaming setting

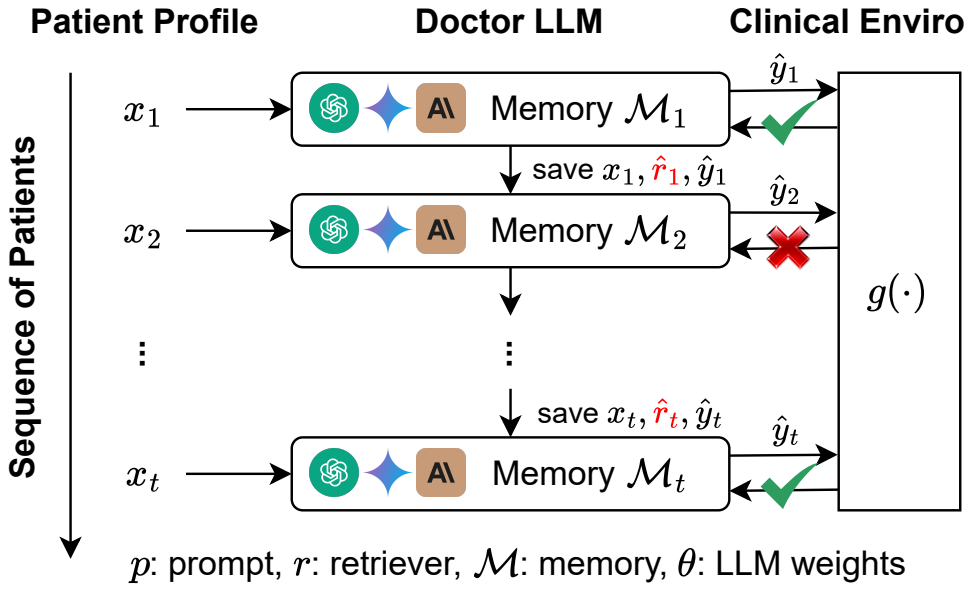


Figure 5.1: Continuous improvement of LLMs in the clinical environment through memory mechanism of self-generated diagnostic reasoning and predictions.

In the streaming setting [25], we explore how LLMs can continuously improve themselves in the clinical environment. Just as doctors enhance their diagnostic skills over time through patient encounters, we simulate a realistic clinical scenario where each time point denotes an exposure to a new patient profile. The process is shown in Figure 5.1, which involves the following steps at each time step i:

1. The doctor LLM processes a patient profile x_i and predict the diagnosis $\hat{y}_i$.

2. The clinical environment provides feedback on the correctness (v or x) of the diagnosis $\hat{y}_i$.

3. If the diagnosis $\hat{y}_i$ is correct, the LLM updates its memory $\mathcal{M}_i$ by saving the patient profile x_i, the self-generated rationale $\hat{r}_i$, and the prediction $\hat{y}_i$ to the memory.

doi:10.6342/NTU202403801

Single-Agent Memory. In the single-agent memory setting, a single LLM (doctor agent) uses its own memory to continuously improve its diagnostic predictions over time by the process detailed in Algorithm 1 with $K = 1$. The memory $\mathcal{M}$ is updated only when $\hat{y}$ is correct (i.e., $fb_t = 1$), since it is shown that incorrect examples degrade LLMs' performance in previous works [26, 27].

Multi-Agent Memory. The intuition behind using a multi-agent memory system is that each LLM has its own strengths, and leveraging multiple agents can potentially enhance overall performance. However, traditional multi-agent methods are often very expensive because they involve linear inference costs proportional to the number of agents ($O(k)$) [28, 29]. To address this issue, we propose a solution that involves sharing a common memory among the agents and using a round-robin scheduling approach. This method allows different LLMs to take turns in making predictions, effectively reducing the inference cost while still benefiting from the diverse strengths of multiple agents. The procedure is detailed in Algorithm 1. Experimental results in Section 5.3 show that this simplistic round-robin algorithm leads to a striking performance improvement.

Algorithm 1 Round-Robin Algorithm for Single- and Multi-Agent Memory Mechanisms

Initialize K doctor LLM agents $f_0(\cdot|\theta_0)$, $f_1(\cdot|\theta_1)$, ..., $f_{K-1}(\cdot|\theta_{K-1})$; ▷ K = 1 in Single-Agent Memory
Initialize prompt $p(\cdot)$, retriever $r(\cdot)$, and external memory $\mathcal{M}_0$, all shared between LLMs;
for $t = 1, 2, \ldots, T$ **do**
    Receive a patient profile x_t from the stream;
    Select the next agent by $k = t \bmod K$;
    The k-th agent predicts $\hat{r}_t, \hat{y}_t = f_k(p(x_t, r(\mathcal{M}_{t-1}))|\theta_k)$;
    Receive correctness signal $fb_t = g(x_t, \hat{y}_t)$; ▷ $fb_t \in \{0, 1\}$
    if $fb_t = 1$ **then** ▷ which means the self-output $\hat{y}_t$ is correct
        $\mathcal{M}_t \leftarrow \mathcal{M}_{t-1} \cup \{(x_t, \hat{r}_t, \hat{y}_t)\}$;
    else
        $\mathcal{M}_t \leftarrow \mathcal{M}_{t-1}$;
    end if
end for

doi:10.6342/NTU202403801

5.3 Experiments

5.3.1 Implementation

We conduct experiments using three different LLMs: GPT (*gpt-3.5-turbo-0125*), Gemini (*gemini-1.0-pro-001*), and Claude (*claude-3-haiku-20240307*). These models are cost-effective LLMs that balance performance and affordability. The three LLMs are used to initialize $K = 3$ agents in the multi-agent memory method. We implement memory $\mathcal{M}$ as a key-value vector database, and use `BAAI/bge-base-en-v1.5` to encode the patient profile x_t as the key embeddings.

5.3.2 Results

The experimental results reveal several significant findings:

- **In non-streaming setting, ZS-DRCoT > baseline:** Our ZS-DRCoT method demonstrates a superior ability to predict diagnoses compared to the baseline method. This indicates that the diagnostic reasoning elicited by ZS-DRCoT enhances not only the collection of diagnostic information but also the diagnostic prediction capability of LLMs.

- **Streaming methods > non-streaming methods:** The introduction of the memory mechanism in the streaming setting significantly improves diagnostic performance over the non-streaming methods. This suggests that the memory mechanism is effective in further eliciting the intrinsic capabilities of LLMs through self-generated outputs, allowing the models to adapt over time.

doi:10.6342/NTU202403801

- **ZS-DRCoT + memory > baseline + memory:** The combination of ZS-DRCoT with the memory mechanism outperforms the baseline with memory. This highlights the importance of saving self-generated rationales, as they provide valuable context and reasoning that enhance future predictions.

- **Multi-agent memory > single-agent memory:** The multi-agent memory approach yields a substantial performance boost compared to the single-agent memory, without incurring additional computational costs due to the round-robin algorithm. This "free bost" demonstrates the significant advantage of leveraging multiple LLMs in a shared memory setting.

In addition to Table 5.1, we also demonstrate the performance curves in Figure 5.2 for better visualization of continuous improvements.

Method/LLM	**GPT**	**Gemini**	**Claude**	**Average**
Non-streaming				
Baseline	47.56	50.57	60.43	52.85
ZS-DRCoT	53.18	59.30	62.13	58.20
Streaming				
Baseline + Single-agent memory	66.16	69.50	76.02	70.56
Baseline + Multi-agent memory	-	-	-	**83.50**
ZS-DRCoT + Single-agent memory	73.19	82.65	84.98	80.27
ZS-DRCoT + Multi-agent memory	-	-	-	**90.48**

Table 5.1: Non-streaming and streaming performance across different methods and LLMs (GPT: *gpt-3.5-turbo-0125*; Gemini: *gemini-1.0-pro-001*; Claude: *claude-3-haiku-20240307*).

doi:10.6342/NTU202403801

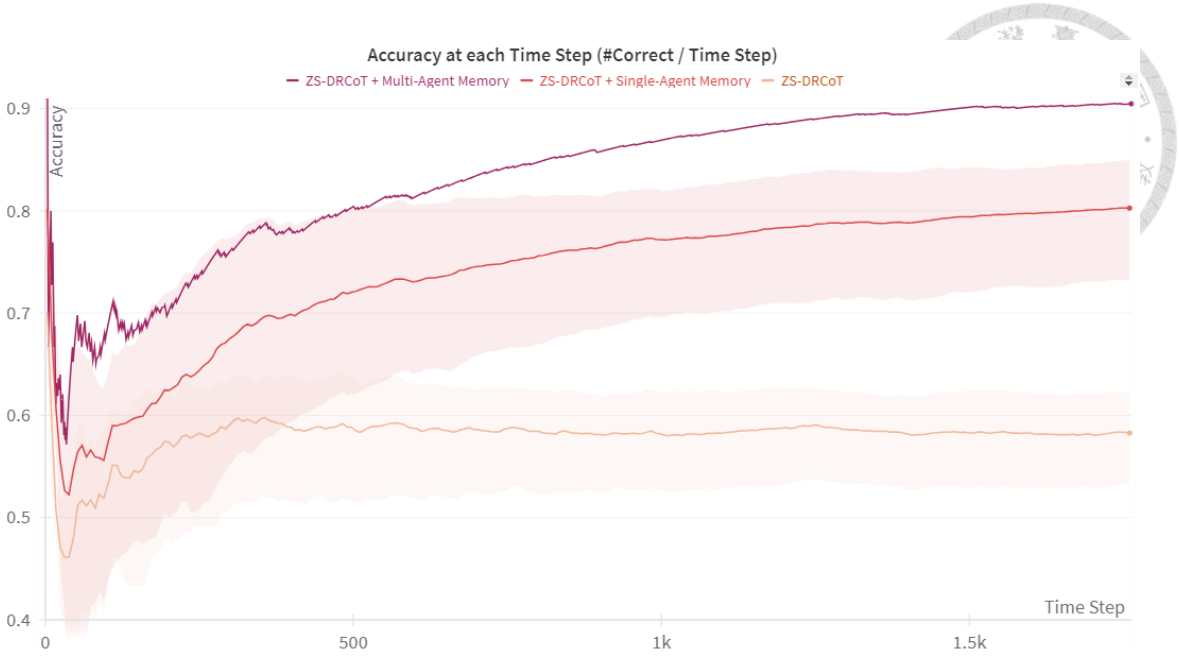


Figure 5.2: Performance curves of ZS-DRCoT, ZS-DRCoT + Single-Agent Memory, and ZS-DRCoT + Multi-Agent Memory. The shaded area represents the variance of three LLMs used in the experiments.

5.4 Discussion

5.4.1 Confusion Matrices of Single-Agent and Multi-Agent Memory Methods

To analyze why memory sharing works, we visualize the confusion matrices for a subset of diagnoses related to upper respiratory tract diseases. Figure 5.3 presents the confusion matrices for three different LLM agents: `gpt-3.5-turbo-0125`, `gemini-1.0-pro`, and `claude-3-haiku-20240307`, along with the matrix of Multi-Agent Memory (MAM). Each matrix illustrates the proficiency of an agent across various medical diagnosis categories. It is evident that each model excels in certain areas while struggling in others. For instance, `gpt-3.5-turbo-0125` shows high accuracy in predicting "acute rhinosinusitis" and "allergic sinusitis" but struggles with "chronic rhinosinusitis" and "URTI". In

doi:10.6342/NTU202403801

contrast, `gemini-1.0-pro` performs well in "URTI", and `claude-3-haiku` could solve "chronic rhinosinusitis". The diversity in performance across models suggests that their collective past experiences can provide complementary strengths, thereby enhancing overall performance when these experiences are shared.

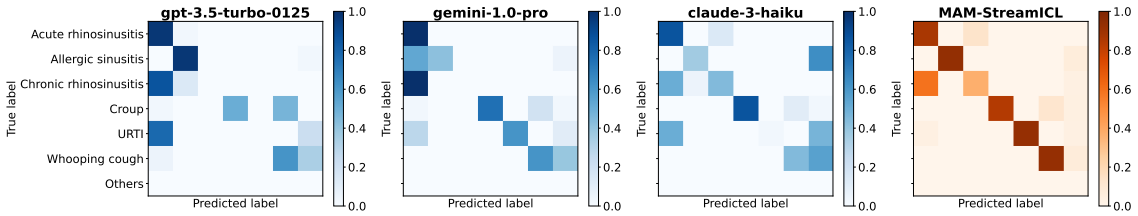


Figure 5.3: Confusion matrices of diagnoses subset of upper respiratory tract diseases.

5.4.2 Ablation Studies on Multi-Agent Memory

We conduct ablation studies using Algorithm 2 with the baseline to demonstrate the importance of multi-agent memory.

Algorithm 2 Ablated Version of Round-Robin Algorithm for Multi-Agent Memory

Initialize K doctor LLM agents $f_0(\cdot|\theta_0)$, $f_1(\cdot|\theta_1)$, ..., $f_{K-1}(\cdot|\theta_{K-1})$;
Initialize prompt $p(\cdot)$, retriever $r(\cdot)$, and external memory $\mathcal{M}_0$, all shared between agents;
Select an agent $f_s(\cdot|\theta_s)$ as the source of single-agent memory; ▷ For example, we can choose `gemini-1.0-pro-001`
for $t = 1, 2, \ldots, T$ **do**
    Receive a patient profile x_t from the stream;
    Select the next agent by $k = t \bmod K$;
    The k-th agent predicts $\hat{r}_t, \hat{y}_t = f_k(p(x_t, r(\mathcal{M}_{t-1}))|\theta_k)$; ▷ $\hat{y}_t$ is used to for evaluation
    The chosen single agent predicts $\hat{y}_{t_s} = f_s(p(x_t, r(\mathcal{M}_{t-1}))|\theta_s)$; ▷ Counterfactual ablation
    Receive feedback signal $fb_{t_s} = g(x_t, \hat{y}_{t_s})$; ▷ $\hat{y}_{t_s}$ is used for receiving feedback
    if $fb_{t_s} = 1$ **then** ▷ which means the self-output $\hat{y}_t$ is correct
        $\mathcal{M}_t \leftarrow \mathcal{M}_{t-1} \cup \{(x_t, \hat{y}_{t_s})\}$;
    else
        $\mathcal{M}_t \leftarrow \mathcal{M}_{t-1}$;
    end if
end for

doi:10.6342/NTU202403801

The parts different from the original multi-agent algorithm are highlighted in red. This ablated algorithm can be seen as a counterfactual experiment, where we use multiple agents for *inference* but only one chosen agent for the *memory* mechanism. The results in Table 5.2 demonstrate several key findings:

- **Multi-Agent Memory (Ablation):** When using multiple agents for inference but relying on a single agent for memory, we still observe an improvement over the baseline and single-agent memory setups. This suggests that even without a full multi-agent memory mechanism, multi-agent inference can contribute positively to performance.

- **Multi-Agent Memory:** The complete multi-agent memory setup, where both inference and memory are handled by multiple agents, provides the highest performance boost. This configuration leverages the strengths of multiple LLMs for both inference and memory, resulting in a substantial improvement in diagnostic accuracy.

These findings indicate that both multi-agent memory and multi-agent inference are beneficial for enhancing LLMs' diagnosis prediction ability.

Method	GPT	Gemini	Claude	Memory	Inference
Baseline	47.56	50.57	60.43	x	single agent
+ Single-Agent Memory	66.16	69.50	76.02	single agent	single agent
+ Multi-Agent Memory (ablation)	65.31	72.05	81.52	single agent	multi agent
+ Multi-Agent Memory	**83.50**	**83.50**	**83.50**	multi agent	multi agent

Table 5.2: Ablation studies with baseline + Multi-Agent Memory. The ablated version only uses memory of the single corresponding agent, while still uses round-robin algorithm for multi-agent inference. We can see that both multi-agent memory and inference are beneficial for performance boost. The detailed algorithm for this ablation study can be found in Algorithm 2.

doi:10.6342/NTU202403801

Chapter 6 Conclusions

In this work, we investigated the diagnostic reasoning capabilities of large language models (LLMs) in the context of medical history taking. We proposed several simple and purely in-context methods to improve LLMs' diagnostic reasoning abilities without any finetuning. Our major findings are that we can elicit LLMs' diagnostic reasoning abilities through medical doctors' reasoning steps by (1) few-shot demonstration; (2) zero-shot reasoning structure specification; and (3) LLM self-generated rationales and predictions to achieve striking performance boost. These findings demonstrate that LLMs possess considerable untapped reasoning potential that can be elicited through appropriate prompting techniques, even in specialized domains such as medical diagnosis.

Despite these advancements, there are still limitations and open research questions worth exploring:

- **Stopping Criterion in History Taking:** While our methods improve diagnostic reasoning, it remains unclear whether LLMs can effectively determine when they have collected sufficient information to stop the history-taking process. Future work could focus on developing mechanisms for LLMs to autonomously decide when to conclude the information-gathering phase.
- **Continuous Improvement in Question Asking:** Although we have proposed meth-

doi:10.6342/NTU202403801

ods for LLMs to enhance their diagnostic prediction abilities over time, achieving continuous improvement in their ability to ask pertinent questions remains a challenge. Further research could explore strategies for LLMs to refine their questioning techniques based on feedback and previous interactions.

- **Scaling and Maintenance of the Memory Pool:** The scalability and maintenance of the memory pool used in our multi-agent memory mechanism pose practical challenges. Future studies could investigate efficient ways to manage and scale the memory pool while ensuring the integrity and relevance of stored information.

In summary, this thesis provides a foundation for enhancing LLMs' diagnostic reasoning abilities through in-context learning methods. While significant progress has been made, continued research and development are necessary to fully realize the potential of LLMs in medical diagnosis and other specialized domains.

doi:10.6342/NTU202403801

References

[1] Jerome P Kassirer. Diagnostic reasoning. Annals of internal medicine, 110(11):893–900, 1989.

[2] Friedemann Ohm, Daniela Vogel, Susanne Sehner, Marjo Wijnen-Meijer, and Sigrid Harendza. Details acquired from medical history and patients' experience of empathy–two sides of the same coin. BMC medical education, 13(1):1–7, 2013.

[3] Kai-Fu Tang, Hao-Cheng Kao, Chun-Nan Chou, and Edward Y Chang. Inquire and diagnose: Neural symptom checking ensemble using deep reinforcement learning. In NIPS workshop on deep reinforcement learning, 2016.

[4] Hao-Cheng Kao, Kai-Fu Tang, and Edward Chang. Context-aware symptom checking for disease diagnosis using hierarchical reinforcement learning. In Proceedings of the AAAI conference on artificial intelligence, volume 32, 2018.

[5] Zhongyu Wei, Qianlong Liu, Baolin Peng, Huaixiao Tou, Ting Chen, Xuan-Jing Huang, Kam-Fai Wong, and Xiang Dai. Task-oriented dialogue system for automatic diagnosis. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 201–207, 2018.

[6] Lin Xu, Qixian Zhou, Ke Gong, Xiaodan Liang, Jianheng Tang, and Liang Lin. End-to-end knowledge-routed relational dialogue system for automatic diagnosis.

doi:10.6342/NTU202403801

In Proceedings of the AAAI conference on artificial intelligence, volume 33, pages 7346–7353, 2019.

[7] Yuan Xia, Jingbo Zhou, Zhenhui Shi, Chao Lu, and Haifeng Huang. Generative adversarial regularized mutual information policy gradient framework for automatic diagnosis. In Proceedings of the AAAI conference on artificial intelligence, volume 34, pages 1062–1069, 2020.

[8] Junying Chen, Dongfang Li, Qingcai Chen, Wenxiu Zhou, and Xin Liu. Diaformer: Automatic diagnosis via symptoms sequence generation. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 36, pages 4432–4440, 2022.

[9] Zhenyu Hou, Yukuo Cen, Ziding Liu, Dongxue Wu, Baoyan Wang, Xuanhe Li, Lei Hong, and Jie Tang. Mtdiag: an effective multi-task framework for automatic diagnosis. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 37, pages 14241–14248, 2023.

[10] Huimin Wang, Wai Chung Kwan, Kam-Fai Wong, and Yefeng Zheng. Coad: Automatic diagnosis through symptom and disease collaborative generation. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 6348–6361, 2023.

[11] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. Advances in neural information processing systems, 33:1877–1901, 2020.

[12] Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian

doi:10.6342/NTU202403801

Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. arXiv preprint arXiv:2109.01652, 2021.

[13] 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. arXiv preprint arXiv:2212.13138, 2022.

[14] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. arXiv preprint arXiv:1312.5602, 2013.

[15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. Advances in neural information processing systems, 30, 2017.

[16] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. Advances in neural information processing systems, 35:24824–24837, 2022.

[17] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. Advances in neural information processing systems, 35:22199–22213, 2022.

[18] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168, 2021.

doi:10.6342/NTU202403801

[19] Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. Transactions of the Association for Computational Linguistics, 9:346–361, 2021.

[20] Katharina E Keifenheim, Martin Teufel, Julianne Ip, Natalie Speiser, Elisabeth J Leehr, Stephan Zipfel, and Anne Herrmann-Werner. Teaching history taking to medical students: a systematic review. BMC medical education, 15:1–12, 2015.

[21] Arsene Fansi Tchango, Rishab Goel, Zhi Wen, Julien Martel, and Joumana Ghosn. Ddxplus: A new dataset for automatic medical diagnosis. Advances in neural information processing systems, 35:31306–31318, 2022.

[22] Cheng-Kuang Wu, Wei-Lin Chen, and Hsin-Hsi Chen. Large language models perform diagnostic reasoning. arXiv preprint arXiv:2307.08922, 2023.

[23] Andrea Madotto, Zhaojiang Lin, Genta Indra Winata, and Pascale Fung. Few-shot bot: Prompt-based learning for dialogue systems. arXiv preprint arXiv:2110.08118, 2021.

[24] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. Advances in neural information processing systems, 35:27730–27744, 2022.

[25] Cheng-Kuang Wu, Zhi Rui Tam, Chieh-Yen Lin, Yun-Nung Chen, and Hung-yi Lee. Streambench: Towards benchmarking continuous improvement of language agents. arXiv preprint arXiv:2406.08747, 2024.

doi:10.6342/NTU202403801

[26] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 11048–11064, 2022.

[27] Xinxi Lyu, Sewon Min, Iz Beltagy, Luke Zettlemoyer, and Hannaneh Hajishirzi. Z-icl: Zero-shot in-context learning with pseudo-demonstrations. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 2304–2317, 2023.

[28] Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. Improving factuality and reasoning in language models through multiagent debate. In Forty-first International Conference on Machine Learning.

[29] Justin Chih-Yao Chen, Swarnadeep Saha, and Mohit Bansal. Reconcile: Round-table conference improves reasoning via consensus among diverse llms. arXiv preprint arXiv:2309.13007, 2023.

doi:10.6342/NTU202403801