Bridging the Gap in Complex Claims: A Dual-Adaptive Multi-Agent Approach and Evaluation Benchmark for Chinese Insurance Reasoning

Anonymous ACL submission

Abstract

Insurance claims reasoning is a complex process that necessitates the integration of multi-source evidence while ensuring regulatory compliance and fairness. While Large Language Models (LLMs) show promise, existing evaluation systems lack the rigor for high-stakes scenarios with real economic and legal implications. To address this, we introduce **InsClaimQA**, the first clause-to-conclusion dataset for rigorous insurance claims reasoning. InsClaimQA features multi-difficulty grading, real-world derivations, expert annotations for legal traceability, and mandated explainable reasoning. To meet these high demands, we propose **DAMA**, a modular **D**ual-**A**daptive **M**ulti-**A**gent framework. DAMA uses specialized agents, context-aware routing, and a closed-loop quality control system to ensure reliable and transparent decisions. Evaluations confirm InsClaimQA's quality, with 98.7% accuracy and 0.96 RAGAs fidelity for explanations. DAMA significantly improves decision accuracy by 8.15% and reduces financial risk by 57.4%, proving its practical reliability in critical insurance applications. Code and data are available in the supplementary materials.

1 Introduction

Insurance claim processing requires careful review of evidence to verify claims and ensure fairness (Owens et al., 2022). For example, car claims compare witness accounts with vehicle data, while health claims check medical history against guidelines. Modern insurance claim systems use various data, such as policy records, claimant descriptions and third-party assessment reports (Devaraj, 2023), but the lack of explainability in black-box algorithmic models may lead to a crisis of trust among stakeholders(Hassija et al., 2024; Gonzalez, 2024).

Recent advances in Large Language Models (LLMs) offer new approaches for insurance claim reasoning (Troxler and Schelldorfer, 2024; Balona, 2024; Li et al., 2025). Advanced LLMs (Hurst et al., 2024; Liu et al., 2024a) efficiently parse structured claim clauses and unstructured descriptions to conduct multi-step causal inference. Their semantic understanding, rule mapping, and evidence reconstruction align well with compliance review and causal verification needs in claims (Zhang et al., 2023; Stanly and Aruna, 2024). However, the enterprise-specific challenges in insurance, including real-time policy updates and cross-functional evidence resolution, require organization-aligned algorithmic solutions to ensure scalable, compliant, and transparent decision-making.

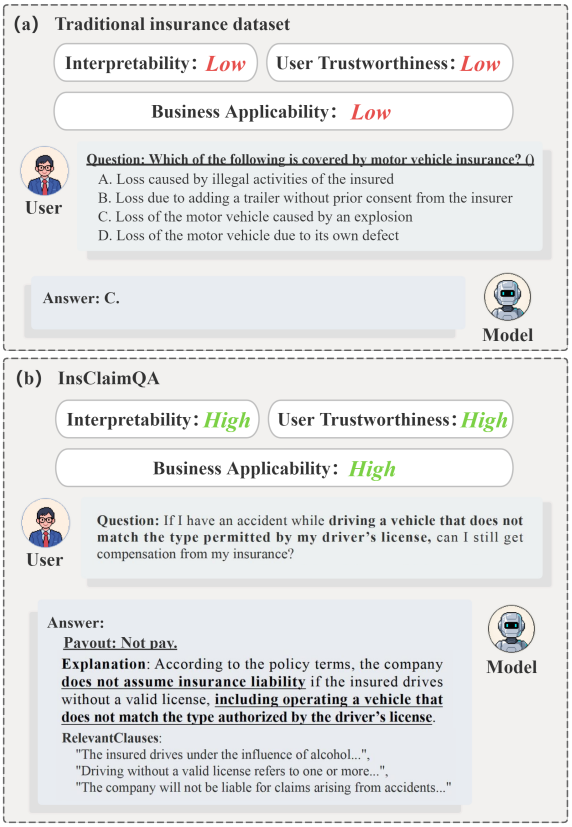

Figure 1: (a) Traditional insurance claim datasets predominantly consist of structured questionnaire responses. (b) Our dataset, derived from real-world operations, offers better interpretability, user trust, and business applicability.

While LLMs show potential for insurance claims reasoning, their real-world use is limited by insufficient evaluation frameworks. Existing datasets focus on rote-learning tasks like multiple-choice exams rather than practical complex reasoning (Koto, 2024; Troxler and Schelldorfer, 2024), ignoring high-stakes factors like legal accountability and ethical compliance. To address this, **we introduce InsClaimQA, the first clause-to-conclusion dataset for rigorous insurance claims reasoning.** Our key innovations include: **1) Rigorous Quality Control**: The dataset features a multi-difficulty grading system, real-world scenario derivation, and expert-driven annotation, including unclear cases to mirror authentic insurance complexities. **2) Legal Traceability**: Each case is annotated with verifiable legal clause references, requiring models to ground their decisions in specific policy terms. **3) Explainable Reasoning**: We define adjudication as a three-class classification task Pay, Not Pay, and Possibly Pay or Possibly Not Pay and mandate full reasoning chains, enabling quantitative evaluation of cross-clause reasoning and multi-criteria decision-making. Figure 1 contrasts InsClaimQA with traditional datasets.

To meet the high-stakes and high-explainability demands of insurance claims reasoning and cover edge cases, we propose **DAMA**, a modular **D**ual-**A**daptive **M**ulti-**A**gent collaborative reasoning framework with an agent resource pool comprising three agent types. **Clauses-Aware agents** handle pre-existing condition analysis via offline semantic decomposition of insurance clauses. **Query-aware agents** ensure timely, accurate clause interpretation through evidence retrieval and web searches. **Predefined agents** perform core functions such as exclusion clause verification and boundary analysis to prevent reasoning hallucinations. DAMA uses a context-aware routing mechanism to assign agents based on claim type, clause complexity and evidence availability. An independent multi-perspective scoring system evaluates output quality on clause citation completeness and reasoning coherence. If scores fall below a threshold, agents re-evaluate evidence and cross-validate results, forming a closed-loop for dynamic quality control.

For InsClaimQA, volunteer evaluations confirm the reliability of underwriting conclusions, achieving 98.7% accuracy and aligning with expert judgments. Explanations achieve a 0.96 RAGAs fidelity score (Es et al., 2024), aligning with clauses and ensuring logical coherence to validate the dataset's quality. In terms of model performance, DAMA utilizing only general-purpose LLMs without domain-specific fine-tuning achieves 90.11% claim decision accuracy, representing an 8.15% improvement over the single-reasoning baseline. We introduce Capital Loss Rate (CLR) to quantify financial risks from model misclassifications, such as incorrectly approving claims that should be denied or deemed uncertain. DAMA reduces CLR by 57.4% compared to baselines, demonstrating its practicality and reliability in high-stakes insurance scenarios.

The contributions of our work are as follows:

- We introduce InsClaimQA, the first dataset for rigorous insurance claims reasoning, meticulously crafted with multi-difficulty cases and expert annotations. It ensures legal traceability with verifiable clause references and demands explainable reasoning through a three-class classification with full reasoning chains.
- We propose DAMA, a novel dual-adaptive multi-agent framework designed for high-stakes insurance claims, utilizing specialized agents for robust edge case handling. Its modular design and closed-loop quality control ensure reliable and transparent decision-making.
- Evaluations confirm InsClaimQA's quality, achieving 98.7% accuracy and high RAGAs fidelity scores for explanations. DAMA significantly improves claim decision accuracy by 8.15% and reduces financial risk (CLR) by 57.4%, proving its practical reliability.

2 Related Work

2.1 Evaluation of Reasoning Abilities in LLM

As large language models(LLMs) advance rapidly, building a scientific evaluation system to accurately assess their reasoning abilities has become critical in AI research. Current evaluation datasets serve specialized purposes: AIME 2024(MAA, 2024), MATH-500(Lightman et al., 2023), and LiveCodeBench(Jain et al., 2024) focus on mathematical and programming tasks; MMLU(Hendrycks et al., 2020) and GPQA Diamond(Rein et al., 2023) target knowledge-intensive domains; AlpacaEval 2.0(Dubois et al., 2024) evaluates basic language skills, like fluency and grammar, and complex capabilities, like logical reasoning, common sense, and dialogue interactions; FRAMES(Krishna et al., 2024) specializes in assessing long-context understanding through multi-paragraph texts and diverse tasks. However, these datasets fail to capture the

complexity of high-stakes scenarios with real economic and legal risks, potentially leading developers to overlook issues like poor model interpretability in real-world applications.

2.2 General-purpose and Reasoning LLM

General-purpose LLM like Llama3(Grattafiori et al., 2024), GPT-4o(Hurst et al., 2024) and DeepSeek-V3(Liu et al., 2024a) demonstrate strong multimodal processing, knowledge integration, and specialized capabilities. Recently, large-scale reasoning models have gained prominence in AI. Among them, ChatGPT-o1(Jaech et al., 2024) enhances reasoning through chain-of-thought prompting, exploring multiple solutions via reasoning markers and breaking down complex problems for strategic improvements. Meanwhile, DeepSeek-R1(Guo et al., 2025) employs a multi-stage training system, combining reinforcement learning with cold-start data and group-relative policy optimization to boost reasoning efficiency with streamlined training templates and reward mechanisms. Even with advanced analytical skills, these models struggle with high-stakes tasks such as insurance claims, where cross-domain knowledge, ambiguous policy language, and many edge cases prevent precise or rigorous analysis.

2.3 Multi-Agent Collaboration Framework

Recent research has explored multi-agent collaboration frameworks to enhance LLM capabilities. A common approach, exemplified by CAMEL(Li et al., 2023), uses role-playing, where agents assume specialized roles, break tasks into sub-steps, and solve them collaboratively. The multi-agent debate framework(Du et al., 2023; Li et al., 2024; Liu et al., 2024b), involves agents independently solving tasks and refining responses through mutual reasoning to reach consensus. Simpler voting mechanisms(Wang et al., 2022; Lee et al., 2024), like relative majority voting, select outputs based on the highest vote count in a single round. However, these traditional frameworks rely on manually defined agent roles and quantities, requiring prior developer knowledge. This poses challenges in dynamic scenarios like insurance clause determination, where frequent updates to products, regulations, and market risks demand rapid adaptation. Fixed multi-agent setups may lead to delays or errors, disrupting insurance operations and customer experience.

3 InsClaimQA: A Comprehensive Dataset for Complex Insurance Claims Reasoning

We introduce InsClaimQA, a novel multi-difficulty question-answering dataset specifically designed to evaluate and advance insurance claims reasoning. In this section, we first define the insurance claims reasoning task. Then, we describe the construction process of the dataset in detail. Finally, we analyse the dataset in various ways.

3.1 Problem Formulation

Insurance companies need to decide whether to approve a claim based on the policyholder's account of the situation and the relevant insurance policy terms, requiring claims reasoning that produces both a conclusion and its corresponding explanation. We define this as a multiclass classification reasoning problem. Given the policyholder's claim description $S \in \mathcal{S}$ (usually a short text) and the complete policy terms $P \in \mathcal{P}$ (usually a long document), the system must predict the claim conclusion $y \in \mathcal{Y}$ (where $\mathcal{Y} = \{\text{pay}, \text{not pay}, \text{possibly pay or possibly not pay}\}$) and generate an explanation $E \in \mathcal{E}$ based on specific policy clauses $\{c_i\} \subseteq P$.

This task poses several challenges: the model needs to comprehend the ambiguous information in concise claim descriptions and the exact meaning of extensive policy documents, while simultaneously providing accurate decisions and comprehensible reasoning. The *possibly pay or possibly not pay* category is included to address the inherently unclear scenarios prevalent in insurance practice, such as cases with insufficient evidence or contradictory clauses, thus establishing this as a rigorous and significant task for critical situations.

Category Type	Category	Number of Questions
Question Difficulty	Easy	502
	Medium	572
	Difficult	541
Question Category	Insured's Health Status	470
	Obligations of the Insurer	87
	Occupational Risks	70
	Accidental Causes	491
	Medical Treatment Protocols	320
	Claim Limitations	127
	Borderline Scenarios	50

Table 1: Dataset Statistics: Question Difficulty and Question Category Distribution

3.2 Dataset Construction

We utilize a publicly available Chinese fact-based Q&A dataset from the Aliyun Tianchi Competition(AFAC2024 Challenge Group - Competition 2: Insurance Clause-Based Q&A)(Group, 2024). This dataset comprises the complete clauses of 245 insurance products along with corresponding fact-based question-answer pairs. Through careful filtering and systematic rewriting of these questions, we have developed the first multi-difficulty graded Q&A dataset specifically designed for insurance claims reasoning tasks in serious scenarios.

Question Generation We rigorously filter the original dataset to obtain 1,615 questions reconstructable as insurance claims scenarios. We select questions based on three criteria: mappable insurance scenarios, transformable causal logic for clause reasoning, and domain-specific semantics. We establish a novel three-tier difficulty classification system (detailed in Appendix A.2) to reconstruct the original fact-based questions into graded claims reasoning challenges. Simple-level questions explicitly contain clause keywords and permit direct answer derivation from single clauses without specialized knowledge. Medium-level problems demand multi-clause collaborative reasoning involving conditional judgments or basic causal chains. Difficult questions require understanding of fundamental insurance law principles, interpretation of clause ambiguities, and resolution of conflicts between exclusion clauses and coverage terms.

Answer Generation The answer generation process involves three domain experts who conduct comprehensive data labeling. For each question, they determine claim conclusions across three categories: pay, not pay, or possibly pay/not pay. They also construct clause-based reasoning explanations and identify up to three core supporting clauses per question. To ensure data quality, we implement a rigorous validation protocol that includes secondary review of inconsistent conclusions and multiple manual inspections of the finalized dataset.

For instance, the original question *"What disease might severe autoimmune hepatitis develop into?"* is reformulated as *"Should the insured be compensated if diagnosed with severe autoimmune hepatitis at a qualified hospital during the policy waiting period, according to the 'Tong You e-Life Critical Illness Insurance' clauses?"* with detailed reasoning: *"Per policy terms, severe autoimmune hepatitis is defined as a chronic necroinflammatory liver disease of unknown etiology, characterized by immune-mediated destruction of hepatocytes leading to hepatic inflammation and necrosis, potentially progressing to cirrhosis. This condition falls within coverage scope without being listed in exclusions, thus qualifying for compensation."* The claims conclusion is *"Pay"*.

3.3 Dataset Analysis

In this section, we analyze the The InsClaimQA Dataset from two dimensions: dataset overview and question complexity.

Dataset Overview Our Chinese insurance payout inference Q&A dataset, InsClaimQA, comprises 1,615 high-quality question-answer pairs derived from 197 insurance product clauses, with an average clause length of 24,320 words. The dataset spans five principal insurance categories (detailed in Appendix A.1), offering comprehensive coverage of mainstream risk scenarios: 1) medical insurance, 2) travel insurance, 3) accident insurance, 4) comprehensive insurance, and 5) pension insurance. This taxonomic breadth ensures representation across critical insurance domains while maintaining focus on high-stakes claims decisions.

Question Complexity The dataset features a balanced three-level difficulty distribution, as detailed in Table 1, with easy (31.1%), medium (35.4%), and difficult (33.5%) cases proportionally representing distinct cognitive demands in claims assessment. Separately, we identify seven core assessment dimensions characterizing insurance claims (categorized in Appendix A.3, examples in Appendix A.4 and A.5). These dimensions include three subjective factors: the insured's health status, policyholder obligation fulfillment, and occupational risks; and four objective determinants: medical treatment protocols, accidental causes, policy limitations, and borderline scenarios. The distribution of questions across these categories is further summarized in Table 1. This multidimensional structure preserves the authentic complexity of underwriting decisions while remaining amenable to computational modeling.

4 DAMA: Dual-Adaptive Multi-Agent Framework for Insurance Claims Reasoning

We introduce DAMA, a novel dual-adaptive multi-agent framework specifically designed for auto-

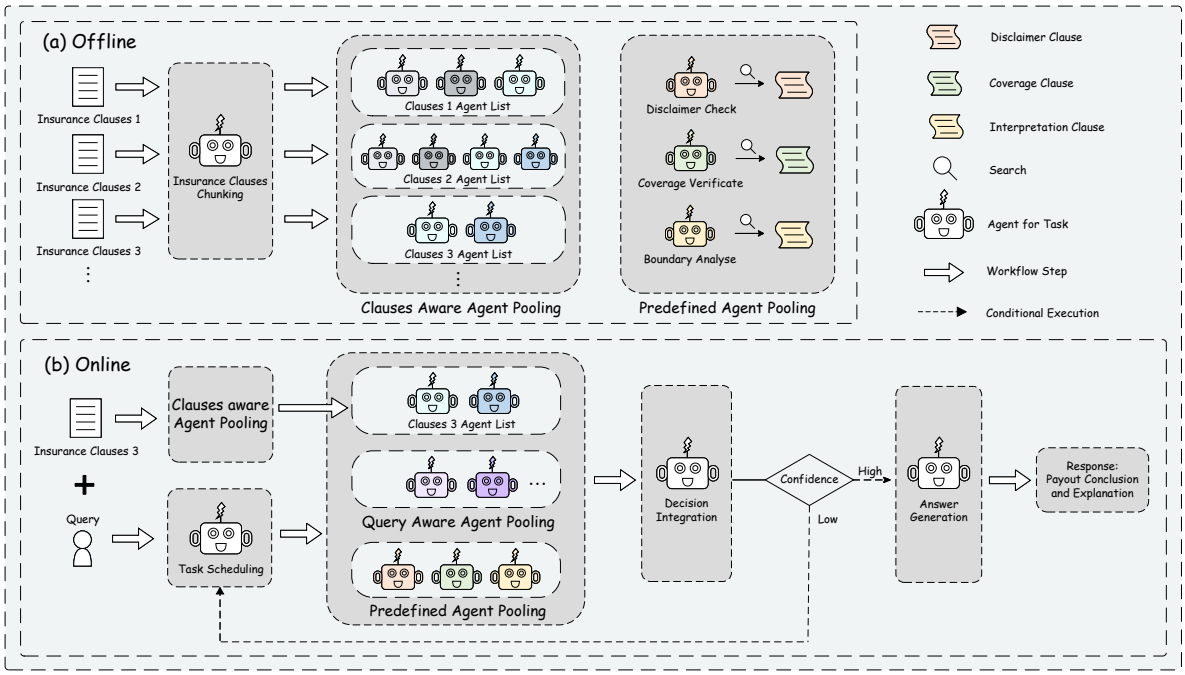

Figure 2: An overview of the proposed Dual-Adaptive Multi-Agent Framework for Insurance Claims Reasoning.

mated insurance claims reasoning. The framework comprises two primary phases: the offline deployment phase and the online reasoning phase. In the offline deployment phase, insurance clauses are systematically segmented, and a list of specialized agents is dynamically defined for each clause, thereby constructing a **Clauses-Aware Agent Pool**. Concurrently, a **Predefined Agent Pool** is established, containing agents designed to execute default tasks critical for the reasoning process. The online reasoning phase dynamically assembles agent configurations by integrating the Clauses-Aware Agent Pool, the Predefined Agent Pool, and a dynamically generated **Query-Aware Agent Pool** which is tailored to the specifics of a user's query. This integrated approach facilitates an automated reasoning workflow for policyholder inquiries, ultimately deriving claims conclusions and their corresponding explanations. An overview of the proposed framework is depicted in Figure 2.

4.1 Offline Deployment

During the offline deployment phase, the clause segmentation agent structurally segments the full text of each insurance product's clauses based on pre-designed prompts. The clause segmentation agent partitions these clauses into distinct semantic modules, as detailed in Appendix B.1. These modules include "policy name", "basic information", "insurance liability", "exclusion of liability", "interpretation clause", "other clauses", "full text" and "agent list." This segmentation provides a precise search scope for specialized sub-agents, enabling them to efficiently locate key information within confined textual boundaries. Concurrently, this clause segmentation agent dynamically defines a list of potential agents and their corresponding task specifications relevant to claim reasoning for each insurance product. These agents collectively form the Clauses-Aware Agent Pool. This pool facilitates the selection of pertinent agents for specific tasks during the subsequent online reasoning phase.

Furthermore, we establish a Predefined Agent Pool, consisting of agents engineered with specific prompts (detailed in Appendix B.4, B.5 and B.6) to execute essential tasks and judgments during online reasoning. These predefined agents are invoked by default and encompass three fundamental tasks: exclusion matching, coverage matching, and underwriting boundary analysis. The exclusion clause matching agent ascertains if a user's description triggers any exclusion clauses. The coverage matching agent employs multi-step reasoning to determine if the described scenario falls within the policy's coverage. The claims boundary analysis agent identifies missing information or contradictions in the user's description, such as unspecified hospital grades or incomplete accident

timestamps, thereby providing uncertainty explanations and analyses for special cases to inform subsequent decision-making.

4.2 Online Reasoning

The online reasoning phase encompasses four critical steps: dynamic routing, multi-agent collaboration, decision fusion, and explanation generation. This modular design enhances the accuracy of clause interpretation and, through collaborative reasoning among agents, effectively addresses the limitations of traditional end-to-end models concerning long-text comprehension and complex logical inference.

Dynamic Routing A routing agent, guided by a meticulously designed prompt (detailed in Appendix B.3), first analyzes user intent to dynamically generate a list of agents potentially required for the claims inference task based on the user query, forming the Query-Aware Agent Pool along with their respective task descriptions. Subsequently, this Query-Aware Agent Pool is combined with the offline-deployed Clauses-Aware Agent Pool and the Predefined Agent Pool. This consolidated information serves as a structured input for a large language model (LLM), which leverages its task planning capabilities to determine and output the specific combination of sub-agents to be activated, thereby achieving dynamic scheduling for collaborative multi-agent operation, as detailed in Appendix B.2.

Multi-Agent Collaboration Each sub-agent leverages an LLM, guided by customized prompt engineering, to perform its specialized function. To optimize framework efficiency, agents from the Predefined Agent Pool are executed by default to complete fundamental tasks. Furthermore, the system caches task descriptions from the dynamically generated agent pools and reuses cached results when similar tasks recur, reducing the average number of API calls. The outputs from all sub-agents are standardized into a structured JSON format, encompassing fields such as judgment results and supporting clauses, which provides consistent input for the subsequent decision fusion stage. This modular architecture facilitates the seamless integration of LLMs from various providers, showcasing the framework's compatibility and extensibility. Moreover, as each agent processes only specific subtasks, the risk of semantic confusion common in end-to-end models is significantly mitigated.

Decision Fusion The decision fusion agent integrates inputs from various antecedent agents. It initially converts the structured JSON outputs from sub-agents into natural language summaries. Subsequently, it constructs multi-turn conversational prompts (detailed in Appendix B.7) to guide an LLM in performing hierarchical reasoning. To enhance stability, the system employs confidence assessment for self-consistency checking, accepting results only when the confidence score surpasses a predefined threshold; otherwise, it triggers a route backtracking mechanism.

Explanation Generation Finally, the explanation generation agent synthesizes the conclusions and interpretations from all contributing agents (detailed in Appendix B.8). It formulates judgments and summaries that align with manual underwriting rules and outputs the final underwriting conclusion (pay, not pay, possibly pay or possibly not pay) along with an explanation grounded in the policy's terms and conditions. This process adheres to the stringent requirements of the insurance industry while preserving the fluency of LLM-generated text. This approach ensures that the decision-making process is compliant with insurance terms and enhances the credibility of the outcomes through transparent and interpretable reasoning paths.

5 Experiments

In this section, we first conduct extensive experiments to verify the effectiveness of our proposed framework. We begin by outlining the experimental setup, detailing the compared models, parameter settings, and evaluation metrics. Subsequently, we present the main results and provide a comprehensive analysis and case study, comparing our approach against existing inference frameworks, generic models, and specialized inference models. This includes a discussion of the strengths and limitations of each approach, supported by quantitative results and qualitative examples.

5.1 Experimental Setup

To assess the efficacy of our Dual-Adaptive Multi-Agent Framework, we benchmarked it against several prominent LLM reasoning methods, categorized as follows: 1) Few-shot General LLMs. 2) General LLMs with Chain-of-Thought(Wei et al., 2022) prompting and few-shot learning. 3) Few-shot Reasoning LLMs. Within our framework, we utilized Qwen-Long, DeepSeek-V3, and DeepSeek-

Methods		Accuracy↑	Precision↑	Recall↑	F1↑	CLR↓	Faithfulness↑
General LLM	GPT-4o	68.24	70.71	68.24	69.07	14.49	55.30
	Qwen-Long	80.34	73.90	70.52	71.40	7.21	64.72
	Qwen-2.5-72B-Instruct	81.19	75.73	73.34	72.44	10.74	65.70
	DeepSeek-V3	81.96	77.56	74.19	73.82	10.02	66.35
Reasoning LLM	GPT-o1	67.02	72.26	67.02	68.89	10.02	49.48
	Qwen3-30B-A3B	76.43	70.30	64.11	65.51	7.53	43.41
	DeepSeek-R1	80.98	74.54	71.96	72.47	7.45	51.17
General LLM with CoT	GPT-4o + CoT	72.89	73.40	72.89	73.11	9.34	57.02
	DeepSeek-V3 + CoT	82.55	83.18	80.85	81.97	9.11	68.63
Ours	DAMA(Qwen-Long)	86.23	78.56	76.89	77.71	5.07	68.29
	DAMA(DeepSeek-R1)	88.24	82.63	80.35	81.47	**3.07**	68.91
	DAMA(DeepSeek-V3)	**90.11**	**87.80**	**85.20**	**86.27**	3.42	**69.16**

Table 2: Results of different methods on InsClaimQA dataset. "CLR (Capital Loss Rate)" indicates the proportion of cases where the model incorrectly recommended payout when the correct decision should have been either no payout or uncertain payout. "Faithfulness(RAGAs metric)" indicates the correlation between the reasoning process and the original clauses.

R1 for LLM invocations, chosen due to their superior performance and lower capital loss rates among the evaluated general and inference models. Importantly, all selected models possess a sufficient context window to process the entire input, including the prompt, relevant insurance clauses, and the query, in a single call. The temperature parameter for all LLMs was set to 0.3 to ensure result stability and reduce stochasticity in the generated outputs.

5.2 Dataset Evaluation

The insurance underwriting inference task presents a dual challenge, requiring both accurate classification to determine the underwriting decision and coherent open-text generation to justify the conclusion based on relevant clauses. Consequently, we defined distinct evaluation metrics for the underwriting conclusions and explanations, respectively. For evaluating claim conclusions, we employed standard metrics including accuracy, precision, recall, and F1-score. We introduced a Capital Loss Rate (CLR) metric, representing the proportion of incorrect conclusions that would result in financial losses for the insurance company. This metric quantifies instances where the correct conclusion should be "not pay" or "possibly pay or possibly not pay" but the model incorrectly predicts "pay", with calculation details in Appendix C.

For evaluating the quality of the explanations, we employed fidelity metrics from the RAGAs framework(Es et al., 2024), which measure the correlation between the generated reasoning process and the original clauses. To ensure the dataset's reliability, three volunteers assessed the dataset's claim conclusions and explanations. The average accuracy of the volunteers' claim conclusions was 98.7%, indicating the dataset's high quality. Similarly, the annotators scored the correlation between the explanations and the original text on a scale from 0 to 1, with an average score of 0.96, demonstrating the reasonableness and coherence of the explanations.

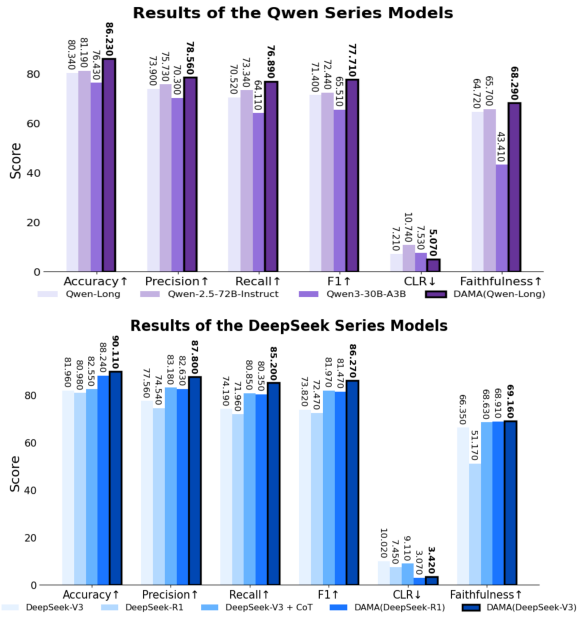

Figure 3: The comparison of various models in the Qwen and DeepSeek series.

5.3 Main Results

Table 2 presents a comprehensive comparison of DAMA against various baselines.

Superior Performance Over General LLMs DAMA consistently outperforms general LLMs, both with and without Chain-of-Thought (CoT) prompting, across all metrics including accuracy, precision, recall, F1-score, capital loss rate, and faithfulness. For example, DeepSeek-V3 within our framework achieves significantly higher accuracy (90.11%) and lower CLR (3.42%) compared to standalone DeepSeek-V3 (81.96% accuracy, 10.02% CLR) and its CoT variant (82.55% accuracy, 9.11% CLR), demonstrating the substantial benefits of our structured reasoning approach.

Enhancement of Reasoning LLMs While reasoning-oriented LLMs inherently exhibit better performance than general LLMs, integrating them into our framework further amplifies their capabilities. DeepSeek-R1's accuracy increases from 80.98% to 88.24%, and its CLR decreases from 7.45% to 3.07% within our framework. This highlights the framework's ability to leverage and enhance the strengths of specialized reasoning models.

Model-Agnostic Framework Benefits The framework demonstrates its model-agnostic nature by consistently delivering superior results across different underlying LLMs, including Qwen-Long, DeepSeek-R1 and DeepSeek-V3. It reduces CLR and increases faithfulness across these models. For example, Qwen-Long's CLR decreases from 7.21% to 5.07%, and its faithfulness increases from 64.72% to 68.29% within the framework. The consistent performance improvement, regardless of the base model, underscores the framework's generalizability and its ability to enhance diverse models through a standardized application without compromising its advantages in complex reasoning tasks. These results collectively highlight the framework's utility in advancing reasoning capabilities across various agent configurations within the challenging insurance underwriting context.

Intra-series Model Performance Comparison As demonstrated in Figure 3 and for positive metrics like Accuracy, Precision, Recall, F1-score, and Faithfulness, in both Qwen and DeepSeek series models, DAMA-integrated versions generally outperform original models. For the negative metric of Capital Loss Rate (CLR), DAMA shows a significant reduction, indicating enhanced risk control. Overall, DAMA optimizes both series on multiple key metrics, demonstrating its effectiveness in improving model performance.

Configuration	Accuracy	F1-Score	CLR	Faithfulness
DAMA$_{\textit{No Disclaimer Agent}}$	84.47	81.14	6.50	51.34
DAMA$_{\textit{No Coverage Agent}}$	83.64	80.69	5.87	51.92
DAMA$_{\textit{No Boundary Agent}}$	75.33	60.87	12.76	50.90
Full Framework	**90.11**	**86.27**	**3.42**	**69.16**

Table 3: Ablation Study: Impact of Agent Removal on Performance

5.4 Detailed Analysis

Table 3 presents the results of our ablation study, which examines the contribution of each agent type to the framework's overall performance.

Impact of Agent Removal Removing any of the Disclaimer, Coverage, or Boundary agents leads to a significant performance decrease, highlighting the importance of each agent type. The full agent configuration achieves the highest accuracy (90.11%), F1-score (86.27%), and faithfulness (69.16%), with the lowest capital loss rate (3.42%).

Specific Agent Contributions Excluding the Disclaimer agent significantly reduces faithfulness (to 51.34), indicating its crucial role in maintaining interpretability. Removing the Coverage agent lowers accuracy and F1-score, suggesting its importance for comprehensive reasoning. Notably, removing the Boundary agent results in the most drastic performance drop, underscoring its pivotal function in defining reasoning limits and ensuring coherent arguments. The ablation study confirms the synergistic interdependence of all agent components in maximizing the framework's effectiveness.

6 Conclusions

In this paper, we present the insurance claims reasoning task for large language models for the first time. To evaluate the insurance claims reasoning abilities of large language models, we construct a dataset, InsClaimQA, the first high-explainability dataset for evaluating complex insurance claims reasoning and addressing legal, economic, and ethical gaps in existing benchmarks. Based on the dataset, we propose a novel modular dual-adaptive multi-agent framework. This framework enables semantic clause decomposition and dynamic agent allocation through dynamic generation and context-aware routing, enhancing accuracy and explainability. Experiments demonstrate InsClaimQA's strong clause matching and logical integrity. Our framework achieves state-of-the-art performance, improving payout judgment accuracy by 7.56% and substantially mitigating capital loss rate compared to baselines.

Limitations

Our Chinese-language InsClaimQA dataset, while comprehensive, may not fully capture the diversity of all real-world insurance claims in other languages due to its inherent scope. Future work should expand the dataset with more insurance products, claim types, and real-world data to improve generalizability.

There is room for improvement in our method. The dual-adaptive multi-agent framework's reliance on pre-defined agent pooling and clause-aware routing may limit its adaptability to unforeseen claim scenarios. Its effectiveness depends on clause decomposition and agent role quality. Future research should explore more flexible agent allocation methods like reinforcement learning to enhance robustness.

Ethical Considerations

This paper introduces an insurance claims reasoning dataset constructed from a publicly available source. The original dataset is openly accessible, and our data processing steps involved anonymization of real claim descriptions by using pronouns such as "the policyholder" or "my" thus ensuring no privacy-sensitive information is included. Consequently, this study does not raise significant ethical concerns regarding data privacy.

References

Caesar Balona. 2024. Actuarygpt: Applications of large language models to insurance and actuarial work. *British Actuarial Journal*, 29:e15.

Surendra Mohan Devaraj. 2023. Ai and cloud for claims processing automation in property and casualty insurance. *International Journal Of Engineering And Technology Research (Ijetr)*, 8(1):38–46.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. In *Forty-first International Conference on Machine Learning*.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B Hashimoto. 2024. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*.

Shahul Es, Jithin James, Luis Espinosa Anke, and Steven Schockaert. 2024. Ragas: Automated evaluation of retrieval augmented generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 150–158.

Victoria Gonzalez. 2024. Evaluating interpretable models for financial fraud detection.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.

AFAC2024 Challenge Group. 2024. Aliyun tianchi competition: Insurance clause-based q&a. https://tianchi.aliyun.com/competition/entrance/532194/information. Accessed: 2024-03-15.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, and 1 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Vikas Hassija, Vinay Chamola, Atmesh Mahapatra, Abhinandan Singal, Divyansh Goel, Kaizhu Huang, Simone Scardapane, Indro Spinelli, Mufti Mahmud, and Amir Hussain. 2024. Interpreting black-box models: a review on explainable artificial intelligence. *Cognitive Computation*, 16(1):45–74.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, and 1 others. 2024. Openai o1 system card. *arXiv preprint arXiv:2412.16720*.

Naman Jain, King Han, Alex Gu, Wen-Ding Li, Fanjia Yan, Tianjun Zhang, Sida Wang, Armando Solar-Lezama, Koushik Sen, and Ion Stoica. 2024. Livecodebench: Holistic and contamination free evaluation of large language models for code. *CoRR*, abs/2403.07974.

Fajri Koto. 2024. Cracking the code: Multi-domain llm evaluation on real-world professional exams in indonesia. *arXiv preprint arXiv:2409.08564*.

Satyapriya Krishna, Kalpesh Krishna, Anhad Mohananey, Steven Schwarcz, Adam Stambler, Shyam Upadhyay, and Manaal Faruqui. 2024. Fact, fetch, and reason: A unified evaluation of retrieval-augmented generation. *CoRR*, abs/2409.12941.

Dongjun Lee, Choongwon Park, Jaehyuk Kim, and Heesoo Park. 2024. Mcs-sql: Leveraging multiple prompts and multiple-choice selection for text-to-sql generation. *arXiv preprint arXiv:2405.07467*.

Dongchen Li, Zhuo Jin, Linyi Qian, and Hailiang Yang. 2025. Textual analysis of insurance claims with large language models. *Journal of Risk and Insurance*.

Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. 2023. Camel: Communicative agents for" mind" exploration of large scale language model society.

Yunxuan Li, Yibing Du, Jiageng Zhang, Le Hou, Peter Grabowski, Yeqing Li, and Eugene Ie. 2024. Improving multi-agent debate with sparse communication topology. *arXiv preprint arXiv:2406.11776*.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024a. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Tongxuan Liu, Xingyu Wang, Weizhe Huang, Wenjiang Xu, Yuting Zeng, Lei Jiang, Hailong Yang, and Jing Li. 2024b. Groupdebate: Enhancing the efficiency of multi-agent debate using group discussion. *arXiv preprint arXiv:2409.14051*.

MAA. 2024. American invitational mathematics examination - aime. In *American Invitational Mathematics Examination - AIME 2024*.

Emer Owens, Barry Sheehan, Martin Mullins, Martin Cunneen, Juliane Ressel, and German Castignani. 2022. Explainable artificial intelligence (xai) in insurance. *Risks*, 10(12):230.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R Bowman. 2023. GPQA: A graduate-level google-proof q&a benchmark. *arXiv preprint arXiv:2311.12022*.

Anupa Stanly and K Aruna. 2024. Autonomous systems revolutionizing health insurance industry: Achieving operational excellence in services. In *Modeling, Simulation, and Control of AI Robotics and Autonomous Systems*, pages 131–151. IGI Global.

Andreas Troxler and Jürg Schelldorfer. 2024. Actuarial applications of natural language processing using transformers: Case studies for using text features in an actuarial context. *British Actuarial Journal*, 29:e4.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

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.

Wen Zhang, Jingwen Shi, Xiaojun Wang, and Henry Wynn. 2023. Ai-powered decision-making in facilitating insurance claim dispute resolution. *Annals of Operations Research*, pages 1–30.

A Detailed Dataset Information

To enhance the clarity and understanding of the InsClaimQA dataset, we provide additional detailed information regarding its characteristics and construction.

A.1 Insurance Categories and Examples

Table 4 presents a comprehensive overview of the five principal insurance categories included in the InsClaimQA dataset. This categorization ensures broad coverage of mainstream risk scenarios, reflecting the diverse types of insurance products relevant to claims reasoning. The table further provides specific examples for each category, illustrating the breadth of product types covered.

A.2 Insurance Claims Question Difficulty Grading Standards

The criteria for classifying the difficulty of insurance claims questions are detailed in Table 5. This table outlines the key characteristics that define each of the three difficulty levels: Simple, Medium, and Difficult. Examples are provided for each level to illustrate the varying cognitive demands and reasoning complexities involved in answering questions across the dataset.

A.3 Insurance Claims Question Category

Table 6 delineates the seven core assessment dimensions used to categorize insurance claims questions within the InsClaimQA dataset. These dimensions are broadly divided into subjective and objective factors, capturing the multifaceted nature of real-world underwriting and claims decisions. Each category includes a concise description of its focus.

A.4 Subjective Factors Examples

Specific examples illustrating the subjective factors identified in Table 6 are provided in Table 7. These examples include detailed scenarios related to the insured's health status and medical history, obligations of the insured, and occupational and behavioral risks, demonstrating the real-world scenarios captured by these categories.

A.5 Objective Factors Examples

Table 8 provides concrete examples for the objective factors that characterize insurance claims questions. These examples elucidate scenarios related to medical behaviors and treatment methods, accidents and external causes, claim conditions and limits, and special scenarios and edge cases, further illustrating the nuanced complexities within the dataset.

B Agent Prompt Configurations

This appendix details the prompt configurations for various intelligent agents employed in our system. Each subsection outlines the purpose and prompt structure of a specific agent, including Clause Segmentation, Routing, Dynamic Agent Definition, Exclusion Handling, Coverage Assurance, Boundary Constraints, Decision Making, and Explanation Generation. The prompt details illustrate how each agent is instructed to perform its designated task.

B.1 Clause Segmentation Agent

This agent processes raw insurance policy text to segment it into structured categories. It strictly adheres to a predefined set of rules, ensuring the original text's integrity within each section, preventing subsection splits, and merging similar content. Crucially, it differentiates coverage and exclusion clauses based on various application rules specified in the policy, such as distinct regulations for ordinary versus high-risk activities. The agent outputs the segmented information in a JSON format, which includes the policy name, basic information, liability (coverage), exclusions of liability, other clauses, and a suggested list of specialized agents for further claim verification based on the segmented content. An example of the prompt content for this agent is illustrated in Figure 4.

B.2 Router Agent

This agent analyzes user insurance claim questions, referred to as the Query, and selects the necessary sub-agents to handle the question. The selection process depends on whether a predefined list of available agents exists for the specific insurance policy. An illustration of the prompt content for the Router Agent in both scenarios is provided in Figure 5.

Scenario 1: Using Available Agent List. When a list of available agents is provided for the policy, the Router Agent analyzes the user's question and selects the relevant agents from this list.

Scenario 2: Using Default Logic. If no specific list of available agents is found, the Router Agent employs a default logic to determine the necessary sub-agents based on the question's content and the inherent requirements of insurance claim

Clause Segmentation Agent Prompt

System_prompt:
The clause content I want to process is: {data}. Please process it strictly according to the following rules:

Processing principles:
Maintain complete original text under each category without any rewriting (including line breaks and punctuation)
Do not split subsections or create nested structures
Merge same-type content into the same text block
Must distinguish completely different coverage rules, such as "Ordinary Sports Insurance Benefit Rules" vs "High-Risk Sports Insurance Benefit Rules"

Output format:
{
"policy_name": "Insurance policy product name",
"basic_Information": "Basic contract information including insurer, application rules, effective time (contains: insurer statement, eligibility, policy activation conditions, contact info, etc.)",
"liability": "Insurance coverage, complete text of all coverage descriptions. Extract according to various coverage rules in the product - different rules have different liabilities! (Contains: all coverage scenarios, payout standards, insured amounts, etc.)",
"exclusion_of_liability": "Complete text of all exclusion clauses. Extract according to various coverage rules - different rules have different exclusions! (Contains: all non-payout scenarios and explanations)",
"other_clauses": "Definition clauses, complete text of all term explanations in the content (contains: professional term definitions, special notes, etc.)",
"agent_list": "Based on clause content, provide several agents closely related to current clauses for more detailed claims verification"
}

Execution requirements:
Each category field stores complete original text of corresponding type
Leave empty string if a category has no content
Retain section headings from original text in the content
Do not add summaries, keywords or any derivative content - only provide JSON format output!
For coverage and exclusions sections, analyze according to coverage rules in the policy terms (e.g. sports insurance: "liability: Ordinary Sports Insurance Benefit Rules... High-Risk Sports Insurance Benefit Rules...")
Also provide an agent list based on clause segmentation, defining agents for detailed claims verification.

Special handling:
If a large text block contains multiple category contents (e.g. both exclusions and claims instructions), prioritize the category with larger proportion
Mark legal reference clauses with ※※Law Name Article X※※
Retain numbered lists and formatting symbols (e.g. •, ■) from original text
Core requirements: Strictly distinguish between "Coverage Scope" and "Exclusions" as two core sections. Identify all application types (e.g. ordinary/high-risk sports insurance) as secondary classification dimensions. For each application type, must extract both corresponding coverage details and exclusion clauses!!! The full insurance text may contain different application options corresponding to different coverage scopes and exclusions - all must be segmented out.
Begin processing the insurance clause text provided by user, output strictly according to format.

User_prompt:
Clause_text

Example output:
{
"policy_name": "Aviation Accident Insurance 2021 Edition",
"basic_Information": "Application Notice\nInsurance Period: This product's insurance period is 10 days or one year...\nInsurer: ZhongAn Online P&C Insurance Co., Ltd...",
"liability": "During insurance period, when insured is aboard civil aircraft...\nAccidental death benefit...\nAccidental disability benefit...",
"exclusion_of_liability": "Insured's claims resulting from...\n1) Applicant's intentional acts...\n17) During mental and behavioral disorders...",
"other_clauses": "After incident occurs, promptly call customer service...\nRequired materials:\n1) Claim application...",
"agent_list": "[\"Multi-Claim Coordinator: Handles differential compensation from other sources\", \"Document Reviewer: Strictly verifies completeness and authenticity of claim materials\", \"Timing Supervisor: Ensures claims process meets time limits\"]"
}

Figure 4: Example prompt content for the Clause Segmentation Agent, including the detailed System Prompt and a placeholder for the User Prompt containing the insurance clause text to be processed.

Router Agent Prompt

a) Using Available Agent List	***b) Using Default Logic***
System_prompt: You are an insurance claim routing agent. Your task is to analyze user questions and select the necessary sub-agents from the available agent list.	***System_prompt:*** You are an insurance claim routing agent. Your task is to analyze user questions and decide which sub-agents are needed to handle it. 【Task Requirements】 Analyze the user question content and select the necessary agents from the following: - "Exclusion Clause Matching": Select when the question might involve exclusion clauses. - "Coverage Matching": Select when the question involves coverage judgment. - "Claims Boundary Analysis": Select when the question might involve special circumstances. - "Evidence Retrieval": Select when specific clauses need to be cited. Must strictly output in the following JSON format, and only output JSON, without any explanations or extra text: ["Exclusion Clause Matching", "Coverage Matching"] Note: The default must include ["Exclusion Clause Matching", "Coverage Matching", "Claims Boundary Analysis"].
User_prompt: Query:Please strictly output the required agent list in the specified JSON format, and only output JSON:	*User_prompt:* Query:Please strictly output the required agent list in the specified JSON format, and only output JSON:
【Available Agents (Example)】 ["Exclusion Clause Matching", "Coverage Matching", "Claims Boundary Analysis", "Evidence Retrieval"]	【Available Agents (Example)】 ["Exclusion Clause Matching", "Coverage Matching", "Claims Boundary Analysis", "Evidence Retrieval"]

Figure 5: Prompt content for the Router Agent under two scenarios: a) with an available agent list and b) using default logic.

Dynamic Definition Agent Prompt

System_prompt:
You are an intelligent agent task definition generator. You need to create a task definition and prompt template for the new agent "{agent_name}".

【Task Requirements】

Define the specific task of the agent based on its name and the insurance clause content.
Generate the system prompt template required for this agent.
The output format must be JSON, containing the following fields:
"description": Detailed task description of the agent
"system_prompt": System prompt template, which can include {full_text} as a placeholder for the clause content
"output_format": Description of the expected output JSON format

User_prompt:
Please generate a task definition and prompt template for the agent '{agent_name}', which will handle insurance clause-related questions.

【Output Example】
{
"description": "This agent is responsible for...",
"system_prompt": "You are a...\n【Insurance Clause Content】\n{full_text}...",
"output_format": {
"field1": "Description",
"field2": "Description"
}
}

Figure 6: Example prompt content for the Dynamic Agent Definition Agent, including the System Prompt outlining the task and output format, and the User Prompt requesting the definition and template for a given agent name.

reasoning, ensuring that essential agents like "Exclusion Clause Matching","Coverage Matching" and "Claims Boundary Analysis Agent" are considered.

B.3 Dynamic Definition Agent

This agent is responsible for dynamically generating task definitions and prompt templates for new intelligent agents within the system. Given a new agent's name, specified as {agent_name} in both the System and User Prompts, and access to insurance clause content, this agent defines the new agent's specific purpose. It then creates a system prompt template, which can incorporate the full insurance text using the {full_text} placeholder, and specifies the expected JSON output format for the new agent. This capability enables the system to adapt and create specialized agents on demand to address various insurance-related tasks. An example of the prompt content for this agent is presented in Figure 6.

B.4 Exclusion Clause Matching Agent

This agent analyzes user-provided insurance claim scenarios, referred to as the Query, to determine if any exclusion clauses within the insurance policy are triggered. It compares the user's description, the Query, against the exclusion clauses extracted from the insurance chunk file, represented as {exclusion_text}. This agent performs a clause-by-clause analysis to provide a conclusion on whether an exclusion clause is triggered, along with an explanation. An example of the prompt content for this agent is shown in Figure 7.

Exclusion Clause Matching Agent Prompt

System_prompt:
You are an insurance claim analysis expert. Your task is to determine whether the situation described by the user triggers the exclusion clauses in the insurance policy.

【Task Requirements】
- Based on the content of the exclusion clauses, analyze whether the situation mentioned in the user's description meets the exclusion conditions in any clause.
- If there is a match, please indicate the most relevant exclusion clause in the "clause" field (directly quoting the original text).
- If there is no match, it means that no relevant exclusion clauses have been found, and the content of the answer can be empty.
- In the "triggered" field, indicate whether the insurance clause is triggered in this case (output "Triggered Exclusion Clause" or "Not Triggered Exclusion Clause"), and explain why it is triggered or not triggered in the "explanation" field.
- Strictly follow the text content of the exclusion clauses, and do not introduce external knowledge.

【Content of Exclusion Clauses of Insurance Products】
{exclusion_text}

【Output Requirements (JSON format)】
```
{
   "triggered": <str>,
   "clause": <str>,
   "explanation": <str>
}
```

User_prompt:
Query

Figure 7: Example prompt content for the Exclusion Clause Matching Agent, including the System Prompt and a placeholder for the User Prompt (Query).

Coverage Matching Agent Prompt

System_prompt:
You are an insurance claim analysis expert. Your task is to determine whether the situation described by the user falls within the insurance coverage.

【Task Requirements】
- Based on the content of the coverage scope, analyze whether the situation mentioned in the user's description meets the coverage conditions in any clause.
- If there is a match, please indicate the most relevant coverage clause in the "clause" field (directly quoting the original text).
- If there is no match, it means that no relevant coverage clauses have been found, and the content of each field can be empty.
- In the "covered" field, indicate whether the insurance clause is covered in this case (output "Within the coverage" or "Not within the coverage"), and explain why it is covered or not covered in the "explanation" field.
- Strictly follow the text content of the coverage scope, and do not introduce external knowledge.

【Content of Insurance Coverage Scope】
{liability_text}

【Output Requirements (JSON format)】
```
{
   "covered": <str>,
   "clause": <str>,
   "explanation": <str>
}
```

User_prompt:
Query

Figure 8: Example prompt content for the Coverage Matching Agent, including the System Prompt and a placeholder for the User Prompt (Query).

B.5 Coverage Matching Agent

This agent analyzes user-provided insurance claim scenarios, referred to as the Query, to determine if they fall within the insurance policy's coverage. It compares the user's description, the Query, against the coverage clauses extracted from the insurance chunk file, represented as {liability_text}. This agent performs a clause-by-clause analysis to provide a conclusion on whether the claim is covered, along with an explanation. An example of the prompt content for this agent is presented in Figure 8.

B.6 Claims Boundary Analysis Agent

This agent analyzes user-provided insurance claim scenarios, referred to as the Query, to determine if the claim outcome might fall into a "possibly pay, possibly not pay" category. It evaluates the comprehensiveness and rigor of the user's description, the Query, against the full text of the insurance policy, represented as {full_text}. This agent looks for potential special circumstances, missing information, or boundary cases that could lead to an uncertain payment decision. An example of the prompt content for this agent is presented in Figure 9.

B.7 Decision Fusion Agent

This agent serves as the central decision-making unit, integrating the analytical outputs from various sub-agents to arrive at a final insurance claim decision. It synthesizes these results with the complete insurance policy text (represented as {full_text}), generating a conclusive decision ("Pay," "Not Pay," or "Possibly Pay or Possibly Not Pay"), a confidence score, a step-by-step reasoning chain, and the most relevant insurance clauses. This agent also determines if any sub-agents need to be rerun based on the consistency and plausibility of their findings, always prioritizing the original policy wording in case of conflicting or unreasonable sub-agent outputs. The input to this agent includes the user's initial question (Query) and the JSON-formatted results from other agents ({agent_results}).

B.8 Explanation Generation Agent

This agent is responsible for crafting clear and user-friendly explanations for the final insurance claim decision. It synthesizes the ultimate decision from the Decision Fusion Agent with the analytical insights provided by various sub-agents. The agent grounds its explanations in the original text of the relevant insurance clauses, providing a detailed reasoning process. Additionally, it highlights any crucial special notes that the user should be aware of. The input for this agent includes the initial user query {question}, the name of the insurance product {policy_name}, the JSON-formatted outcome from the Decision Fusion Agent {decision_result}, and the JSON-formatted results from other sub-agents {agent_results}. An example of the prompt content for this agent is provided in Figure 11.

Claims Boundary Analysis Agent Prompt

System_prompt:
You are an insurance claim analysis expert. Your task is to analyze the user's description and determine if the conclusion might be "possibly payable, possibly not payable."

【Task Requirements】
- Based on the full text of the insurance policy, analyze whether the user's description is comprehensive, whether the user's keyword expressions are rigorous, whether the user's description fully matches the requirements of the clauses, and whether there are principles of proximate cause, special circumstances, or boundary cases that could lead to a "possibly payable, possibly not payable" conclusion (e.g., a pre-existing health condition not disclosed by the user might be unrelated to the root cause of the claim).
- If special circumstances exist, indicate potential exceptions to the clauses that may affect the payment decision in the "special_cases" field, and explain the key factual elements that the user needs to supplement or more detailed information that would help determine the payment conclusion in the "missing_info" field.
- If it can be accurately determined whether the user's description is payable or not payable, it indicates that the user's description is very comprehensive and no further confirmation is needed; the "special_cases" and "missing_info" fields should be empty.
- Strictly follow the full text of the insurance policy and do not introduce external knowledge.
- Indicate the payment conclusion in the "is_complete" field (output "possibly payable, possibly not payable" or "payable" or "not payable").

【Full Text of Insurance Product】
{full_text}

【Output Requirements (JSON format)】
```
{
   "is_complete": <str>,
   "special_cases": <list[str]>,
   "missing_info": <list[str]>
}
```

User_prompt:
Query

Figure 9: Example prompt content for the Claims Boundary Analysis Agent, including the System Prompt and a placeholder for the User Prompt (Query).

Decision Fusion Agent Prompt

System_prompt:
You are an insurance claim decision fusion agent. Your task is to integrate the analysis results from various sub-agents and, in conjunction with the full text of the insurance policy, generate the final payment decision and reasoning chain. The results from the sub-agents may not always be absolutely correct. If there are unreasonable judgments, the content of the original clauses shall prevail.

【Output Requirements (JSON format)】
```
{
   "decision": <str>,  # "Payable" or "Not Payable" or "Possibly Payable, Possibly Not Payable"
   "confidence": <float>,  # Confidence level (0-1)
   "reasoning_chain": <list[str]>,  # Reasoning steps
   "relevant_clauses": <list[str]>,  # Original text of the most relevant insurance clauses (up to 3)
   "need_rerun": <bool>  # Whether it is necessary to rerun certain agents
}
```

User_prompt:
【Question】
{question}

【Results from Multiple Sub-agents】
{json.dumps(agent_results, indent=2)}

【Full Text of Insurance Product】
{full_text}

Figure 10: Example prompt content for the Decision Fusion/Logic Reasoning Agent, including the System Prompt and the structure of the Input Text (incorporating the Query, sub-agent results, and full policy text)

C Calculation of CLR

We formally define the **Capital Loss Rate (CLR)** as follows. Let N denote the total number of samples in the test set, N_{NP} the number of samples with ground truth label "Not Pay", and N_{PP} the number of samples with ground truth label "Possibly Pay or Possibly Not Pay". Let FP_{NP} represent the false positives where the model incorrectly predicts "Pay" for "Not Pay" cases, and FP_{PP} denote the false positives for "Possibly Pay or Possibly Not Pay" cases. For each misclassified sample i, let $Loss_i$ denote the actual financial loss incurred by the insurer and $Amount_i$ denote the claim amount.

The CLR metric is computed as:

$$\text{CLR} = \frac{\sum_{i=1}^{FP_{\text{NP}}+FP_{\text{PP}}} Loss_i}{\sum_{i=1}^{FP_{\text{NP}}+FP_{\text{PP}}} Amount_i} \times \frac{FP_{\text{NP}} + FP_{\text{PP}}}{N_{\text{NP}} + N_{\text{PP}}} \tag{1}$$

In cases where the loss proportion is uniform across misclassifications, this simplifies to:

$$\text{CLR} = \frac{FP_{\text{NP}} + FP_{\text{PP}}}{N_{\text{NP}} + N_{\text{PP}}} \tag{2}$$

D Ethical Compliance and Data Integrity

This study maintains rigorous ethical standards through multiple safeguards: The dataset originates from the publicly available Aliyun Tianchi Competition: Insurance Clause-Based Q&A (AFAC2024 Challenge Group, 2024) which permits academic use under its open-access license. All personal identifiers were systematically replaced with generic terms ("the policyholder", "the insured") following GDPR-inspired anonymization protocols to eliminate sensitive information. Licensed insurance professionals conducted triple verification of all annotations to ensure both factual accuracy and compliance with Chinese insurance regulations (CIRC Standards 2023). The dataset's design incorporates bias mitigation measures including a multi-difficulty grading system and balanced category distribution to prevent demographic or product-specific biases.

E Data Documentation

The InsClaimQA dataset utilized in this study is derived from a publicly available Chinese fact-based Q&A dataset sourced from the Aliyun Tianchi Competition (AFAC2024 Challenge Group - Competition 2: Insurance Clause-Based Q&A). The dataset comprises meticulously curated insurance claim scenarios spanning five principal domains: Medical Insurance (covering hospitalization, critical illness, and outpatient care), Travel Insurance (including public transportation and international travel coverage), Accident Insurance (encompassing occupational and sports-related injuries), Comprehensive Insurance (family protection plans), and Pension Insurance (savings-type and whole life products). All textual data is exclusively in Simpli-

Explanation Generation Agent Prompt

System_prompt:
You are an insurance claim explanation generation agent. Your task is to generate a user-friendly explanation based on the decision result and the analysis of various sub-agents.

【Output Requirements (JSON format)】
{
"final_decision": <str>, # Final conclusion, content is "Payable" or "Not Payable" or "Possibly Payable, Possibly Not Payable"
"explanation": <str>, # Detailed reasoning explanation based on the original text of the insurance clauses
"relevant_clauses": <list[str]>, # Original text of the most relevant insurance clauses (up to 3)
"special_notes": <list[str]> # Special notes
}

User_prompt:
【Question】
{question}

【Insurance Product】{policy_name}

【Decision Agent Result】
{json.dumps(decision_result, indent=2)}

【Results from Multiple Sub-agents】
{json.dumps(agent_results, indent=2)}

Figure 11: Example prompt content for the Explanation Generation Agent, including the System Prompt and the structure of the Input Text (incorporating the Query, policy name, decision result, and sub-agent results).

fied Chinese, reflecting authentic insurance policy language with specialized legal and medical terminology. The dataset's 1,615 expert-annotated cases maintain rigorous quality standards (98.7% inter-annotator agreement) while preserving privacy through systematic anonymization - personal identifiers in claim descriptions were replaced with generic references (e.g., "the policyholder" or "the insured"). The multi-difficulty grading system (31.1% easy, 35.4% medium, 33.5% difficult cases) ensures balanced representation of both routine claims and edge scenarios requiring complex clause interpretation. Demographic variables are intentionally excluded as insurance claims inherently focus on contractual circumstances rather than author characteristics, aligning with standard practices in actuarial research.

F Recruitment and Payment

The dataset construction involved collaboration with three domain experts recruited through professional insurance industry networks, ensuring participants possessed relevant qualifications in underwriting and claims adjudication. Compensation was determined based on standard consulting rates for insurance professionals in China, with each expert receiving approximately $50 per hour, commensurate with their specialized expertise and the complexity of annotation tasks. All participants provided informed consent prior to engagement, and the payment structure was reviewed by our institutional ethics committee to confirm its adequacy relative to local economic standards and professional norms. No crowdsourced or student annotators were utilized, as the technical nature of insurance clause interpretation required credentialed practitioners.

G Annotator Demographics and Data Provenance

The InsClaimQA dataset was annotated by a team of three domain experts with professional backgrounds in insurance underwriting and legal compliance, all based in mainland China. The annotators consisted of two male and one female professional aged 28-35 years, each holding at least 3 years of experience in claims assessment at major Chinese insurers (Ping An Insurance, CPIC, and China Life). All annotations were conducted in Mandarin following standardized guidelines developed in collaboration with the Insurance Association of China. The original data derives exclusively from the publicly available Aliyun Tianchi Competition corpus (Group, 2024), which contains anonymized insurance product clauses and synthetic claim scenarios compliant with China's Personal Information Protection Law (PIPL). No personally identifiable information (PII) was accessed or included in our processed dataset, and all case descriptions were further sanitized by replacing specific claimant references with generic terms (e.g., "the policyholder").

Data and AI Usage Statement. All data used in this study, including the *InsClaimQA* dataset derived from the Aliyun Tianchi Competition (AFAC2024), were obtained from publicly available sources with proper anonymization to eliminate privacy-sensitive information. No proprietary or restricted data were utilized. AI tools were employed solely for auxiliary purposes: (1) `ChatGPT-4` assisted in refining non-technical prose during manuscript polishing (e.g., grammar checks and fluency improvements), and (2) `GitHub Copilot` accelerated routine code implementation (e.g., JSON parsing scripts). All AI-generated content was rigorously validated against original sources, and core research contributions (dataset construction, methodology, and analysis) remain entirely human-originated.

Category	Examples
Medical Insurance	• Hospitalization Medical Insurance • Accident Medical Insurance • Child Health Insurance • Critical Illness Insurance • Outpatient Insurance • Cancer-Specific Insurance
Travel Insurance	• Public Transportation Insurance • Flight Accident Insurance • Long-Distance Travel Insurance • International Travel Insurance
Accident Insurance	• Sports Accident Insurance • Senior Accident Insurance • Driving Accident Insurance • Family Accident Plan • Work Injury Insurance
Comprehensive Insurance	• Family Comprehensive Protection Plan
Pension Insurance	• Savings-Type Pension Insurance • Increasing Whole Life Insurance

Table 4: Insurance Categories and Examples

Difficulty Level	Key Characteristics	Examples
Simple	• Direct keyword matching • Single clause extraction • No complex reasoning	• "Does the policy cover hospitalization?" (Answer found directly in the policy definition)
Medium	• Inference required • Multi-clause combination • Simple causal/conditional reasoning • Basic insurance concepts	• "If the patient has these symptoms, are they covered for disease X?" (Requires mapping symptoms to disease)
Difficult	• Duty of disclosure/Insurance principles • Exclusion clause complexities • Ambiguous interpretations • Industry/judicial precedents	• "Does the exclusion clause A override the coverage clause B in this specific scenario considering industry practices?"

Table 5: Insurance Claims Question Difficulty Grading Standards

Factor Type	Category	Description
Subjective Factors	Health Status and Medical History	The impact of the insured's physical health status and past medical history on underwriting/claims settlement
	Obligations of the Insurer	The contractual performance responsibilities of the insurer/insured
	Occupational and Behavioral Risks	Additional risks caused by occupation or behavior
Objective Factors	Medical Behaviors and Treatment Methods	The compliance of medical behaviors and the qualifications of medical institutions
	Accidents and External Causes	Whether the nature of the accident falls under accidental injury or an exempted situation
	Claim Conditions and Limits	Thresholds and amount limits for claims settlement
	Special Scenarios and Edge Cases	Determination of unconventional or complex scenarios

Table 6: Insurance Claims Question Category

Category	Examples
Health Status and Medical History	• Diagnosis of a certain disease (such as cancer, diabetes, etc.) • Period of disease diagnosis (before/during/after insurance application) • Concealment of medical history (such as failure to disclose a history of depression) • Deterioration of pre-existing disease after insurance application • Outbreak of hereditary disease (such as congenital heart disease) • Imaging shows a certain symptom (such as a lung nodule detected in a physical examination)
Obligations of the Insurer	• Failure to pay insurance premiums as agreed (premium arrears) • Insufficient application materials (such as lack of a pathological report) • Failure to notify the insurer in a timely manner (such as delayed reporting of a claim)
Occupational and Behavioral Risks	• High-risk occupations (such as firefighters, miners) • Concealment of occupation (such as changing to high-altitude work after insurance application)

Table 7: Subjective Factors Examples (Part 1)

Category	Examples
Medical Behaviors and Treatment Methods	• Treatment in non-approved institutions (such as non-designated hospitals) • Whether hospitalization for a cold is claimable (over-treatment of minor illnesses) • Whether the fees for extra beds during the treatment period are reimbursable • Medical malpractice (such as complications caused by surgical errors) • Abnormal vaccine reactions (such as allergic reactions after vaccination)
Accidents and External Causes	• Post-alcohol accidents (such as injuries from drunk driving) • Injuries suffered during illegal acts (such as injuries from fighting) • High-risk activities (such as injuries from rock climbing) • Whether theft is claimable (related issues in property insurance)
Claim Conditions and Limits	• Medical expenses not reaching the specified amount (such as not exceeding the deductible) • Whether ambulance fees for emergency treatment are claimable • Whether the fees for extra beds during the treatment period are reimbursable
Special Scenarios and Edge Cases	• Diagnosis of a disease after insurance application (waiting-period issues) • Injuries during justifiable defense in illegal acts • Causal relationship between medical malpractice and disease exacerbation

Table 8: Subjective Factors Examples (Part 2)