ASSIGNMENT 2: PROMPT ENGINEERING & AGENT

Bohui Wu, 225010300
School of Science and Engineering (SSE)
The Chinese University of Hong Kong, Shenzhen
Shenzhen, China
225010300@link.cuhk.edu.cn

Prompt engineering is a relatively new discipline for developing and optimizing prompts to efficiently use language models (LMs) for a wide variety of applications and research topics. Prompt engineering skills help to better understand the capabilities and limitations of large language models (LLMs). Researchers use prompt engineering to improve the capacity of LLMs on a wide range of common and complex tasks such as question answering and arithmetic reasoning. Developers use prompt engineering to design robust and effective prompting techniques that interface with LLMs and other tools. Additionally, prompt engineering is essential for creating AI agents — autonomous systems that can perceive their environment, make decisions, and take actions to achieve specific goals. By carefully crafting prompts, engineers can guide these agents to perform complex, multi-step tasks, reason about their actions, and interact with both users and other digital tools in more sophisticated ways.

1 INTRODUCTION

Why is prompt engineering essential in the era of large language models? With the rapid advancement of large language models (LLMs) such as ChatGPT (OpenAI, 2025), DeepSeek (DeepSeek Platform, 2025), and Qwen, (Qwen Team, 2025) which have demonstrated remarkable capabilities in natural language understanding and reasoning tasks, the ability to interact with them effectively has become increasingly important. *Prompt engineering* — the process of designing and optimizing effective prompts — has emerged as a crucial discipline for unlocking the potential of LLMs across diverse applications, while enhancing their reliability, interpretability, and overall task performance. (Bhandari, 2023) By carefully crafting prompts, researchers and developers can gain deeper insights into model behavior, better control output quality, and achieve more consistent results across complex tasks.

What are the main prompting techniques, and how do they improve LLM reasoning and performance? A variety of prompting techniques have been developed to enhance the reasoning capabilities and robustness of LLMs. (Prompting Guide, 2025)For example, *zero-shot prompting* directly presents a question to the model without providing any examples, whereas *few-shot prompting* supplies a few demonstration instances to guide its responses. In contrast, *Chain-of-Thought (CoT) prompting* encourages the model to reason step by step before producing the final answer. Comparing these approaches helps reveal how LLMs respond to different types of contextual and instructional cues.

How do agent-based methods build upon prompt engineering to enable autonomous reasoning and tool use? In parallel, *agent-based methods* such as *Automatic Reasoning and Tool-use (ART)* extend traditional prompting by enabling multi-step reasoning, self-reflection, and simulated tool interactions. (Prompting Guide, 2025) These agents can autonomously determine when additional information is required, plan reasoning steps accordingly, and integrate external knowledge before generating their final responses. By combining reasoning and action, agent-based frameworks further expand the scope of prompt engineering, bridging the gap between static instruction following and dynamic problem solving.

In this report, I focus on **Task 1: LLMs as a Knowledgeable Doctor**, evaluating and comparing several prompting techniques (*Zero-shot, Role-based, Few-shotand and CoT*) and an agent-based method (*ART*). The comparison is conducted in terms of accuracy, completion time, reasoning quality, and ability to handle complex questions, providing insights into how prompt engineering and agent frameworks enhance LLM reasoning performance.

2 PROBLEM DEFINITION

Definition of the task *Task 1: LLMs as a Knowledgeable Doctor*, which aims to evaluate the reasoning and domain knowledge capabilities of LLMs in the context of the *pharmacist license examination*. The task requires the model to read multiple-choice questions covering topics such as pharmacology, pharmaceutics, and medical regulations, and to select the correct answer(s) based on professional knowledge and reasoning.

Input: A multiple-choice question $\mathbf{Q}$ consisting of a stem (the problem statement) and several candidate options $\mathbf{O} = \{O_1, O_2, ..., O_n\}$.

Output: The predicted correct option(s) $\hat{\mathbf{A}} \subseteq \mathbf{O}$, represented by their corresponding option letters (e.g., `A`, `BC`, or `ABD`).

Objective: The objective is to measure the ability of different prompting and agent-based techniques to generate accurate, consistent and explainable answers. The model response is considered correct if its predicted option set $\hat{\mathbf{A}}$ exactly matches the ground-truth answer $\mathbf{A}$ provided in the dataset.

Evaluation Criteria: Performance is primarily evaluated using **accuracy**:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of questions}}$$

Dataset: The dataset consists of 100 pharmacist exam questions sampled from the *National Pharmacist Qualification Examination (China)*, each containing the question stem, multiple options, and the ground-truth answer. Each record also includes an explanatory field (`analysis`) describing the rationale behind the correct answer, which is utilized in some experiments (e.g., for knowledge-base construction in the ART method). (see Appendix A.1).

3 PROMPTS AND THEIR DESIGN PHILOSOPHY

3.1 ZERO-SHOT AND ROLE-BASED PROMPTING

Zero-shot prompting is the simplest and most direct approach to interacting with LLMs. (Wei et al., 2021) It provides the model with only the question and options, without any examples or reasoning instructions. This design aims to evaluate the model's inherent knowledge and generalization ability learned during pretraining. In my experiment, the zero-shot setting served as a baseline.

Role-based prompting, as an extension of the *Zero-shot prompting*, introduces an explicit professional identity to the model, such as instructing it to act as a "licensed pharmacist" or "clinical expert." The purpose of this design is to align the model's responses with domain-specific knowledge and reasoning styles, thereby enhancing answer consistency and contextual relevance.

3.2 FEW-SHOT PROMPTING

Few-shot prompting provides the model with several example questionanswer pairs before presenting the target question. (Mann et al., 2020) By observing these demonstrations, the model can implicitly learn the desired answer format and reasoning pattern within the same prompt. This approach leverages in-context learning to guide the model's behavior without updating its parameters.

3.3 CHAIN-OF-THOUGHT PROMPTING

Chain-of-Thought prompting encourages large language models to generate intermediate reasoning steps before producing the final answer. (Wei et al., 2022) Instead of directly responding, the model is guided to "think aloud," breaking down complex problems into a sequence of logical inferences. This design philosophy aims to make the model's decision process more transparent and interpretable while improving its ability to handle multi-step reasoning tasks. By explicitly eliciting reasoning chains within the prompt, CoT transforms the model from a passive answer generator into an active problem solver capable of structured thought.

3.4 Agent-based Prompting and ART

Agent-based prompting represents a conceptual extension beyond traditional prompting, enabling large language models to act as autonomous decision-makers rather than passive responders. While conventional prompts rely on a single-turn instructionresponse pattern, agent-based methods introduce iterative reasoning, planning, and tool-use, allowing the model to dynamically determine what information is needed and how to obtain it. (Yao et al., 2022)

Within this paradigm, the *Automatic Reasoning and Tool-use (ART)* framework integrates two key capabilities: reasoning about the problem and invoking external knowledge sources when necessary. (Paranjape et al., 2023) This design philosophy transforms the model from a static language generator into an adaptive problem-solving agent capable of combining internal reasoning with external information retrieval, thereby enhancing interpretability, adaptability, and task performance.

3.5 Some example prompts

See Appendix A.2 for some example prompts.

4 Experiments

4.1 Experiment setting

Model configuration. All experiments were implemented in Python using the LangChain framework with the `DeepSeek` model as the underlying large language model. The model temperature was fixed at 0 to ensure deterministic outputs. Prompts were formatted dynamically to insert the question and options into each prompting template.

Prompting techniques. Four prompting techniques were compared: (1) **Zero-shot prompting**, the baseline without examples or reasoning instructions; (2) **Role-based prompting**, which adds a professional pharmacist persona to stabilize responses; (3) **Few-shot prompting**, which provides several example questionanswer pairs as demonstrations; and (4) **Chain-of-Thought prompting**, which explicitly guides the model to perform step-by-step reasoning. All prompting templates were implemented as Python string templates and executed in a unified evaluation pipeline.

Agent-based methods. Two agent-based approaches were implemented based on the *Automatic Reasoning and Tool-use (ART)* framework. The first, **Simplified ART**, simulates reasoning and tool-use behavior via structured prompts that include analysis, tool-decision, and reasoning steps. The second, **Enhanced ART with Two-Fold Cross Validation**, incorporates a real knowledge base built from the *analysis* fields of 50 questions in each fold and allows the agent to decide when to retrieve external information. This two-fold setup ensures fair evaluation and avoids data leakage.

Evaluation procedure. Each method was evaluated on the same 100 questions (50 per fold for ART). The evaluation pipeline automatically queried the model for each question and compared its extracted answer with the ground truth.

4.2 Quantitative Evaluations

Table 1 presents the quantitative performance of four prompting and one agent-based method on the pharmacist exam dataset. The results reveal clear differences in both accuracy and completion time across the evaluated approaches.

Among the prompting techniques, *Chain-of-Thought (CoT) prompting* achieved the highest accuracy among all prompt-based methods, demonstrating the effectiveness of explicit reasoning guidance in improving the model's logical consistency. By contrast, *Zero-shot prompting* served as a minimal baseline, measuring the model's inherent ability to recall domain knowledge without any contextual cues, achieving only 73% accuracy. *Role-based prompting*, which adds a professional pharmacist identity, provided a modest improvement to 81%, suggesting that persona grounding enhances factual stability but does not fully induce deeper reasoning. *Few-shot prompting*, which exposes the model to example demonstrations, achieved 79.4%, showing that in-context learning can improve performance but is still constrained by the diversity and representativeness of the exemplars.

Compared to these prompting techniques, the agent-based methods demonstrated stronger overall performance. The *Simplified ART* agent structures the prompt into distinct reasoning stages — *analysis*, *tool decision*, *reasoning*, and *answer* — thereby simulating the behavior of an autonomous agent. (see Appendix A.3.1) Although no external tools were actually invoked, the model was explicitly instructed to decide whether tool usage would be beneficial and to articulate that decision within its reasoning process. This simulated tool-use behavior, combined with structured reasoning guidance, improved the model's ability to plan and justify its answers, resulting in higher accuracy (87%) with moderate inference time (9.25 s per question).

Table 1: Comparison of different prompting and agent-based methods on pharmacist exam questions.

Method	**Avg. Time (s/question)**	**Accuracy (%)**
Zero-shot Prompting	1.38	73.0
Role-based Prompting	1.42	81.0
Few-shot Prompting	1.33	79.4
Chain-of-Thought Prompting	14.43	85.0
Agent-based: Simplified ART	9.25	87.0
Agent-based: ART (2-Fold Cross Validation)	9.50	90.0

The enhanced *ART (Two-Fold Cross Validation)* framework further advanced the agent's reasoning ability by incorporating actual tool-use and external knowledge retrieval. In this setup, the agent first constructs a domain-specific knowledge base from the *analysis* fields of 50 questions in each fold. During inference, the model autonomously decides whether to query this knowledge base by generating a Stage 1 "tool decision" prompt, retrieves relevant explanations through a simulated search function, and then integrates the returned information in Stage 2 reasoning before producing the final answer. (see Appendix A.3.2) This two-stage reasoning and retrieval process allows the agent to adaptively use external information rather than relying solely on internal recall. As shown in Table 2, the two folds achieved 94% and 86% accuracy, averaging 90%.

Table 2: Detailed results of Agent-based ART under two-fold cross validation.

Metric	**Fold 1**	**Fold 2**
Knowledge base size (keywords)	245	190
Accuracy (%)	94.0	86.0
Correct / Total	47 / 50	43 / 50
Avg. time (s/question)	9.56	9.44
Tool usage rate (%)	98.0	98.0
Tool helpful cases	18	16
Avg. reasoning length (chars)	250	245
Average accuracy (%)	**90.0**	
Total time (minutes)	15.8	

From an efficiency perspective, the *CoT* and *ART* approaches naturally required longer inference times due to multi-step reasoning and simulated tool-use. However, this trade-off proved beneficial, as the structured reasoning process substantially improved accuracy and interpretability.

Overall, the ART agent achieved the best balance between reasoning quality and efficiency, showing a 17% improvement over the zero-shot baseline while maintaining a reasonable average completion time of 9.5 seconds per question. These quantitative findings collectively demonstrate that while prompting techniques can elicit varying degrees of reasoning behavior, agent-based frameworks such as ART further enhance performance by enabling autonomous decision-making and adaptive knowledge retrieval.

4.3 Case Study

While quantitative evaluations provide an overall comparison of accuracy and efficiency across different prompting and agent-based methods, they do not fully reveal *how* these models reason and *why* certain methods succeed where others fail. To gain a deeper understanding of the behavioral differences between reasoning-based prompting and agent-based approaches, we conducted a detailed case study analysis.

This case study focuses on comparing the *Chain-of-Thought (CoT)* and *Agent-based ART* frameworks, as they both employ multi-step reasoning but differ fundamentally in structure. Specifically, CoT relies solely on internal step-by-step reasoning, whereas ART introduces an additional decision layer that allows the model to determine whether external knowledge retrieval is required and to incorporate the retrieved information into its reasoning process.

To conduct the analysis, we first identified questions where CoT and ART produced different outcomes — particularly those that CoT answered incorrectly but ART answered correctly. By examining the reasoning traces and outputs from both methods on these representative cases, we aim to uncover how structured reasoning and adaptive tool-use influence the model's problem-solving behavior in complex, knowledge-intensive scenarios.

Table 3: Detailed comparison between Chain-of-Thought (CoT) and ART (2-Fold) on 100 pharmacist exam questions.

Metric	**CoT**	**ART (2-Fold)**
Accuracy (%)	87.0	88.0
Correct / Total	87 / 100	88 / 100
Avg. time (s/question)	14.43	9.50
Total time (minutes)	24.0	15.8
Only CoT correct	4	–
Only ART correct	–	5
Both correct	83	
Both incorrect	8	
Net improvement (questions)	+1	
Improved question indices	[4, 20, 43, 70, 86]	
Degraded question indices	[11, 17, 25, 67]	

Comparative findings. Table 3 summarizes the per-question comparison between the *Chain-of-Thought (CoT)* and *Agent-based ART* methods. While both approaches demonstrated strong overall performance, the ART agent achieved a slightly higher accuracy (88%) compared to CoT (87%) and completed reasoning in less time on average (9.5 s vs. 14.4 s per question). Despite the seemingly small quantitative gap, a closer inspection reveals meaningful behavioral differences.

Specifically, ART corrected five questions that CoT failed on, while introducing four new errors, leading to a net improvement of one question. The two methods agreed on 83 correctly answered items and both failed on eight, indicating a large overlap in fundamental reasoning capability. However, the cases where ART outperformed CoT tended to involve complex or knowledge-dependent reasoning, such as identifying drug interactions, dosage constraints, or classification of controlled substances—tasks that require the integration of factual knowledge and contextual understanding.

These patterns suggest that ART's two-stage reasoning process, which includes an explicit tool-decision step and optional knowledge retrieval, helps the agent better identify when external information is needed and how to incorporate it effectively. In contrast, CoT's purely internal reasoning is often sufficient for simpler problems but may lead to incomplete or inconsistent conclusions on more specialized medical or regulatory topics. To illustrate these behavioral differences in depth, we next analyze a representative example where CoT produced an incorrect answer but ART succeeded in reasoning to the correct one.

Representative example: regulatory reasoning. To further understand the differences between reasoning styles, we select a representative question where the *Chain-of-Thought (CoT)* method produced an incorrect answer while the *Agent-based ART* framework succeeded. The chosen item concerns pharmaceutical retail regulations ("Which medicines can be displayed for self-selection in retail pharmacies?"), a type of problem that requires both pharmacological knowledge and an explicit understanding of legal constraints. Such questions provide a clear setting to observe how adaptive reasoning and tool-use contribute to improved decision-making.

In this case (see Appendix A.4), the *Chain-of-Thought (CoT)* model demonstrated a detailed and logically consistent reasoning process. It accurately recalled the distinction between prescription and non-prescription drugs and identified that only over-the-counter (OTC) products may be freely displayed. However, it failed to recognize that certain compound cough preparations—though commonly sold OTC—are still restricted by the *Drug Administration Law of the People's Republic of China*, leading to an overgeneralized answer (**BD**) instead of the correct choice (**B**).

The *Agent-based ART* method, by contrast, explicitly recognized the need for external regulatory knowledge during reasoning. At the tool-decision stage, the agent generated a query ("需要工具: 药品零售企业开架自选销售范围") and retrieved the relevant regulatory clause prohibiting the open display of narcotics, compound formulations containing controlled substances, and other special categories. Incorporating this retrieved information, the agent refined its reasoning and correctly selected only **B**.

This case exemplifies how ART's adaptive reasoning structure bridges the gap between factual recall and regulatory compliance. Whereas CoT's purely internal reasoning may overlook domain-specific constraints, ART's two-stage framework encourages deliberate information seeking and grounded decision-making. The result reinforces the quantitative finding that ART performs better on knowledge-intensive, regulation-dependent questions, supporting its advantage in structured, context-aware reasoning.

5 Conclusion

This study systematically compared multiple prompting strategies and an agent-based framework for pharmacist exam question answering. Experimental results demonstrated that while traditional prompting techniques such as Zero-shot, Role-based, Few-shot, and Chain-of-Thought prompting can elicit progressively stronger reasoning behaviors, their performance remains limited by the static nature of single-turn prompts.

In contrast, the ART framework extends prompt engineering into an adaptive reasoning paradigm, enabling explicit planning, self-evaluation, and simulated tool-use. Both the Simplified and Two-Fold versions of ART achieved higher accuracy and interpretability with moderate computational cost. The case analysis further revealed that ART excels in knowledge-intensive or regulation-dependent questions by actively identifying when and how to incorporate external information.

Overall, this work highlights the evolution from handcrafted prompts toward autonomous, agent-based reasoning systems—bridging static instruction following and dynamic, tool-augmented intelligence.

ACKNOWLEDGMENT

The complete implementation, experimental results, and related materials, such as some figures, are available at our GitHub repository: `https://github.com/wubh576/MDS-5110-NLP`.

This is the second assignment for MDS 5110. And my course instructor is Prof. Wang [1], and his lab homepage is here [2]. I would like to thank Prof. Wang and the Teaching Team for their guidance and support.

REFERENCES

Prabin Bhandari. A survey on prompting techniques in llms. *arXiv preprint arXiv:2312.03740*, 2023.

DeepSeek Platform. Deepseek api platform. `https://platform.deepseek.com/`, 2025. Accessed: 2025-10-29.

Ben Mann, Nick Ryder, Melanie Subbiah, J Kaplan, P Dhariwal, A Neelakantan, P Shyam, G Sastry, A Askell, S Agarwal, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 1(3):3, 2020.

OpenAI. Chatgpt platform. `https://chatgpt.com/`, 2025. Accessed: 2025-10-29.

Bhargavi Paranjape, Scott Lundberg, Sameer Singh, Hannaneh Hajishirzi, Luke Zettlemoyer, and Marco Tulio Ribeiro. Art: Automatic multi-step reasoning and tool-use for large language models. *arXiv preprint arXiv:2303.09014*, 2023.

Prompting Guide. Prompt engineering techniques guide. `https://www.promptingguide.ai/techniques`, 2025. Accessed: 2025-10-29.

Qwen Team. Qwen api platform. `https://qwen.ai/apiplatform`, 2025. Accessed: 2025-10-29.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

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.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*, 2022.

[1] `https://wabyking.github.io/old.html`
[2] `https://freedomintelligence.github.io/`

A APPENDIX

A.1 DATASET EXAMPLE

```
{
  "question": "27. 根据国家药品监督管理局，公安部，国家卫健康委员会的有关规定，服固体制剂每剂量单位含羟考酮碱不超过5毫克，且不含其他醉药品，精神药品或者药品类易制毒化学品的复制剂列（）。",
  "option": {
    "A": "含醉药品复制剂的管理",
    "B": "第类精神药品管理",
    "C": "第类精神药品管理",
    "D": "医疗毒性药品管理",
    "E": ""
  },
  "analysis": "服固体制剂每剂量单位含羟考酮碱不超过5毫克，且不含其他醉药品、精神药品或药品类易制毒化学品的复制剂列第类精神药品管理。",
  "answer": "B",
  "question_type": "最佳选择题",
  "source": "2021年执业药师职业资格考试《药事管理与法规》"
}
```

A.2 SOME EXAMPLE PROMPTS

A.2.1 ZERO-SHOT PROMPTING

```
prompt_zero = """请回答下列药师执照考试题:

{question}

{options}

只输出所有正确选项的字母，紧凑写在一行（如: A 或 BC 或 ABD）。不要输出任何多余文字。"""
```

A.2.2 ROLE-BASED PROMPTING

```
prompt_role = """你是一位拥有15年临床经验的资深执业药师，曾多次参与执业药师考试命题和阅卷工作，对药学知识和考试重点非常熟悉。

现在请运用你的专业知识回答以下考题:

题目: {question}

选项:
{options}

请直接输出正确答案（格式: A 或 BC 或 ABD）: """
```

A.2.3 FEW-SHOT PROMPTING

```
# 从前 3 题构造 few-shot 示例
few_shot_examples = ""
for i in range(3):
    q = data[i]['question']
    opts = '\n'.join([f"{k}: {v}" for k, v in data[i]['option'].items() if v])
    ans = data[i]['answer']
    few_shot_examples += f"【示例 {i+1}】\n题目: {q}\n{opts}\n正确答案: {ans}\n\n"
```

```
# 定义 few-shot prompt 模板
prompt_fewshot = f"""以下是几道药师执照考试的示例题及其正确答案。请学习示例的答题风格，并以相同格式回答最后一道题。

{few_shot_examples}

请模仿上述示例的答题方式，回答以下正式考试题。
只输出正确答案选项字母（如：A 或 BC 或 ABD），不要输出任何其他内容。

题目：{{question}}
选项：
{{options}}"""
```

A.2.4 CoT Prompting

```
prompt_cot = """你是资深药剂师，请分析以下考题：

题目：{question}

选项：
{options}

让我们一步一步地思考并分析每个选项。最后输出答案（格式：答案：A 或 答案：BC）"""
```

A.3 Algorithm

A.3.1 Simplified ART

Algorithm 1: Simplified Agent-based Reasoning (ART)

Input: Question q, Options $O = \{O_1, O_2, \ldots, O_n\}$, Model Chain M
Output: Predicted Answer $\hat{a}$

Step 1: Prompt Construction
Format the prompt as follows:
"你是智能药师 Agent，具备推理和工具调用能力。"
Include question q and options O;
Define available tools: `search_drug_database()`, `search_regulation()`, `search_clinical_guide()`;
Specify response format: （分析 → 工具决策 → 推理 → 答案）.

Step 2: Model Inference
Feed the constructed prompt to the model:
$r \leftarrow M$.`invoke(prompt)`

Step 3: Answer Extraction
Extract the predicted option letter from r using regex:
$\hat{a} \leftarrow$ `get_ans`(r)

Step 4: Evaluation Loop
for *each question* q_i *in dataset* D **do**
  Run Steps 13 to obtain $\hat{a}_i$;
  Compare with ground truth a_i;
  Compute accuracy: $\text{Acc} = \dfrac{\sum_i \mathbf{1}(\hat{a}_i = a_i)}{|D|}$;
return *Accuracy and detailed reasoning outputs.*

A.3.2 ENHANCED ART (TWO-FOLD CROSS VALIDATION)

Algorithm 2: Agent-based ART with 2-Fold Cross Validation

Input: Dataset D, Model Chain M
Output: Average accuracy and performance metrics

Step 1: Data Split
Divide D into two folds: (D_1, D_2), each with 50 questions.
Fold 1: D_1 as training, D_2 as testing;
Fold 2: D_2 as training, D_1 as testing.

Step 2: Knowledge Base Construction
For each fold, build a lightweight knowledge base KB from training analyses:
Extract key terms (drug / disease / keyword) $\rightarrow$ link to textual explanations.

Step 3: Two-Stage Agent Reasoning
For each test question q with options O:
(a) **Stage 1: Tool Decision** — decide whether external knowledge is needed;
(b) **Stage 2: Reasoning & Answering** — combine KB results (if any) and infer final answer $\hat{a}$.

Step 4: Evaluation per Fold
Compare predicted $\hat{a}$ with ground truth a; record accuracy, time, and tool usage.

Step 5: Cross-Fold Aggregation
Compute mean accuracy and average reasoning efficiency across folds:

$$\text{Accuracy}_{avg} = \frac{\text{Acc}_{fold1} + \text{Acc}_{fold2}}{2}$$

return *Average accuracy, tool usage rate, and reasoning length.*

A.4 CASE STUDY

Example Output

题目编号: 4
【题目】 19. 根据药品管理法律法规及相关件的规定，药品零售企业可以开架选销售的药品是（）。
【选项】 A: 复草 B: 复银花颗粒 C: 复地芬诺酯 D: 复枇杷喷托维林颗粒
【正确答案】 B
======================【CoT 方法】======================
预测答案: BD 判定: 错误
完整输出: 好的，我们先一步步分析。
—
1. 题目考点 题目问的是“药品零售企业可以开架自选销售的药品”，这涉及到药品分类管理规定。根据中国《药品经营质量管理规范》及相关规定： - **非处方药（OTC）** 可以开架自选销售，分为甲类OTC（须药师指导）和乙类OTC（安全性更高，可超市等销售）。 - **处方药** 必须凭处方销售，不得开架自选。 - 某些特殊药品（如含麻制剂、精神药品等）即使是非处方药，也可能有专门限制，不能开架。
—
2. 分析每个选项
A: 复方甘草片 - 含阿片粉，属于“含麻醉药品口服复方制剂”，必须凭处方销售，且不得开架自选，有严格登记要求。 - 不能开架自选。
B: 复方金银花颗粒 - 常见中成药，一般是 OTC（乙类非处方药），安全性高。 - 可以开架自选。
C: 复方地芬诺酯片 - 地芬诺酯属于麻醉药品（止泻药），被列为“必须凭处方销售的药品”，且是特殊管理药品，严禁开架销售。 - 不能开架自选。
D: 复方枇杷喷托维林颗粒 - 喷托维林是镇咳药，复方枇杷喷托维林颗粒属于 OTC（常见为甲类或乙类）。 - 查药品数据库：喷托维林单方是 OTC 乙类，复方制剂一般也是 OTC，但需确认是否在禁止开架的“含特殊药品复方制剂”名单中。 - 含喷

托维林的复方制剂，不属于“含麻制剂”或“含精神药品复方制剂”限制开架的名单（限制开架的主要是含麻黄碱类、可待因、阿片、地芬诺酯等）。 - 因此，此药通常按OTC 管理，可以开架销售。 - 可以开架自选。
—
3. 结论 可以开架自选的是 **B 和 D**。
—
答案：BD
======================【ART 方法】======================
预测答案: B 判定: 正确
工具决策: 需要工具：药品零售企业开架自选销售范围
工具返回: 【工具返回】 药品零售企业不得经营的药品：醉药品、放射性药品、第类精神药品、终妊娠药品（包括含有“司酮”成分的所有药品制剂）、蛋同化制剂、肽类激素（胰岛素除外）、药品类易制毒化学品、疫苗，以及我国法律法规规定的其他禁零售的药品。药品零售企业也不得经营中药配颗粒、医疗机构制剂。
完整输出: 根据《药品管理法》及相关规定，药品零售企业不得开架自选销售非处方药以外的药品，特别是含特殊药品复方制剂等有严格管理要求的品种。
- **A: 复方甘草片** —— 含阿片成分，属于必须凭处方销售、严格管理的药品，不得开架自选。 - **B: 复方金银花颗粒** —— 属于中成药，若为OTC（非处方药），一般可开架销售。 - **C: 复方地芬诺酯片** —— 属于必须凭处方销售的药品，且易滥用，不得开架。 - **D: 复方枇杷喷托维林颗粒** —— 含喷托维林（镇咳药），属于处方药，不得开架。
因此，可以开架自选销售的应为 **B: 复方金银花颗粒**（假设其为非处方药）。
答案：B