

MMA: Cross-Domain Knowledge Integration via Mixture of Multi-Domain Agents

Kehang Jia, Juntao Li^{*}, Xiaobo Liang, Yisheng Xiao, Yixuan Yang, Min Zhang

Soochow University, China

{khjia,ysxiao,xyyangy}@stu.suda.edu.cn

{ljt,xbliang,minzhang}@suda.edu.cn

Abstract

Rather than merely to retain previously acquired generalization, achieving synergistic improvements between generalization and domain specialization in foundation models remains a significant challenge in both pre-training and post-training. As an alternative, we propose a test-time cross-domain knowledge integration method, Mixture of Multi-domain Agents (MMA), which dynamically combines the outputs of general-purpose and domain-specific models to enhance their performance on complex, domain-specific tasks. MMA formulates the integration process as a search problem, using Monte Carlo Tree Search (MCTS) to find the path that optimally harmonizes the respective strengths of different models in generalization and domain-specific knowledge. In addition, We design specific action spaces to control the knowledge integration between multiple models, and cross-inspection reward is introduced to fairly score strategies in different domains. Experiments in diverse domains show that MMA can effectively combine the strengths of different models to enhance their performance. For instance, in legal tests, the average performance of all tasks increased from 42.57% to 53.68%. In financial tests, it improved from 56.01% to 62.68%¹.

1 Introduction

Large Language Models (LLMs) demonstrate impressive capabilities in handling complex tasks that require both specific knowledge and sophisticated reasoning. However, these models tend to entangle two inherently distinct actions: *memory recall* and *reasoning* (Jin et al., 2024). Such entanglement can lead to the issue such as catastrophic forgetting (Kaushtik et al., 2021) during continual pre-training or domain-specific fine-tuning. As illustrated in Figure 1, we observe a clear perfor-

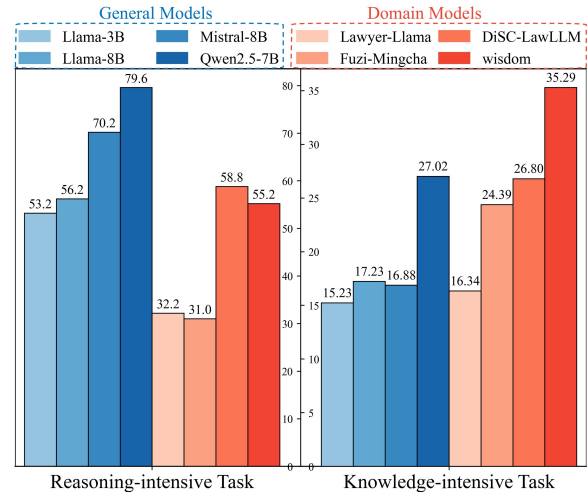


Figure 1: Compared to general models, legal domain-specific models exhibit stronger performance on legal knowledge-intensive tasks than on legal reasoning-intensive tasks.

mance divergence across task types: on reasoning-intensive tasks, the general model Qwen2.5-7B outperforms the domain-specific model wisdom by 24.4 points, whereas on knowledge-intensive tasks, wisdom exceeds Qwen2.5-7B by 8.27 points. This phenomenon reveals a strong complementarity between general and domain-specific models, motivating the development of dynamic integration frameworks that unify the strengths of both reasoning and memory recall.

To address the aforementioned challenges, we explore test-time cross-domain integration strategies, which offer two primary advantages: (1) low dependence on high-quality domain-specific data; (2) high computational efficiency. Unlike traditional training approaches that require carefully balance datasets to harmonize general reasoning capabilities with domain-specific knowledge, test-time integration enables model adaptability even when domain data is scarce or imbalanced. Moreover, test-time methods aim to dynamically leverage the complementary strengths of general and

^{*} Juntao Li is the Corresponding Author.

¹Code is available at <https://github.com/khjia810/MMA>

domain-specific models, thereby mitigating the gap between knowledge utilization and generalization and enhancing the effectiveness of knowledge integration without introducing the additional costs.

In this paper, we propose Mixture Multi-Domain Agents (MMA), which decomposes problem solving into a process of collaboration–verification among multiple models for cross-domain knowledge integration. Specifically, unlike previous Monte Carlo Tree Search (MCTS) research(Qi et al., 2024; Hao et al., 2023; Browne et al., 2012) that mainly focuses on the problem formulation level, MMA designs an MCTS action-space strategy for model interaction and knowledge integration. This action space defines operations such as model invocation, knowledge extraction, inspection and review, enabling the dynamic exploration of diverse integration strategies. Furthermore, to ensure method robustness in cross-domain knowledge integration scenarios, we cannot rely on any single model to serve as a verification model across all domains; for example, even broadly capable models like GPT-4o cannot cover every specialized field. MMA addresses this challenge through cross-inspection reward. Specifically, for nodes that require evaluation, MMA uses models that were not involved in the final answer generation, along with part of that node’s reasoning trajectory for subsequent answer generation. The consistency of the two answers for the node is then used to provide unsupervised feedback.

Extensive experiments using general models in conjunction with different domain-specific models separately on different domain reasoning tasks have demonstrated the effectiveness of MMA. The results show that cross-domain knowledge integration can effectively combine the strengths of multiple models at different dimensions, achieving a synergistic optimization of specialized knowledge and generalization. In legal tasks, our method improves the performance of the legal domain model from 42.57% to 53.68%. In finance, it improves the performance of the financial domain model from 56.01% to 62.68%.

In summary, our contributions are as follows:

- We introduce a test-time cross-domain knowledge integration paradigm, which can harmonize domain knowledge and general reasoning capability with low computational cost.
- We reformulate cross-domain reasoning as a search problem, and design a novel action

space for MCTS to explore diverse integration strategies across a mixture of models.

- To enhance the effectiveness of the integration process, we propose a cross-inspection reward mechanism that guides the MCTS search towards more reliable reasoning trajectories.

2 Related Work

Monte Carlo Tree Search. Starting from the initial root node, the search tree is gradually optimized by iteratively executing a four-stage process of "selection-expansion-simulation-backpropagation". Each completion of the above four-stage process is considered as one rollout. Hao et al. (2023) combined MCTS with LLMs by constructing world models and reward functions, thereby enhancing the planning capabilities. In addition, some studies(Vodopivec et al., 2017; Wu et al., 2024) have enhanced the performance of models by combining MCTS and reinforcement learning methods. In addition to the direct application of reinforcement learning, some studies(Lin et al., 2025; Hu et al., 2025) have also designed new action spaces to utilize the MCTS algorithm. For example, Qi et al. (2024) proposed the rStar algorithm, which enhances MCTS by decomposing problems into subproblems and other human-like reasoning actions, significantly improving the model’s reasoning capabilities. ToA(Ye et al., 2024) further combines MCTS with multi-agent, achieving model coordination through a multi-step decision-making process. However, most existing MCTS approaches cannot support multi-agent collaboration, and although ToA realizes multi-agent collaboration, its research goal is data synthesis, and its settings are impractical for capability enhancement. In contrast, our proposed method effectively realizes integration among multiple agents, and enhances their performance on complex, domain-specific tasks.

Multi-Agent Collaboration. Unlike deepening the capabilities of a single model to enable it to plan and reason in a human-like manner(Hao et al., 2023; Zhou et al., 2022; Khot et al., 2020), multi-agent(Tran et al., 2025) focuses on the study of interactions between agents as a way of combining capabilities between agents. Some studies(He et al., 2024; Rasal, 2024; Chen et al., 2023) assign different roles to LLMs, allowing them to interact and reason about the problem through collaboration

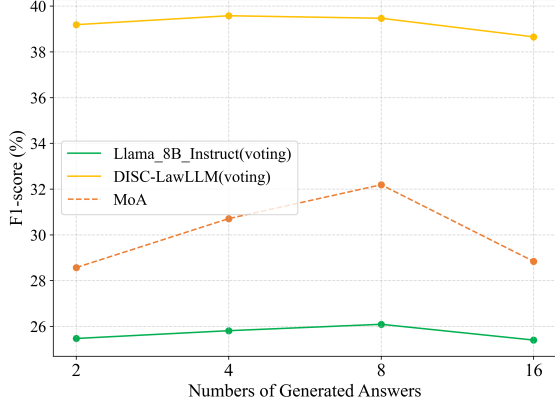


Figure 2: Performance comparison of the MoA method and its baseline models with different number of generated answers on the Legal Judgment Prediction (LJP) task of LawBench.

or debate. Multi-agent reinforcement(Sun et al., 2024; Wei et al., 2025) learning enables collaboration by allowing multiple agents to interact within a shared environment based on reward signals. Recently, mixture-of-agents(Wang et al., 2024) combined the outputs of multiple LLMs to obtain a higher-quality final result. Later, Li et al. (2025) proposed Self-MoA, based on whether mixing different LLMs is truly beneficial. This method uses only the best-performing LLM to generate multiple outputs through multiple samplings, and then aggregates these outputs into a high-quality final result using an aggregator. Although these methods improve model performance through multi-agents, they have only explored their effectiveness for signal-domain tasks. In contrast, MMA effectively achieves cross-domain knowledge integration, realizing a synergistic optimization of specialized knowledge and generalization, rather than merely facilitating collaboration among models.

3 Preliminary

As a representative approach to knowledge integration, Mixture of Agents (MoA) aggregates the outputs of multiple models with the assumption that referencing the outputs of multiple models can generate higher quality responses. However, MoA may neglects for the heterogeneity in domain expertise across different agents, which may result in knowledge conflicts and suboptimal performance. As shown in Figure 2, we make the following key observations: (1) MoA outperforms the general model LLaMA3-8B-Instruct but underperforms the domain-specific model DISC-LawLLM,

suggesting that traditional ensemble-based integration methods are not well-suited for cross-domain scenarios. (2) Even when incorporating simple knowledge alignment mechanisms, MoA still exhibits a significant performance gap compared to DISC-LawLLM, indicating its limited ability to resolve domain-specific knowledge conflicts. *These findings highlight the limitations of existing methods and motivate the development of more adaptive agent collaboration mechanisms for cross-domain knowledge integration.*

4 Mixture of Multi-Domain Agents

Due to the confusion caused by directly mixing multi-model answers, we attempted to explore multiple integration strategies to avoid introducing erroneous knowledge. we enable models to interact and verify with each other, i.e., models can generate answers using their own knowledge or knowledge provided by other models, and then modify the provided knowledge as well as reward scoring of responses. As shown in Figure 3, we establish a framework that enables dynamic exploration of integration strategies.

4.1 Dynamic Integration

Based on the characteristics of the MCTS method, we can sample multiple reasoning paths to obtain answers. In contrast to the previous mathematical and reasoning directions for problem formulation, delineation, or enhanced reasoning capability, we design action spaces for enabling model interaction and knowledge integration. Through our designed action space, models can generate answers by leveraging either their own knowledge or the domain knowledge provided by the domain-specific model. After the domain-specific model provides specialized knowledge, the knowledge is continuously refined based on the answers generated by the general model using this knowledge. We also introduce an element extraction step to render the question clearer. The action space we designed is described as follows:

A1: Generate Answer with General Model.

This action is one of the core aspects of cross-domain knowledge integration. Based on the information contained in the current node (i.e., the question and the injected domain knowledge), the general model is invoked to generate answers. By leveraging the general model’s strong generalization and the specialized knowledge provided

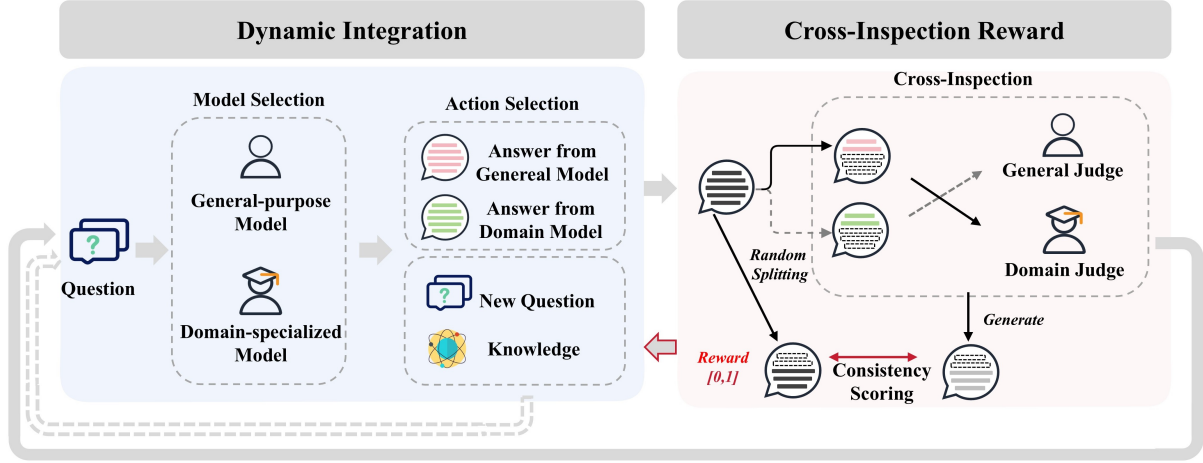


Figure 3: The multiple rounds iterative process of MMA comprises two components: (i) Multiple strategies are realized through selection and interaction between different models. (ii) Cross-inspection reward, in which a verification model computes a consistency reward for the selected path.

at this node, this action integrates the strengths of both models in terms of domain knowledge and generalization.

A2: Generate Answer with Domain-specific Model. Directly invoking the domain-specific model to generate answers can fully utilize the expertise accumulated in the domain-specific model. This action prevents the general model from failing to leverage the domain-specific model’s knowledge effectively, leading a bias where the general model has been guided by its own internal knowledge when answering using the knowledge of the domain-specific model.

A3: Generate and Refine Knowledge. Based on observations from the Self-MoA experiments, we found that this method yields only marginal improvements over the best baseline model.

Previous mixture-of-agents approaches re-input the answers produced by LLMs into other LLM, thus achieving knowledge integration. However, we suggest that the answers generated by LLMs are insufficient to provide the complete knowledge to solve the problem and prove the conjecture correct in an ablation study.

To address this, we employ the domain-specific model to conduct in-depth analysis of problems and generate relevant knowledge that can be used to solve them. At the same time, this knowledge is constantly reviewed and refined, thus continuously improving and enriching the provided domain knowledge.

Actions

- A1: Generate Answer with General Model
- A2: Generate Answer with Domain-specific Model
- A3: Generating and Refine Knowledge
- A4: Extract the Elements

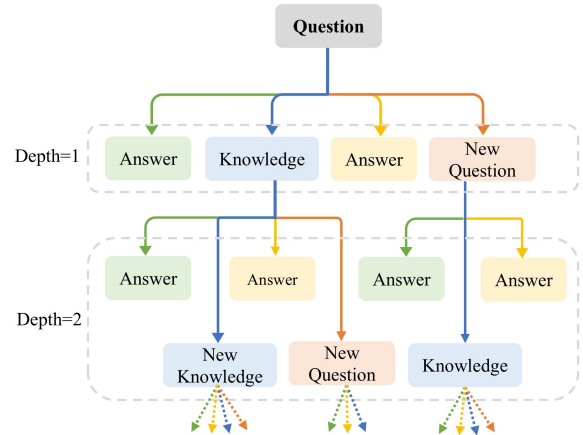


Figure 4: An example showing the extension of MCTS. Given a problem, MCTS explores the state space through various actions to construct the search tree and converges toward the optimal integration strategy.

A4: Extract the Elements. We retain part of the formulation of the question actions, that is, element extraction. This action rephrases the original question and subdivides the conditions, making the overall problem structure clearer.

The above four actions constitute an action space $\{A_1, A_2, A_3, A_4\}$ that can be used for model interaction and knowledge integration. For different node types, there exist action restrictions. These include (i) the child nodes generated by the A_1 and A_2 actions cannot perform any other actions. (ii) Once action A_4 exists in the path from a node to the root node, it cannot perform A_4 again. To visually show the search-tree construction process, we

present the extension process in Figure 4, the root node represents the problem to be solved, and each child node is generated by expanding the different actions represented by each edge in the graph.

4.2 Cross-Inspection Reward

Design of the Reward Function. The key components in the construction of the MCTS include the action space and the reward function. The reward function evaluates the value of each node, providing guidance for the construction of the tree.

We based on the idea that it can be more convincing that different models got the same answer under the same inference path (Xu et al., 2023; Cobbe et al., 2021), we design a simple but effective reward function, described as follows:

In each rollout, once the simulation process is completed, a path is formed from the root node n_1 to the leaf node n_d , denoted as $t = n_1 \oplus n_2 \oplus \dots \oplus n_d$. The reward r for path is calculated based on the consistency of the answers at the leaf node.

Specifically, LLM M uses the context c of the leaf node n_d to generate the final answer.

$$p(a|c) = \prod_{i=1}^{|a|} M(a_i|c, a_1, \dots, a_{i-1}), \quad (1)$$

where a denotes the answer generated based on the context c , a_i denotes the i -th token in the answer, and $|a|$ is the length of the answer.

Next, the verification model $M_{\text{verification}}$ uses the context c and the first k tokens of the final answer, $a_{1:k}$, to generate the remaining answer.

$$p(\hat{a}|c) = \prod_{i=k}^{|a|-k} M_{\text{verification}}(a_i|c, a_1, \dots, a_{i-1}). \quad (2)$$

Finally, the consistency of $\hat{a}$ and $a_{1:k}$ is computed as the reward r . This reward is then updated to each node along the path, as shown below.

$$Q(n_i) \leftarrow Q(n_i) + r, \forall i \in \{1, 2, \dots, d\}, \quad (3)$$

where $Q(n_i)$ denotes the cumulative reward value of node n_i . Initially, for all unexplored nodes, we set $Q(n_i) = 0$ to facilitate the random expansion of the search tree. $\forall i \in \{1, 2, \dots, d\}$ indicates updating the cumulative reward value of all nodes on the path, not just the leaf node. This approach allows a better evaluation of intermediate nodes and converges toward the optimal integration strategy.

Selection of Verification Model. After identifying the reward function, we still face a critical challenge: selecting the appropriate model to serve as the verification model. To ensure robustness across different domains, we cannot find any single general model whose knowledge can cover all domains as our verification model. Otherwise, it will introduce knowledge bias and lead to unfair reward distribution. To address this problem, we design cross-inspection reward. For the selection of the verification model $M_{\text{verification}}$, we use a non- M model as $M_{\text{verification}}$. i.e., we use another LLM in the MMA for validation to avoid self-validation. The effectiveness of this method is demonstrated in an ablation study.

Final Answer Selection. After completing the construction of the MCTS, a path must be selected as the final solution path to obtain the final answer. The path selection process starts from the root node and proceeds recursively as follows:

$$S(n) = \begin{cases} n, & \text{if } n \text{ is } n_{\text{leaf}} \\ S\left(\arg \max_{n' \in \text{children}(n)} Q(n')\right), & \text{otherwise} \end{cases} \quad (4)$$

Starting from the root node, if the current node n is a leaf node, it is directly returned as the endpoint of the path. Otherwise, among all child nodes of n , denoted as $\text{children}(n)$, the one with the highest reward value Q is selected. The same selection process is then recursively applied from that node until a leaf node is reached.

5 Experiments

5.1 Setup

Models and Datasets. We tested the performance of MMA in different domains. The models used include the general model Llama-3.1-8B-Instruct (Meta, 2024), the legal domain model DISC-LawLLM (Yue et al., 2023) and the financial domain model XuanYuan-13B-Chat (Zhang and Yang, 2023). We tested 6 different Q&A datasets from 3 different domains. The legal domain includes Legal Judgment Prediction (LJP) (Fei et al., 2023), Extraction of Criminal Case Amount (ECA) (Fei et al., 2023), and Legal Question Answering (LQA) (Dai et al., 2023). The financial domain includes Financial Exam Questions (FEQ) (Xie et al., 2024) and Financial Knowledge

Test (FKT)(Lei et al., 2023). Additionally, we randomly selected 800 samples from AGIEVAL (General)(Zhong et al., 2023) as a dataset to evaluate the general capabilities.

Implementation Details. In the baseline comparison experiments, the method was set up as follows: We set the depth d of the Monte Carlo Tree Search to 5, meaning that action A_2 can produce a maximum of 5 nodes in terms of depth, while the default node count for the other actions is 1. And We set the number of rollouts (i.e., the number of iterations) for the MCTS to 15. For each node that generates an answer, 5 candidate answers are produced, and the answer with the majority vote is chosen as the node’s answer.

While calculating the reward, for a given answer process, we partition the process into 20% to 60% segments. The first segment of the process is used as input to the verification model, and the verification model then completes the remaining part to calculate the reward.

Baselines. We compared our method with four baselines:(i) Generating answers using a single-round chain-of-thought (CoT)(Wei et al., 2022) with a few examples.(ii) Sampling 16 and 32 candidate answers based on the CoT with examples, then selecting the final answer by majority vote.(iii) MoA(Wang et al., 2024) approach, where preliminary answers from all models are generated and aggregated using Aggregators.(iv) Self-MoA(Li et al., 2025), in which only the best-performing model generates preliminary answers that are then aggregated using Aggregators.

5.2 Main Results

We evaluated the domains covered by the models in MMA. Table 1 compares our method with state-of-the-art baselines of mixture-of-agents on different methods and datasets.

We emphasize three key observations:(1) Consistent with our description in the Preliminary, the MoA method suffers from preliminary answers produced by different models that confuse the aggregator, resulting in significant performance degradation. In legal domain tests, on average, performance improved by 1.92% compared to the legal domain model’s Few-Shot CoT, but decreased by 3.06% compared to multi-sampling (FS-CoT maj@32). In financial domain tests, performance improved by 0.37% over the financial domain model’s Few-Shot CoT, while decreasing by 2.68%

compared to FS-CoT maj@32. Furthermore, Self-MoA fails to fully combine the capabilities of both models, the method exhibits similar performance to the performance of the results generated by FS-CoT maj@32 of the domain-specific model, with a decrease of 0.11% in the legal domain test and an improvement of 0.46% in the financial domain test. (2) MMA improves performance on the domains covered by the models used compared to the best baseline models. When constructing the MMA using the general model Llama-8B-Instruct and the legal domain model DISC-LawLLM, it improved the average performance of the legal domain model on all tasks from 42.57% to 53.68%. Likewise, when general model is paired with the financial domain model XuanYuan-13B chat, it increases the average performance of the financial domain model from 56.01% to 62.68%. (3) MMA’s improvement on complex, domain-specific tasks is greater than its improvement on general tasks. For example, in the general domain, it only improves performance from 39.73% to 43.44% in legal (or 44.37% in financial), while in the legal ECA task, it boosts performance from 58.80% to 70.60%. These demonstrate that multi-domain knowledge integration can significantly enhance model capabilities on specific complex tasks.

5.3 Ablation Study

Effectiveness of the Action Space. As shown in Table 2, we evaluated the effectiveness of each action in our action space. First, we modified action A3 so that it produces answers instead of generating knowledge. Second, we tested the performance under missing A4 compared to the full method.

Effectiveness of the Reward Mechanism. The reward r for candidate answer is computed as

$$r = n_{\text{conf}} \times \text{consistency}(\hat{a}, a_{k+1:|a|}), \quad (5)$$

where n_{conf} denotes the confidence score when generating the candidate answer at node, i.e., the proportion of times that answer appears. $\text{consistency}(\hat{a}, a_{k+1:|a|})$ measures the semantic consistency of the remaining answer $\hat{a}$, which is generated based on the $a_{1:k}$, and the original answer segment $a_{k+1:|a|}$. As shown in Figure 5, we sampled 500 r on the ECA and General tasks to observe the performance of the reward mechanism on the domain-specific and general domains. The verification model can give higher values to positive

Method		Legal				
Model	Config	General	LJP	LQA	ECA	Avg.
Llama-8B-Instruct	maj@1	39.73	16.40	29.88	56.20	34.80
	maj@16	40.66	25.40	35.19	58.40	39.91
	maj@32	41.32	26.67	34.91	59.00	39.63
DISC-LawLLM	maj@1	27.68	49.29	34.52	58.80	42.57
	maj@16	32.98	53.66	36.98	59.80	45.86
	maj@32	36.02	55.59	39.18	59.40	47.55
MoA	sample@32	40.26	43.84	37.66	56.20	44.49
Self-MoA	sample@32	34.30	55.41	38.66	61.40	47.44
MMA		43.44	58.68	42.01	70.60	53.68

Method		Financial			
Model	Config	General	FEQ	FKT	Avg.
Llama-8B-Instruct	maj@1	39.73	46.85	67.33	51.30
	maj@16	40.66	44.59	69.33	51.53
	maj@32	41.32	45.94	69.33	52.20
XuanYuan-13B-chat	maj@1	35.36	52.00	80.67	56.01
	maj@16	38.01	54.05	82.00	58.02
	maj@32	39.09	55.41	82.67	59.06
MoA	sample@32	41.59	49.55	78.00	56.38
Self-MoA	sample@32	40.13	55.86	82.67	59.55
MMA		44.37	59.01	84.67	62.68

Table 1: We mainly test in legal and finance, and report performance on the general test (General). All tests use a two-example few-shot prompt. Here, maj@n denotes the majority vote among n sampled.

Method	ECA	General
MMA	70.60	43.44
w/o A1	68.88	38.41
w/o A2	63.40	40.39
w/ A3 (answer generation)	65.60	42.38
w/o A4	68.60	43.03

Table 2: Ablation study on the effectiveness of the action space for implementing the mixture-of-agents approach. We evaluated this both on domain-specific (ECA) and general tasks (General).

cases (where the node is consistent with the true answer), i.e., the distribution is skewed towards "+1". At the same time, it can give lower values to negative examples (the node is not consistent with the true answer), i.e., the distribution is skewed towards "-0". This demonstrates that the reward mechanism is effective and provides proper guidance for the MCTS.

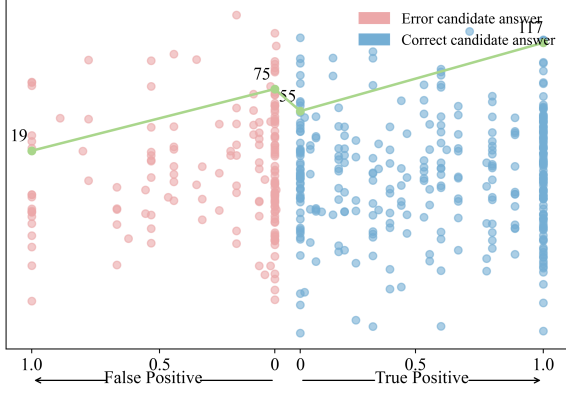
We also evaluated using the general model from MMA as our verification model. As shown in Table 3, employing the general model as a verification model introduces bias, and the lower the model’s capability, the greater the bias. Therefore, for the

Verification Model	Llama-3B & LawLLM	Qwen2.5-7B & LawLLM
General model	52.54	59.19
Cross model	57.35	61.85

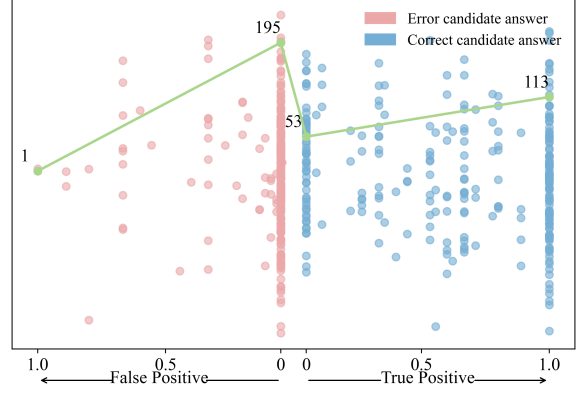
Table 3: We applied MMA to the LJP task using Llama-3B-Instruct or Qwen2.5-7B-Instruct in conjunction with LawLLM. We then evaluated performance under two verification mechanisms: our proposed cross-verification method and, alternatively, using the general model from MMA as the verification model.

robustness of the method, it is not possible to identify a model that is proficient across all domains to serve as our verification model.

Performance under Different Rollouts. Since MMA is based on MCTS for search expansion, completing one full four-stage process constitutes a rollout. Thus, the number of rollouts implies the extent of exploration in the search tree. In Table 4, we observe that the improvement in MMA is high when the number of rounds increases from 5 to 10, indicating that it is necessary to fully explore the state space.

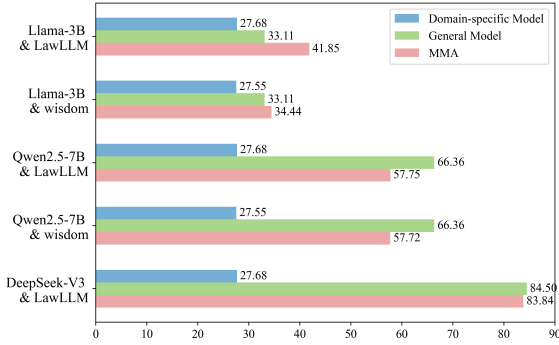


(a) Distribution of rewards on general test

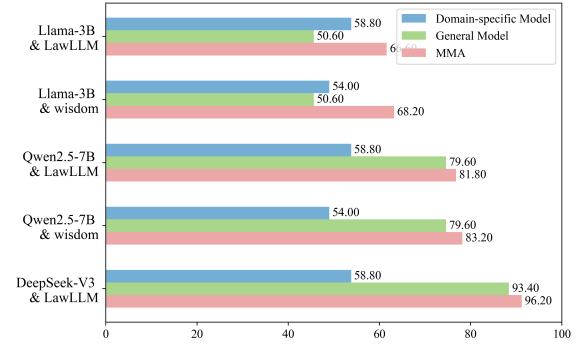


(b) Distribution of rewards on domain-specific test

Figure 5: We sampled 500 reward calculations from nodes in both the domain-specific (b) and general (a) tests. The correctness of candidate answer is determined by comparing it against the ground-truth answer. A higher absolute value of a reward indicates higher confidence in that answer.



(a) Different models' performances on general task



(b) Different models' performances on domain-specific task

Figure 6: We paired general models (Llama-3B-Instruct, Qwen-2.5-7B-Instruct and DeepSeek-V3) with domain-specific models (LawLLM and wisdomInterrogatory), and evaluated each combination on the general task (General) as well as the domain-specific task (ECA).

Rollouts	General	ECA
5	41.72	68.80
10	43.27	70.80
15	43.44	70.60
20	43.20	70.60

Table 4: We perform different numbers of rollouts on domain-specific (ECA) and general (General) tasks. In all other MMA experiments, the number of rollouts is set to 15.

Different combinations of models We further investigated how different combinations of general and domain-specific models affect our method. As shown in Figure 6, we found that even if general model exhibits poor performance on complex, domain-specific tasks, integrating its generalization with specific knowledge enables to achieve strong results. Conversely, if the general model

already shown a strong performance on complex, domain-specific tasks, our approach still provides additional performance gains by further leveraging the strengths of each model. We observed a drop in MMA’s performance on the general test when we used a strong general model, and we attribute this to the fact that the domain model gives useless knowledge leading to the harmonization of generalization and domain-specific knowledge, resulting in a drop in performance on general test.

6 Conclusion

In this work, we propose MMA, a method that achieve cross-domain knowledge integration via mixture of multi-domain agents. By dynamically exploring integration strategies, we find optimal harmonizes the respective strengths of different models in generalization and domain-specific knowledge. Experiments across multiple domains

and tests demonstrate the effectiveness of the method. Extensive ablation studies and analyses further validate the efficacy of the components within the approach.

7 Limitations

Our method requires multiple rounds of iteration, which introduces inference delays and makes it difficult to satisfy application scenarios with a high demand for real-time performance. Moreover, our approach depends on a domain-specific model to provide domain knowledge; if the target domain does not have a sufficiently fine-tuned model or the knowledge coverage of the existing model is insufficient, our approach may not achieve the desired performance.

8 Acknowledge

We want to thank all the anonymous reviewers for their valuable comments. This work was supported by the National Science Foundation of China (NSFC No. 62206194), the Natural Science Foundation of Jiangsu Province, China (Grant No. BK20220488), and the Young Elite Scientists Sponsorship Program by CAST (2023QNRC001). We also acknowledge MetaStone Tech. Co. for providing us with the software, optimisation on high performance computing and computational resources required by this work.

References

- Cameron B Browne, Edward Powley, Daniel Whitehouse, Simon M Lucas, Peter I Cowling, Philipp Bohnlshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. 2012. A survey of monte carlo tree search methods. *IEEE Transactions on Computational Intelligence and AI in games*, 4(1):1–43.
- Weize Chen, Yusheng Su, Jingwei Zuo, Cheng Yang, Chenfei Yuan, Chen Qian, Chi-Min Chan, Yujia Qin, Yaxi Lu, Ruobing Xie, and 1 others. 2023. Agentverse: Facilitating multi-agent collaboration and exploring emergent behaviors in agents. *arXiv preprint arXiv:2308.10848*, 2(4):6.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, and 1 others. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.
- Yongfu Dai, Duanyu Feng, Jimin Huang, Haochen Jia, Qianqian Xie, Yifang Zhang, Weiguang Han, Wei Tian, and Hao Wang. 2023. Laiw: a chinese legal large language models benchmark. *arXiv preprint arXiv:2310.05620*.
- Zhiwei Fei, Xiaoyu Shen, Dawei Zhu, Fengzhe Zhou, Zhuo Han, Songyang Zhang, Kai Chen, Zongwen Shen, and Jidong Ge. 2023. Lawbench: Benchmarking legal knowledge of large language models. *arXiv preprint arXiv:2309.16289*.
- Shibo Hao, Yi Gu, Haodi Ma, Joshua Jiahua Hong, Zhen Wang, Daisy Zhe Wang, and Zhiting Hu. 2023. Reasoning with language model is planning with world model. *arXiv preprint arXiv:2305.14992*.
- Chengbo He, Bochao Zou, Xin Li, Jiansheng Chen, Junliang Xing, and Huimin Ma. 2024. Enhancing llm reasoning with multi-path collaborative reactive and reflection agents. *arXiv preprint arXiv:2501.00430*.
- Yunhai Hu, Yilun Zhao, Chen Zhao, and Arman Cohan. 2025. Mcts-rag: Enhancing retrieval-augmented generation with monte carlo tree search. *arXiv preprint arXiv:2503.20757*.
- Mingyu Jin, Weidi Luo, Sitao Cheng, Xinyi Wang, Wenyue Hua, Ruixiang Tang, William Yang Wang, and Yongfeng Zhang. 2024. Disentangling memory and reasoning ability in large language models. *arXiv preprint arXiv:2411.13504*.
- Prakhar Kaushik, Alex Gain, Adam Kortylewski, and Alan Yuille. 2021. Understanding catastrophic forgetting and remembering in continual learning with optimal relevance mapping. *arXiv preprint arXiv:2102.11343*.
- Tushar Khot, Daniel Khashabi, Kyle Richardson, Peter Clark, and Ashish Sabharwal. 2020. Text modular networks: Learning to decompose tasks in the language of existing models. *arXiv preprint arXiv:2009.00751*.
- Yang Lei, Jiangtong Li, Dawei Cheng, Zhijun Ding, and Changjun Jiang. 2023. Cfbenchmark: Chinese financial assistant benchmark for large language model. *arXiv preprint arXiv:2311.05812*.
- Wenzhe Li, Yong Lin, Mengzhou Xia, and Chi Jin. 2025. Rethinking mixture-of-agents: Is mixing different large language models beneficial? *arXiv preprint arXiv:2502.00674*.
- Qingwen Lin, Boyan Xu, Zijian Li, Zhifeng Hao, Keli Zhang, and Ruichu Cai. 2025. Leveraging constrained monte carlo tree search to generate reliable long chain-of-thought for mathematical reasoning. *arXiv preprint arXiv:2502.11169*.
- AI Meta. 2024. Introducing meta llama 3: The most capable openly available llm to date, 2024. URL <https://ai.meta.com/blog/meta-llama-3>, 26.
- Zhenting Qi, Mingyuan Ma, Jiahang Xu, Li Lyna Zhang, Fan Yang, and Mao Yang. 2024. Mutual reasoning makes smaller llms stronger problem-solvers. *arXiv preprint arXiv:2408.06195*.

- Sumedh Rasal. 2024. Llm harmony: Multi-agent communication for problem solving. *arXiv preprint arXiv:2401.01312*.
- Chuanneng Sun, Songjun Huang, and Dario Pompili. 2024. Llm-based multi-agent reinforcement learning: Current and future directions. *arXiv preprint arXiv:2405.11106*.
- Khanh-Tung Tran, Dung Dao, Minh-Duong Nguyen, Quoc-Viet Pham, Barry O’Sullivan, and Hoang D Nguyen. 2025. Multi-agent collaboration mechanisms: A survey of llms. *arXiv preprint arXiv:2501.06322*.
- Tom Vodopivec, Spyridon Samothrakis, and Branko Ster. 2017. On monte carlo tree search and reinforcement learning. *Journal of Artificial Intelligence Research*, 60:881–936.
- Junlin Wang, Jue Wang, Ben Athiwaratkun, Ce Zhang, and James Zou. 2024. Mixture-of-agents enhances large language model capabilities. *arXiv preprint arXiv:2406.04692*.
- Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.
- Yuan Wei, Xiaohan Shan, and Jianmin Li. 2025. Lero: Llm-driven evolutionary framework with hybrid rewards and enhanced observation for multi-agent reinforcement learning. *arXiv preprint arXiv:2503.21807*.
- Siwei Wu, Zhongyuan Peng, Xinrun Du, Tuney Zheng, Minghao Liu, Jialong Wu, Jiachen Ma, Yizhi Li, Jian Yang, Wangchunshu Zhou, and 1 others. 2024. A comparative study on reasoning patterns of openai’s o1 model. *arXiv preprint arXiv:2410.13639*.
- Qianqian Xie, Weiguang Han, Zhengyu Chen, Ruoyu Xiang, Xiao Zhang, Yueru He, Mengxi Xiao, Dong Li, Yongfu Dai, Duanyu Feng, and 1 others. 2024. Finben: A holistic financial benchmark for large language models. *Advances in Neural Information Processing Systems*, 37:95716–95743.
- Zhiwei Xu, Bin Zhang, Dapeng Li, Zeren Zhang, Guangchong Zhou, Hao Chen, and Guoliang Fan. 2023. Consensus learning for cooperative multi-agent reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 11726–11734.
- Hai Ye, Mingbao Lin, Hwee Tou Ng, and Shuicheng Yan. 2024. Multi-agent sampling: Scaling inference compute for data synthesis with tree search-based agentic collaboration. *arXiv preprint arXiv:2412.17061*.
- Shengbin Yue, Wei Chen, Siyuan Wang, Bingxuan Li, Chenchen Shen, Shujun Liu, Yuxuan Zhou, Yao Xiao, Song Yun, Xuanjing Huang, and 1 others. 2023. Disc-lawllm: Fine-tuning large language models for intelligent legal services. *arXiv preprint arXiv:2309.11325*.
- Xuanyu Zhang and Qing Yang. 2023. Xuanyuan 2.0: A large chinese financial chat model with hundreds of billions parameters. In *Proceedings of the 32nd ACM international conference on information and knowledge management*, pages 4435–4439.
- Wanjun Zhong, Ruixiang Cui, Yiduo Guo, Yaobo Liang, Shuai Lu, Yanlin Wang, Amin Saied, Weizhu Chen, and Nan Duan. 2023. Agieval: A human-centric benchmark for evaluating foundation models. *arXiv preprint arXiv:2304.06364*.
- Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and 1 others. 2022. Least-to-most prompting enables complex reasoning in large language models. *arXiv preprint arXiv:2205.10625*.

A Inference Cost

All of our experiments were conducted on NVIDIA A800 GPUs, and because we employ a multi-agents mixture approach, each inference requires two GPUs. The time required to generate an answer for a single instance is about 2 minutes for 10 rollouts and 2 minutes 50 seconds for 15 rollouts. Consequently, completing inference over an entire dataset (e.g., ECA or LJP) takes roughly 23.6 hours.

B Time Complexity Comparison

We show the time complexity calculation process and conclusions for MMA and MOA as follows: The time complexity of the MMA method we propose is approximately: $O(r \times d/2 \times 2t_{model})$, where r is the number of rollouts and d is the depth of the tree. $2t_{model}$ represents the time required for one model forward inference and one cross-verification. In our experimental comparison, we selected r as 15 and d as 5, resulting in a final value of approximately $O(75 \times t_{model})$.

The time complexity of the MOA method is as follows: $O((n_{sample} \times 2 + 1) \times t_{model})$. Here, n_{sample} is the total number of samples in the model. Ultimately, it is approximately $O(33 \times t_{model})$.

Although MMA incurs greater computational overhead, this overhead results in better inference performance, 44.49 vs. 53.68. That is, MMA is

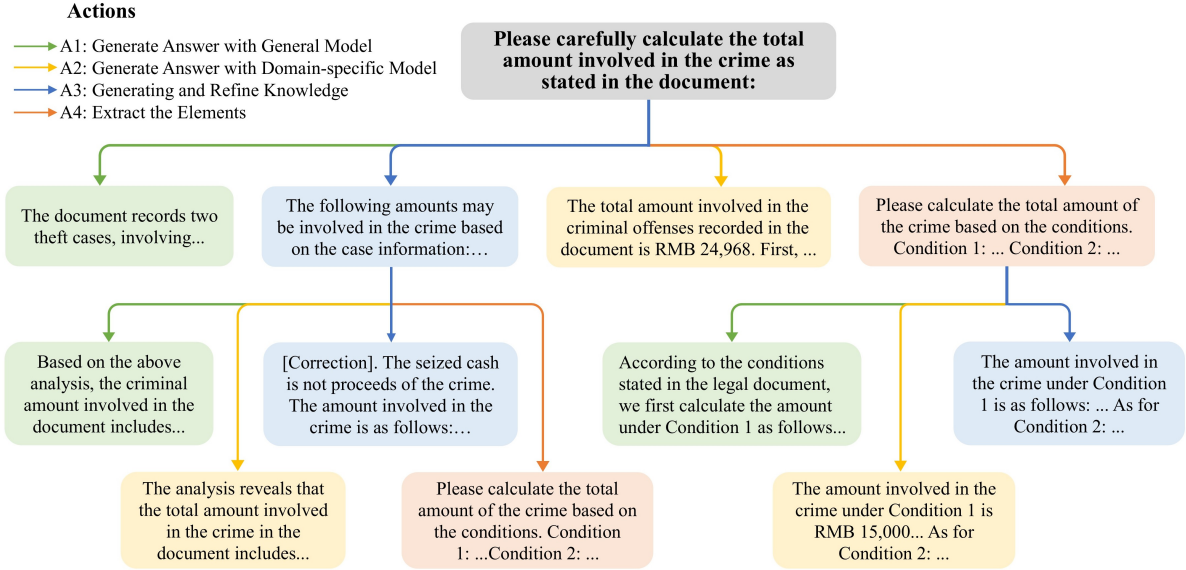


Figure 7: An example showing the extension of MCTS.

designed for scenarios that require high inference accuracy.

C Comparative Experiments and Analysis of Additional Effective Methods

C.1 Multi-Agent Collaboration.

Methods such as DyLAN enhances task adaptability by assigning diverse roles to a powerful general model (e.g., GPT-4o) and employing dynamic agent optimization, it fundamentally relies on multi-agent collaboration within the same base model. They rely on the capabilities of the professional knowledge systems inherent in powerful base models. When tasks involve professional fields that are not fully covered by base models, role simulation alone is often insufficient to compensate for knowledge gaps and effectively improve performance. In contrast, our proposed MMA framework overcomes these limitations by enabling dynamic integration of a general model and a domain-specific model at test time. This design effectively alleviates the limitations of base models, which often stem from inadequate knowledge coverage in unfamiliar or highly specialized fields. MMA requires only one general and one domain-specific model to achieve enhanced performance on domain-specific tasks, facilitating the dynamic integration of generalization and domain knowledge across models.

To ensure fairness, we used DeepSeek as the model when implementing DyLAN. The experi-

Method	ECA	General
LawLLM	58.80	27.68
DeepSeek	93.40	84.50
MMA	96.20	83.84
DyLAN	92.40	81.99

Table 5: We present the comparison results between our framework and DyLAN.

mental results are as follows:

C.2 Retrieval-Augmented Generation (RAG)

RAG retrieves relevant text fragments from a corpus to assist language models in generating responses. Its core function is to provide external knowledge support, making it more suitable for “knowledge-intensive task”—that is, tasks where answers primarily rely on the direct extraction or summarization of facts and provisions. However, since the retrieved information itself lacks reasoning capabilities, RAG cannot substantially enhance the model’s reasoning ability. This limitation becomes especially apparent in tasks requiring logical inference, domain-specific judgment, or strategic decision-making, where RAG’s assistance is often insufficient. In contrast, MMA does not merely provide knowledge—it incorporates models with complementary strengths into the entire reasoning process, enabling a deep integration of knowledge and reasoning. As a result, MMA goes beyond knowledge supplementation; it fundamentally enhances the model’s overall capability.

Model	AGI	LJP	LQA	ECA
Llama-8B	39.73	16.40	29.88	56.20
LexRAG	40.33	24.32	32.71	51.00
MMA	43.44	58.68	42.01	70.60

Table 6: We compare RAG methods and our framework across different general and domain-specific tasks.

At the same time, we added a comparison experiment with the RAG method.

The results show that RAG can improve performance on tasks that heavily rely on domain expertise, such as LJP and LQA (where LJP requires providing professional legal judgments), it performs poorly on tasks like ECA that require reasoning (e.g., calculating the total amount of criminal proceeds), even reducing performance by 2.2%. However, our MMA framework achieves superior results in terms of consistency and reasoning capabilities across tasks and domains by dynamically combining general models and domain-specific models.

D Example

D.1 Action Space

We present a comprehensive action space with illustrative examples, as shown in Figure 7.

D.2 Prompt Examples in ECA

Action 1: Generate Answer with General Model

Instruction: Please carefully calculate the total amount involved in the crime. You must, according to my formatting requirements, first give the analysis and then, beginning with "Final amount:", give the concluding total.

Document: "Upon trial it was established that: On the afternoon of April 8, 2017, the defendant Wang X drove with Ma and others to Jiashan County in Jiaxing City and checked into the Vienna International Hotel. In the early hours of April 11, the defendant Wang X, in collusion with co-defendant Ma (being dealt with separately), took a taxi to Jiaxing City, pried open the door of victim Jiang M.'s home and stole one silver men's Omega wristwatch (valued at RMB 15,000), one silver ladies' Omega wristwatch (valued at RMB 13,000), and RMB 500 in cash. The two then went to Villa A4 in Cui xx, used the same method to break into victim

Yu M.'s home and stole approximately RMB 600 in cash, as well as about RMB 17,000 in cash left in the BMW bearing license plate Zhe Fxxxxx parked at the door. That same morning, defendant Wang X and his accomplices fled to Suzhou.

On the morning of April 14, 2017, the public security authorities arrested defendant Wang X in Room 1228 of the Vienna Hotel in Suzhou City, and detained co-occupants Yang M. and Jiao M.; Ma escaped from the room. Police seized from Wang X cash of RMB 421, a pair of black sneakers, gloves, a flashlight, masks, and other items. It was further established that defendant Wang X had been released on September 27, 2016, had served his previous sentence without commutation, and had already fulfilled his property penalty."

Analysis: On the afternoon of April 8, 2017, Wang X and Ma et al. drove to Jiashan County, Jiaxing City, and checked into the Vienna International Hotel. In the early hours of April 11, Wang X and co-defendant Ma took a taxi to Ji-axing City, pried open the door of victim Jiang's home and stole one silver men's Omega watch (worth RMB 15,000), one silver ladies' Omega watch (worth RMB 13,000), and RMB 500 in cash. The two then went to Villa A4 in Cui xx, used the same method to steal approximately RMB 600 in cash from victim Yu's residence and about RMB 17,000 in cash from a BMW (license plate Zhe Fxxxxx) parked outside. That same morning, Wang X and his accomplices fled to Suzhou. On the morning of April 14, 2017, public security officers arrested Wang X in Room 1228 of the Vienna Hotel in Suzhou, detained co-occupants Yang and Jiao (Ma escaped), and seized RMB 421 in cash, a pair of black sneakers, gloves, a flashlight, masks, and other items from him.

Your answer format should be: "Answer: [analysis]. Final amount: [final amount]" where [analysis] is the screening and calculation of the amounts, and [final amount] is the total criminal amount involved as set forth in the document.

According to the analysis, the amounts involved in the crime recorded in the document include: one silver men's Omega watch valued at RMB 15,000; one silver ladies' Omega watch valued at RMB 13,000; RMB 500 in cash; ap-

proximately RMB 6,000 in cash from victim Yu's home; and approximately RMB 17,000 in cash from the BMW parked at the door. The cash sums to RMB 6,000 + RMB 17,000 = RMB 20,000, and the watches sum to RMB 15,000 + RMB 13,000 = RMB 28,000. The total amount is RMB 28,000 + RMB 20,000 = RMB 48,000. Final amount: RMB 48,000.

Action 2: Generate Answer with Domain-specific Model

Instruction: Please carefully calculate the total amount involved in the crime. You must, according to my formatting requirements, first give the analysis and then, beginning with "Final amount:", give the concluding total.

Document: "Upon trial it was established that: On the afternoon of April 8, 2017, the defendant Wang X drove with Ma and others to Jiashan County in Jiaxing City and checked into the Vienna International Hotel. In the early hours of April 11, the defendant Wang X, in collusion with co-defendant Ma (being dealt with separately), took a taxi to Jiaxing City, pried open the door of victim Jiang M.'s home and stole one silver men's Omega wristwatch (valued at RMB 15,000), one silver ladies' Omega wristwatch (valued at RMB 13,000), and RMB 500 in cash. The two then went to Villa A4 in Cui xx, used the same method to break into victim Yu M.'s home and stole approximately RMB 600 in cash, as well as about RMB 17,000 in cash left in the BMW bearing license plate Zhe Fxxxxx parked at the door. That same morning, defendant Wang X and his accomplices fled to Suzhou.

On the morning of April 14, 2017, the public security authorities arrested defendant Wang X in Room 1228 of the Vienna Hotel in Suzhou City, and detained co-occupants Yang M. and Jiao M.; Ma escaped from the room. Police seized from Wang X cash of RMB 421, a pair of black sneakers, gloves, a flashlight, masks, and other items. It was further established that defendant Wang X had been released on September 27, 2016, had served his previous sentence without commutation, and had already fulfilled his property penalty."

Your answer format should be: "Answer: [analysis]. Final amount: [final amount]" where [analysis] is the screening and calculation of the amounts, and [final amount] is the total criminal amount involved as set forth in the document. ### Response: The total amount involved in the crime as recorded in the document is RMB 24,968. First, defendant Wang X stole from victim Jiang one silver men's Omega watch (valued at RMB 15,000), one silver ladies' Omega watch (valued at RMB 13,000), and RMB 500 in cash, as well as from victim Yu approximately RMB 600 in cash and about RMB 17,000 in cash left inside the BMW (license plate Zhe Fxxxxx) parked at the door. Second, when Wang X and his accomplices fled to Suzhou and were arrested, the police seized RMB 421 in cash from him. Therefore, the total criminal amount is RMB 24,968. Final amount: RMB 24,968.

Action 3: Generate and Refine Knowledge

Instruction: Please analyze the following document, identifying each summarized criminal act and the amount involved, without providing the total criminal amount.

Document: "Upon trial it was established that: On the afternoon of April 8, 2017, the defendant Wang X drove with Ma and others to Jiashan County in Jiaxing City and checked into the Vienna International Hotel. In the early hours of April 11, the defendant Wang X, in collusion with co-defendant Ma (being dealt with separately), took a taxi to Jiaxing City, pried open the door of victim Jiang M.'s home and stole one silver men's Omega wristwatch (valued at RMB 15,000), one silver ladies' Omega wristwatch (valued at RMB 13,000), and RMB 500 in cash. The two then went to Villa A4 in Cui xx, used the same method to break into victim Yu M.'s home and stole approximately RMB 600 in cash, as well as about RMB 17,000 in cash left in the BMW bearing license plate Zhe Fxxxxx parked at the door. That same morning, defendant Wang X and his accomplices fled to Suzhou.

On the morning of April 14, 2017, the public security authorities arrested defendant Wang X in Room 1228 of the Vienna Hotel in Suzhou City,

and detained co-occupants Yang M. and Jiao M.; Ma escaped from the room. Police seized from Wang X cash of RMB 421, a pair of black sneakers, gloves, a flashlight, masks, and other items. It was further established that defendant Wang X had been released on September 27, 2016, had served his previous sentence without commutation, and had already fulfilled his property penalty."

Your answer format should be: "Analysis: [elements]", where [elements] are your analysis details of the document.

Response: On the afternoon of April 8, 2017, Wang X and Ma drove to Jiashan County, Jiaxing City, and checked into the Vienna International Hotel. In the early hours of April 11, Wang X and co-defendant Ma took a taxi to Jiaxing City, pried open the door of victim Jiang's home and stole one silver men's Omega watch (valued at RMB 15,000), one silver ladies' Omega watch (valued at RMB 13,000), and RMB 500 in cash. They then went to Villa A4 in Cui xx, used the same method to break into victim Yu's home and stole approximately RMB 600 in cash and about RMB 17,000 in cash from a BMW (license plate Zhe Fxxxxx) parked outside. That same morning, Wang X and his accomplices fled to Suzhou. On the morning of April 14, 2017, public security officers arrested Wang X in Room 1228 of the Vienna Hotel in Suzhou, detained co-occupants Yang and Jiao (Ma escaped), and seized RMB 421 in cash, a pair of black sneakers, gloves, a flashlight, masks, and other items from Wang X.

Instruction: The following draft answer calculates the criminal amount in the document based on the original analysis. Please review the reasonableness and accuracy of the original analysis according to the draft answer, and then generate the review result in accordance with the review requirements and output format.

Document: "Upon trial it was established that: On the afternoon of April 8, 2017, the defendant Wang X drove with Ma and others to Jiashan County in Jiaxing City and checked into the Vienna International Hotel. In the early hours of April 11, the defendant Wang X, in collusion

with co-defendant Ma (being dealt with separately), took a taxi to Jiaxing City, pried open the door of victim Jiang M.'s home and stole one silver men's Omega wristwatch (valued at RMB 15,000), one silver ladies' Omega wristwatch (valued at RMB 13,000), and RMB 500 in cash. The two then went to Villa A4 in Cui xx, used the same method to break into victim Yu M.'s home and stole approximately RMB 600 in cash, as well as about RMB 17,000 in cash left in the BMW bearing license plate Zhe Fxxxxx parked at the door. That same morning, defendant Wang X and his accomplices fled to Suzhou.

On the morning of April 14, 2017, the public security authorities arrested defendant Wang X in Room 1228 of the Vienna Hotel in Suzhou City, and detained co-occupants Yang M. and Jiao M.; Ma escaped from the room. Police seized from Wang X cash of RMB 421, a pair of black sneakers, gloves, a flashlight, masks, and other items. It was further established that defendant Wang X had been released on September 27, 2016, had served his previous sentence without commutation, and had already fulfilled his property penalty."

Analysis: On the afternoon of April 8, 2017, Wang X and Ma et al. drove to Jiashan County, Jiaxing City, and checked into the Vienna International Hotel. In the early hours of April 11, Wang X and co-defendant Ma took a taxi to Ji-axing City, pried open the door of victim Jiang's home and stole one silver men's Omega watch (worth RMB 15,000), one silver ladies' Omega watch (worth RMB 13,000), and RMB 500 in cash. The two then went to Villa A4 in Cui xx, used the same method to steal approximately RMB 600 in cash from victim Yu's residence and about RMB 17,000 in cash from a BMW (license plate Zhe Fxxxxx) parked outside. That same morning, Wang X and his accomplices fled to Suzhou. On the morning of April 14, 2017, public security officers arrested Wang X in Room 1228 of the Vienna Hotel in Suzhou, detained co-occupants Yang and Jiao (Ma escaped), and seized RMB 421 in cash, a pair of black sneakers, gloves, a flashlight, masks, and other items from him.

Answer pending trial: According to the analysis, the amounts involved in the crime recorded in

the document include: one silver men's Omega watch valued at RMB 15,000; one silver ladies' Omega watch valued at RMB 13,000; RMB 500 in cash; approximately RMB 6,000 in cash from victim Yu's home; and approximately RMB 17,000 in cash from the BMW parked at the door. The cash sums to RMB 6,000 + RMB 17,000 = RMB 20,000, and the watches sum to RMB 15,000 + RMB 13,000 = RMB 28,000. The total amount is RMB 28,000 + RMB 20,000 = RMB 48,000. Final amount: RMB 48,000.

Review requirements: 1. If there are any flaws or errors in the draft answer or the original analysis, begin with [Correction] and provide a complete and correct analysis, without explaining the cause of the error. 2. If it is completely correct, begin with [No Change Needed].

Output format: "[Review Conclusion]. Analysis: [elements]." Where [Review Conclusion] is either [Correction] or [No Change Needed], and [elements] is the complete problem analysis (i.e., each summarized criminal act in the document along with the amount involved; the analysis should not provide the total criminal amount).

Response: On the afternoon of April 8, 2017, Wang X and Ma, along with others, drove to Jiashan County, Jiaxing City, and checked into the Vienna International Hotel. In the early hours of April 11, Wang X and co-defendant Ma took a taxi to Jiaxing City and used the method of prying open doors to enter the residence of victim Jiang, from which they stole one silver men's Omega watch (valued at RMB 15,000), one silver ladies' Omega watch (valued at RMB 13,000), and RMB 500 in cash. The two then went to Villa A4 in Cui xx and used the same method to steal approximately RMB 6,008 in cash from victim Yu's home and about RMB 17,000 in cash from a BMW (license plate Zhe Fxxxx) parked at the entrance. That same morning, Wang X and the others fled to Suzhou. On the morning of April 14, 2017, public security authorities arrested Wang X in Room 128 of the Vienna Hotel in Suzhou City, and found co-occupants Yang and Jiao in the room, while Ma escaped. Police seized RMB 421 in cash, a pair of black sneakers, gloves, a flashlight, masks, and other items from Wang X.

Action 4: Extract the Elements

Instruction: Please split the context of the document into distinct conditions and help me rephrase the text. In the rephrased document, be sure to fully convey all information from the original. Only translate the above into English. Document: "Upon trial it was established that: On the afternoon of April 8, 2017, the defendant Wang X drove with Ma and others to Jiashan County in Jiaxing City and checked into the Vienna International Hotel. In the early hours of April 11, the defendant Wang X, in collusion with co-defendant Ma (being dealt with separately), took a taxi to Jiaxing City, pried open the door of victim Jiang M.'s home and stole one silver men's Omega wristwatch (valued at RMB 15,000), one silver ladies' Omega wristwatch (valued at RMB 13,000), and RMB 500 in cash. The two then went to Villa A4 in Cui xx, used the same method to break into victim Yu M.'s home and stole approximately RMB 600 in cash, as well as about RMB 17,000 in cash left in the BMW bearing license plate Zhe Fxxxxx parked at the door. That same morning, defendant Wang X and his accomplices fled to Suzhou.

On the morning of April 14, 2017, the public security authorities arrested defendant Wang X in Room 1228 of the Vienna Hotel in Suzhou City, and detained co-occupants Yang M. and Jiao M.; Ma escaped from the room. Police seized from Wang X cash of RMB 421, a pair of black sneakers, gloves, a flashlight, masks, and other items. It was further established that defendant Wang X had been released on September 27, 2016, had served his previous sentence without commutation, and had already fulfilled his property penalty."

Response: Condition 1: On the afternoon of April 8, 2017, defendant Wang X and Ma and others traveled to Jiashan County, Jiaxing City, and checked into the Vienna International Hotel.

Condition 2: In the early hours of April 11, Wang X together with co-defendant Ma (handled in a separate case) took a taxi to Jiaxing City, pried open the door of victim Jiang's home, and stole one silver men's Omega watch (valued at RMB 15,000), one silver ladies' Omega watch (valued at RMB 13,000), and

RMB 500 in cash.

Condition 3: At the same time, the two went to Villa A4 in Cui xx, used the same method to break into victim Yu's home, and stole approximately RMB 6,000 in cash and about RMB 17,000 in cash from a BMW (license plate Zhe Fxxxxx) parked at the entrance.

Condition 4: That same morning, Wang X and his accomplices fled to Suzhou.

Condition 5: On the morning of April 14, 2017, public security officers went to Room 1228 of the Vienna Hotel in Suzhou City and arrested Wang X, detaining co-occupants Yang M. and Jiao M., while Ma escaped from the room.

Condition 6: The police seized RMB 421 in cash, a pair of black sneakers, gloves, a flashlight, masks, and other items from Wang X.

Condition 7: It was further established that Wang X had been released on September 27, 2016, had served his prior sentence without commutation, and had already fulfilled his property penalty.