MMFCTUB: Multi-Modal Financial Credit Table Understanding Benchmark

Anonymous ACL submission

Abstract

The advent of multi-modal language models (MLLMs) has spurred research into their application across various table understanding tasks. However, their performance in credit table understanding (CTU) for financial credit review remains largely unexplored due to the following barriers: low data consistency, high annotation costs stemming from domain-specific knowledge and complex calculations, and evaluation paradigm gaps between benchmark and real-world scenarios. To address these challenges, we introduce MMFCTUB (Multi-Modal Financial Credit Table Understanding Benchmark), a practical benchmark, encompassing more than 7,600 high quality CTU samples across 5 table types. MMFCTUB employ a minimally supervised pipeline that adheres to inter-table constraints and maintains data distributions consistency. The benchmark leverages capacity-driven questions and mask-and-recovery strategy to evaluate models' cross-table structure perception, domain knowledge utilization, and numerical calculation capabilities. Utilizing MMFCTUB, we conduct comprehensive evaluations of both proprietary and open-source MLLMs, revealing their strengths and limitations in CTU tasks. MMFCTUB serves as a valuable resource for the research community, facilitating rigorous evaluation of MLLMs in the domain of CTU.

1 Introduction

Tables have become the predominant format for presenting structured information across various domains due to their superior visual (Zheng et al., 2024; Zhou et al., 2025; Kang et al., 2025) accessibility compared to sequential text (Wu et al., 2025; Wang et al., 2024; Shigarov, 2023; Deng et al., 2022). In financial credit review (Brennecke et al., 2021), loan officers need to understand credit report tables(CRTs) and evaluate applicants' economic profiles, which demands accurate cross-table contents recognition, domain knowledge application and numerical computation of assessment metrics. Multi-modal language models (MLLMs) have recently demonstrated remarkable visual (Yang et al., 2025; Wang et al., 2025b) understanding capabilities and achieved promising results when applied to image processing tasks across multiple domains (AlSaad et al., 2024; Ye et al., 2024). Leveraging large-scale pre-training on diverse domain knowledge and mathematical reasoning datasets, MLLMs have developed robust capabilities in domain knowledge integration and numerical computation. These strengths position MLLMs as a promising approach to credit tables understanding (CTU) tasks. Despite these natural advantages, their capacities have not been comprehensively evaluated in CTU tasks during financial credit review for the following challenges: **1. Limited Availability of Data.**Existing table understanding(TU) benchmarks predominantly rely on generic tables collected from public web corpora. However, this approach is inapplicable to CRTs as stringent privacy regulations. **2. Specialized Table Dependencies and Distributions**. CRTs exhibit domain-specific layouts with strong cross-table dependencies and consistent distributions reflecting individuals' economic profiles. Existing benchmarks fail to capture these characteristics, thereby introducing data bias into evaluation. **3. High Annotation Cost.** Current TU benchmarks predominantly focus on tables with minimal domain knowledge and computational requirements. However, CRTs encode complex economic implications and numerical relations, requiring extensive domain expertise and numerical reasoning during annotation, thus incurring prohibitive costs. **4. Misaligned Evaluation Paradigm.** Existing evaluations predominantly process tables in serialized text format. However, this 1D paradigm is inappropriate for practical credit review, where CTU is performed through 2D visual processing, creating a funda-

mental gap between evaluation and practice.

To address these challenges, we present MMFCTUB (Multi-Modal Financial Credit Table Understanding Benchmark), a comprehensive benchmark for evaluating MLLMs on table structure perception, domain knowledge utilization and numerical reasoning in CTU tasks. MMFCTUB comprises 19,000 credit table images with authentic layouts across 5 categories, incorporating 60 interdependent fields with coherent per-individual distributions. The benchmark provides 7,600 test instances evaluating 54 financial indicators that characterize applicants' economic profiles, sourced from 246 applicants with diverse economic backgrounds.

During data curation, we employ MLLM-assisted programmatic generation to synthesize credit table images, questions, and annotations with minimal human intervention and near-realistic data. To minimize data bias, we reconstruct CRTs using structure prompts derived from authentic templates and populate cells with values generated from applicants' economic profiles. To preserve inter-field dependencies while ensuring generation efficiency, we adopt a three-tier generation process: LLMs generate descriptive features, while rule-based programs compute high-precision numerical data. CRT images are rendered by compiling LaTeX code generated from abstract table definitions. During evaluation, MLLMs' are required to generate answers based on table-related questions based on capacity-driven questions, to fine-grained assess MLLMs' knowledge and calculation capacities, we employee a mask-and-recover strategy and hit rate metrics. Based on MMFCTUB, we conduct a comprehensive evaluation of mainstream MLLMs, including both open-source and proprietary models, as depicted in Table 5. In summary, our contributions are:

- We introduce a MLLM-assisted programmatic generation strategy with low human intervention and high data quality.
- We develop MMFCTUB, a novel benchmark for comprehensive CTU, measuring the performance across table structure perception, domain-knowledge utilization and numerical computing.
- We conduct a comprehensive evaluation of several popular MLLMs, uncovering their strengths and weakness across various dimensions.

2 Related Wrok

2.1 Table Understanding

Recent advances in large language models (LLMs) have spurred significant research in table understanding (Wang et al., 2024; Deng et al., 2022; Chen et al., 2024; Cao and Liu, 2025). TableBench (Wu et al., 2025) evaluates LLM performance across 18 domains and four task types using real-world industrial tables. FewTUD (Liu et al., 2022) introduces the first Chinese benchmark for few-shot table understanding. Sui et al. (Sui et al., 2024) adopt a task decomposition approach, assessing LLM capabilities at each intermediate step (Li et al., 2023, 2025; Zhang et al., 2025; Xu et al., 2025). However, these works primarily focus on general-purpose tables represented in serialized formats such as Markdown, Pandas DataFrames, or HTML.

2.2 Financial Table Understanding

Comparing with understanding general tables for LLM (Lu et al., 2025; Chen et al., 2024; Ren et al., 2025), financial tables are facing with more challenges since they are feature with more complex structures, more domain-specific knowledge (Yang et al., 2024; Guo et al., 2025; Lu et al., 2023) and numerical relationships. FinQA (Chen et al., 2021) extracts tables from financial market reports and provides a dataset for numerical calculation (Su et al., 2024; Loukas et al., 2022; Khang et al., 2024) tasks. (Zhu et al., 2021) TAT-QA addresses the unique requirements of financial tables by defining subtasks including sequence tagging, aggregation operators, and scale prediction to evaluate LLM performance.

2.3 MLLMs for Table Understanding Benchmark

Multimodal large language models (MLLMs) (Yang et al., 2025; Wang et al., 2025b) have been widely applied across domains (Tampubolon, 2025; Chen et al., 2025), prompting domain-specific benchmarks to evaluate their capabilities. Financial tasks require domain knowledge, numerical reasoning, and table structure perception. (Zheng et al., 2024) proposes Table-LLaVA, processing table images directly with strong performance across 23 benchmarks. FinTab-LLaVA (Park et al., 2025) introduces a multimodal LLM tuned on FinTMD for financial table QA, fact verification, and description generation. Credit reports encode eco-

Dataset	Dom	CrT	CrTPN	CrCDep	CrTDep	TaS	RP	AnC	Par	Kno	Comp	Ann
TableBench	Gen	×	×	✓	×	Public	×	High	Text	×	×	19k
FewTUD	Gen	×	×	✓	×	Public	×	Mid	Text	×	×	3k
Entrant	Fin	×	×	✓	×	Public	×	Mid	Text	×	×	331k
FinTab-LLaVA	Fin	×	×	✓	×	Public	×	High	Image	✓	✓	7.3k
MMFCTUB	Fin	✓	3	✓	✓	Pub Str+ Syn Data	✓	Low	Image	✓	✓	7.7k

Table 1: Comparison of datasets for TU. **Dom**: Domain. **CrT**: Cross-Table. **CrTPN**: Cross Table Paradigm Number, **CrCDep**: Cross Columns Dependencies, **CrTDep**: Cross Tables Dependencies, **TaS**: Table Source. **RP**: Allign with Credit Review Process. **AnC**: Annotation Cost. **Par**: Input Paradigm. **Kno**: Domain Knowledge. **Comp**: Computation. **Anno**: Annotations. **Gen**: General. **Fin**: Financial

nomic profiles through specialized layouts and numerical relationships critical for loan decisions, yet existing benchmarks inadequately assess MLLM performance on such tables. We present MMFCTUB, a comprehensive benchmark evaluating MLLMs on financial credit report understanding.

3 MMFCTUB

In this section, we deleve into the meticulous construction of MMFCTUB, introducing our strategic approach to innovative methods employed to generation simulation credit report tables data, design of a comprehensive capability taxonomy, innovative methods leveraged to assess specific capacities, evaluation metrics definition. Additionally, we present detailed statistics of MMFCTUB and contrast it with existing finance table benchmarks, tthereby illustrating its unique features and contributions to the field.

3.1 Benchmark Construction

Credit Table Selection. In practical loan review processes, reviewers must scan multiple relevant tables and extract target data for computation, requiring frequent cross-table understanding. We therefore adopt a cross-table input paradigm for our benchmark. Based on consultation with domain experts, we select 5 commonly used table types: credit transaction tables, residence information tables, occupation information tables, account detail tables, and credit agreement tables.

Credit Table Generation. Existing table generation approaches suffer from two critical limitations: (1)Failing to capture the specific characteristics and requirements of domain-specific scenarios, necessitating substantial efforts in data collection and cleaning, and (2) prohibitively high annotation costs when domain experts are involved in large-scale and complex labeling. To address these dataset limitations, We propose a novel credit table generation methodology that aligns with practical credit review processes while requiring minimal human effort. As illustrated in Figure 1, our approach comprises three key components:

1) Instruction Generation. Credit tables exhibit domain-specific layouts and dependencies defined by experts, represented through table metadata. Our data construction pipeline involves 3 steps. (1) extracting metadata from real-world credit tables using MLLMs, (2) constructing abstract table representations via LLM based on semantic metadata (Figure 4), and (3) generating questions targeting credit review indicators through LLM with expert-designed prompts.

2) Labels Preparation. Credit table computations involve domain-specific economic indicators that require complex multi-column and cross-table aggregations, rendering manual annotation impractical. To mitigate this, we leverage LLMs to generate label calculation function code from abstract table definitions and questions. Ground truth labels are obtained by executing these functions on the generated table data.

3) Data Construction. Credit tables exhibit complex inter-column and cross-table dependencies aligned with applicants' economic profiles. We employ a three-stage generation pipeline. Stage 1: LLMs generate user basic information from diverse economic profiles defined by income and credit score distributions. Stage 2: LLMs produce account detail tables using dependency-aware prompts, followed by rule-based population of numerically constrained fields using real-world financial parameters. Stage 3: Summary and agreement tables are generated from account tables to maintain cross-table dependencies. Tables are rendered as images via LLM-generated LaTeX code for visual QA tasks. This pipeline ensures structural authenticity, semantic coherence, and preservation of complex interdependencies characteristic of real credit data.

Capacity Taxonomy. Drawing inspiration from

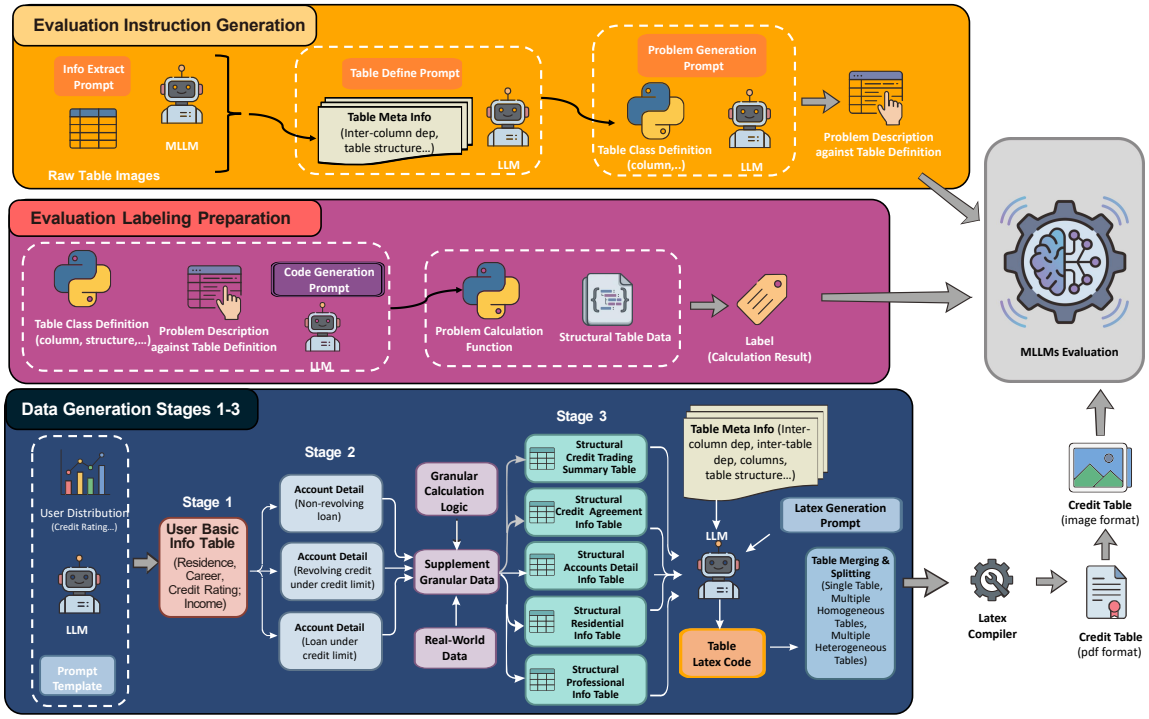

Figure 1: Detail of Finance Credit Table Understanding Dataset Construction.

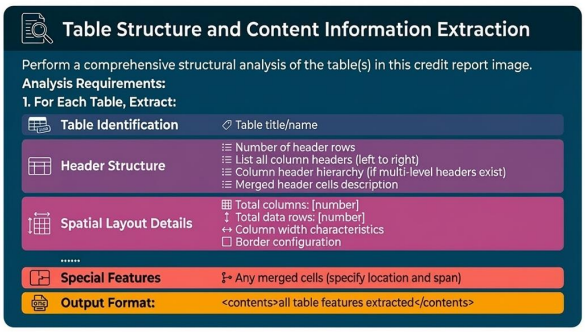

Figure 2: Extracted Table Contents and Meta Info.

the operational workflows of credit reviewers in real-world assessment processes, we propose three task categories to evaluate MLLMs' credit table understanding capabilities: Table Structure Perception (TSP), Domain Knowledge Utilization (DKU), and Numerical Calculation (NC).

TSP assesses models' capacity to perceive table structures, including spatial layouts and cross-table relationships. Evaluation spans two dimensions: structural perception across multiple paradigms (Table 3), and scope perception across three table count ranges (3-5, 6-8, 9-13). By minimizing domain knowledge and computational demands in question design, we isolate structural perception capabilities. Performance is measured by answer accuracy. **DKU**, assesses MLLMs' capacity to apply pre-trained financial knowledge, including terminology, relationships, and formulas. Models must construct formulations and select variables based on entity relationships within perceived table structures. We categorize domain knowledge difficulty into three levels: We define three knowledge difficulty levels. **Knowledge Perception** requires extracting information directly present in tables, such as field names in headers. **Knowledge Analysis** demands deriving information through joint analysis of question semantics and table content, such as computing field-level sums or ratios. **Knowledge Reasoning** requires applying knowledge from prompt context independent of table content, where models must comprehend logical flows and infer intermediate variables for subsequent calculations.

NC, MLLMs' calculation capacity in CTU tasks derives from general arithmetic reasoning and domain-specific numerical relationship understanding. We evaluate three operators frequently used in CTU scenarios: addition (+) for aggregating values across tables (e.g., summing column values from different account tables), subtraction (-) for computing duration metrics (e.g., days between

account opening and closure), and division (÷) for calculating proportional metrics (e.g., credit utilization ratios measuring account borrowing saturation).

Evaluation Paradigm. We establish distinct evaluation paradigms for three CTU capacities, using differentiated questions to isolate specific capability bottlenecks. For **structure perception**, questions require extensive table extraction with minimal computation, measuring accuracy on unmasked samples to isolate structural understanding independent of other capabilities.

For **domain knowledge utilization**, we employ mask-and-recovery evaluation where column names are randomly masked with 'XXXX' tokens. We use Financial Knowledge Hit Rate (FKHR) to measure proficiency in identifying and applying domain-specific knowledge.

$$\text{FKHR}_{ij} = \frac{|P_{ij} \cap K_{ij}|}{|K_{ij}|}, \tag{1}$$

where P_{ij} and K_{ij} denote the predicted and label knowledge groups for question i and group j. The term $|P_{ij} \cap K_{ij}|$ counts correctly identified elements, while $|K_{ij}|$ is the ground truth size. The metric equals 1 when all elements are correctly predicted.

$$\overline{\text{FKHR}}_j = \frac{1}{N} \sum_{i=1}^{N} \text{FKHR}_{ij}, \tag{2}$$

where N denotes the total number of questions in the dataset, and FKHR_{ij} represents the hit rate for the i-th question on the j-th knowledge group. This metric reflects the model's overall prediction performance on a specific knowledge group. For numerical calculation, we assess MLLMs through the mask-and-recovery paradigm. During question generation, the program randomly masks calculation operators from an expert-predefined list in the prompts using 'YYYY' as the mask token. During evaluation, we employ calculation operator hit rate (COHR) as the metric, quantifying MLLMs' proficiency in identifying and applying appropriate operators essential to the calculation process:

$$\text{COHR}_{ij} = \frac{1}{|O_{ij}|} \sum_{k=1}^{|O_{ij}|} \mathbb{1}(O_{ijk} = P_{ijk}), \tag{3}$$

where $O_{ij} = \{O_{ij1}, O_{ij2}, \ldots, O_{ijn}\}$ denotes the ground truth operator sequence for the j-th operator group in the i-th question, and $P_{ij} = \{P_{ij1}, P_{ij2}, \ldots, P_{ijm}\}$ represents the corresponding predicted sequence. The indicator function $\mathbb{1}(\cdot)$ returns 1 if the condition is satisfied and 0 otherwise. $|O_{ij}|$ denotes the length of the ground truth sequence.

Unlike the knowledge hit rate which uses set intersection, COHR enforces strict positional matching: an operator at position k is considered correct only if $O_{ijk} = P_{ijk}$. When the predicted sequence is shorter than the ground truth ($k > |P_{ij}|$), the missing positions are counted as mismatches.

$$\overline{\text{COHR}}_j = \frac{1}{N} \sum_{i=1}^{N} \text{COHR}_{ij}, \tag{4}$$

where N denotes the total number of questions in the dataset. This metric reflects the model's overall performance on a specific operator group.

3.2 Dataset Statistics

MMFCTUB contains 246 users characterized by credit ratings and monthly income, with equal distributions across three rating ranges and realistic income distributions (Figure 5). The dataset comprises 19k tables across multiple types, where table numbers and categories are determined by user economic profiles. All distributions are shown in Figure 5.

The benchmark comprises 18k training and 7.6k test QA pairs across three formats (calculation, single-choice, multiple-choice), each targeting specific MLLM capabilities. To evaluate knowledge utilization and numerical reasoning, we randomly mask predefined knowledge elements and computational operators. Evaluation covers five credit assessment knowledge categories (temporal, amount, account, weight, ratio), each requiring distinct table content scopes and domain reasoning. Knowledge and operator distributions are presented in Figure 5.

4 Experiment

Utilizing MMFCTUB, we conduct a comprehensive evaluation of diverse MLLMs, encompassing both open-source and proprietary systems. During evaluation, we employ the default hyperparameters specified in their respective official implementations for inference. We leverage LaTeX code generation coupled with compilation tools to convert structured tables into image format, with each table rendered as an independent image file for visual processing by the models.

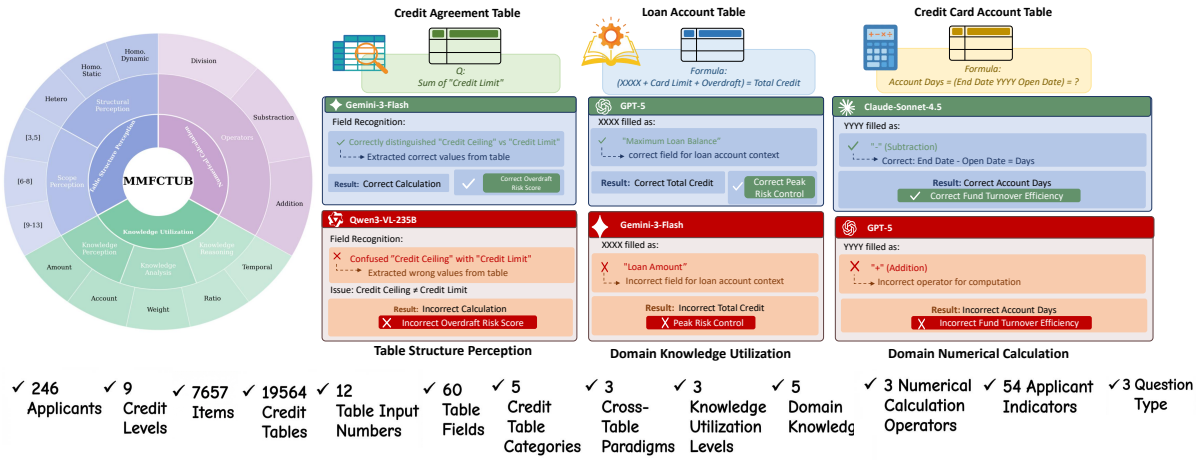

Figure 3: The Comprehensive Taxonomy, Data Examples and Statistical Characteristics of MMFCTUB. The circular taxonomy diagram shows core cognitive levels, knowledge categories and operators.

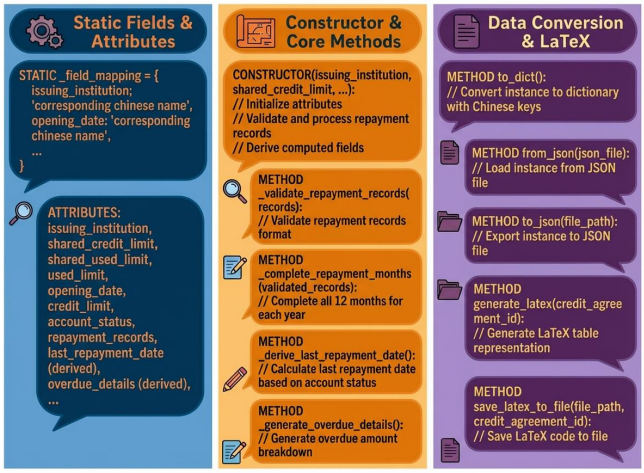

Figure 4: Detail of Definition of Credit Card.

4.1 Main Results

Open-Source MLLMs. We evaluate several prominent open-source MLLMs that support multi-image inference: Qwen-VL-2.5 (Bai et al., 2025), Qwen-VL-3 (Yang et al., 2025), Intern-VL-3 (Wang et al., 2025b), and Intern-VL-3.5 (Wang et al., 2025b). Notably, Qwen3-VL-235B-think achieves competitive performance, surpassing GPT-4o (Islam and Moushi, 2025) by 24% in accuracy, particularly excelling in domain knowledge utilization and numerical calculation. Interestingly, within the same model family, smaller models demonstrate superior performance on complex table understanding tasks. Among 8B-scale open-source models, XuanYuan4.0-VL achieves the best performance across all metrics and even attains the highest score globally on simple cross-table perception. We attribute this to XuanYuan's native financial domain pre-training, which enables better understanding of financial table structures and content dependencies.

Proprietary MLLMs. We evaluate four proprietary MLLMs: Gemini-3-Flash (Comanici et al., 2025), Claude Sonnet 4.5 (Salbas and Buyuktoka, 2025), GPT-5 (Wang et al., 2025a), and GPT-4o. Gemini-3-Flash leads with 15% higher accuracy than the second-best model, excelling in structure recognition and scope perception. GPT-5 shows superior knowledge utilization across all capability levels, while Sonnet 4.5 achieves best numerical computation performance. GPT-4o underperforms at 31% accuracy. We attribute these results to the distinct capability profiles of each model. Notably, GPT-5 achieves second-best overall performance despite weaker numerical calculation than Sonnet 4.5 and Gemini-3-Flash, **suggesting credit table understanding depends more on visual perception and knowledge utilization than numerical calculation, which provides important guidance for further model optimization.**

Table Visual Perception. Results demonstrate that increasing table quantity degrades performance for most MLLMs. Interestingly, however, Gemini-3-Flash achieves its best performance in the [9,13] group. We attribute this phenomenon to the model's superior visual encoding capabilities: additional tables provide more visual evidence for reasoning rather than imposing computational burden, thereby enhancing inference efficiency and ultimately improving understanding performance. **This suggests that given sufficient visual capacity, increasing the number of in-**

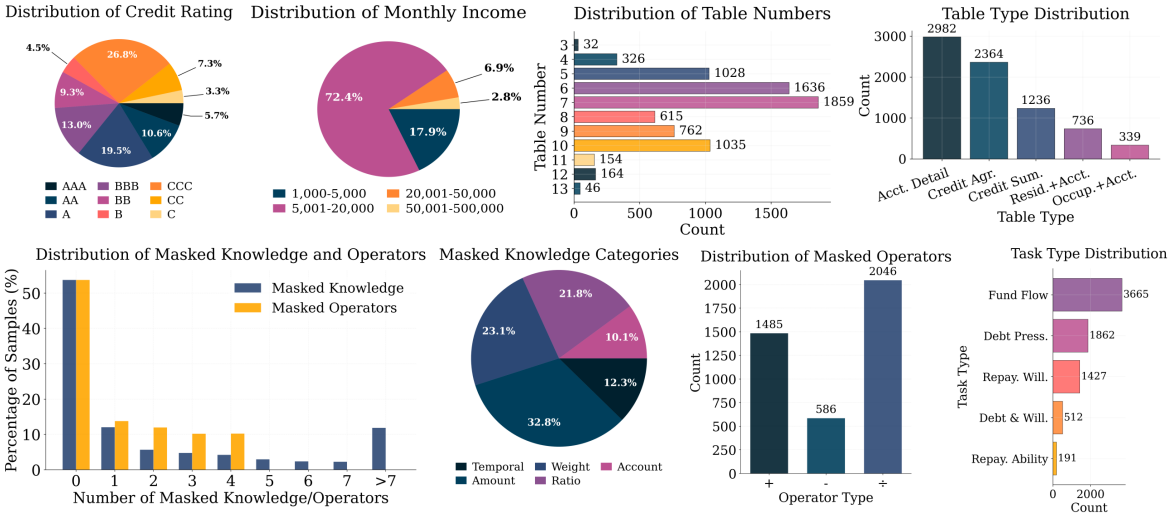

Figure 5: Dataset Details of MMFCTUB.

MLLMs	Structure Perception			Scope Perception			Knowledge Utilization			Numerical Calculation			Final
	Homo. Static	Homo. Dynamic	Hetero.	[3, 5]	[6, 8]	[9, 14]	Know. Percep.	Know. Analy.	Know. Reason.	Add.	Sub.	Div.	Acc.
Proprietary Models													
Gemini-3-Flash	81.70	76.15	63.92	76.81	76.63	80.69	41.47	34.81	24.87	43.83	24.12	52.68	78.18
Sonnet4.5	59.76	50.84	33.39	65.19	59.68	53.01	31.96	31.31	30.33	44.91	18.67	50.88	52.96
Sonnet4.5-think	69.68	56.70	41.26	66.34	59.89	58.67	37.27	34.26	31.45	44.88	18.02	54.95	56.86
GPT-5	77.83	58.41	31.50	68.38	63.82	64.33	52.84	36.71	27.98	22.47	2.92	22.06	63.68
GPT-4o	27.77	27.82	31.59	31.16	28.07	25.44	14.32	9.53	9.49	16.31	16.31	13.48	31.01
Open-source Models													
GLM-4_6V	38.81	52.99	37.10	45.30	43.29	40.94	8.23	8.39	5.28	9.31	4.21	13.37	43.93
GLM-4_1V	45.04	50.65	35.86	57.48	50.59	43.81	12.07	7.39	5.78	6.08	3.81	9.87	45.75
Keye-VL-1.5-8B	44.20	39.66	25.18	52.44	49.28	39.62	4.75	4.19	4.13	6.97	2.12	10.39	40.11
Keye-VL-8B	45.24	40.98	33.85	49.53	50.11	41.30	7.74	8.31	8.34	12.11	2.5	10.77	42.09
MiniCPM-V-4_5	39.65	30.50	26.10	44.28	38.64	29.49	11.64	2.97	4.62	4.56	1.46	5.37	34.01
InternVL3_5-8B	49.37	46.45	36.59	51.66	48.59	43.39	10.04	5.02	3.66	4.66	2.54	5.98	46.37
InternVL3_5-38B	49.83	45.74	35.12	51.83	47.55	44.68	10.83	5.13	4.10	4.71	3.25	6.18	47.39
InternVL3_5-241B-A28B	50.14	55.21	38.57	52.55	48.61	45.69	12.56	5.21	4.85	4.61	3.82	6.55	47.55
Qwen3-VL-8B	37.39	53.18	32.30	36.82	38.50	39.40	15.31	5.41	6.64	14.19	4.94	5.88	42.63
Qwen3-VL-8B-think	46.73	40.30	31.08	42.91	44.55	36.22	10.30	10.8	7.70	14.98	4.57	6.36	43.57
Qwen3-VL-30B-think	54.46	42.55	36.37	44.15	45.09	37.11	11.34	11.05	7.97	14.56	5.63	7.15	45.92
Qwen3-VL-235B-think	59.52	65.00	37.11	60.67	61.32	58.40	25.47	24.48	22.89	24.95	15.10	29.15	55.49
Qwen2.5-VL-72B-Instruct	28.38	24.26	26.14	24.48	23.41	29.69	21.53	5.23	9.99	10.23	5.16	6.25	26.69
XuanYuan4.0-VL-8B	61.74	55.76	31.22	65.96	61.41	49.93	19.41	6.51	3.76	14.90	5.71	11.39	53.07

Table 2: **Performance Comparison of MLLMs Across Different Evaluation Dimensions.** Under Structure Perception: Homo. Static = Cross-table Operations Between Homogeneous Static Tables; Homo. Dynamic = Cross-table Operations Between Homogeneous Dynamic Tables; Hetero.= Cross-table Operations Between Heterogeneous Tables. Under Knowledge Utilization: Know. Percep. = Knowledge Perception; Know. Analy. = Knowledge Analysis; Know. Reason. = Knowledge Reasoning. Under Numerical Calculation: Add. = Addition; Sub. = Subtraction; Div. = Division. Deep red indicates the highest value in each column, light red indicates the second highest.

put tables can actually benefit understanding by providing richer contextual information for reasoning.

Knowledge Utilization. As shown in Figure 2, we observe that almost all MLLMs achieve the highest performance on Know.Percep and the lowest performance on Know.Reason. We attribute this phenomenon to the fact that directly retriev-

MLLMs	Table Type					Domain Knowledge					Task Type					Final Acc
	Credit Trans.	Residence + Acct.	Credit Agreement	Occupation + Acct.	Account Details	Temporal	Amount	Account	Weight	Ratio	Fund Flow	Debt Pressure	Repay. Willing.	Debt & Willing.	Repay. Ability	
Proprietary Models																
Gemini-3-Flash	58.90	56.06	81.70	80.72	89.47	13.76	45.03	37.50	30.64	15.96	77.58	74.87	87.73	63.64	88.54	78.18
Sonnet4.5	47.52	35.00	65.89	29.94	50.84	11.80	35.95	30.74	22.46	36.33	57.26	48.23	49.50	55.14	32.35	52.96
Sonnet4.5-think	51.34	40.28	69.68	43.40	62.62	37.50	41.97	29.07	37.37	37.37	59.53	52.22	38.65	55.41	38.65	56.86
GPT-5	39.63	24.83	77.83	45.89	73.10	20.47	20.47	32.05	18.88	38.64	61.58	61.58	66.49	44.19	76.47	63.68
GPT-4o	20.15	28.93	27.77	37.34	33.78	3.79	13.16	7.12	10.33	6.18	24.78	34.96	43.55	26.45	31.62	31.01
Open-source Models																
GLM-4_6V	34.58	37.70	38.81	35.96	52.99	2.05	1.05	9.79	7.44	3.50	41.97	45.84	47.97	39.81	43.88	43.93
GLM-4_1V	31.31	38.55	45.04	30.70	50.65	2.95	11.33	9.13	7.31	3.17	45.13	49.70	47.33	34 .95	35.20	45.75
Keye-VL-1.5-8B	30.43	23.83	44.20	27.70	39.66	1.11	5.08	5.41	5.05	2.01	38.67	45.44	39.78	35.39	32.26	40.11
Keye-VL-8B	37.71	32.55	45.24	36.41	40.98	3.43	9.56	7.46	9.20	4.82	41.33	44.88	40.74	45.72	27.91	42.09
MiniCPM-V-4_5	27.18	27.83	39.65	27.83	30.05	2.42	6.02	4.69	5.82	2.27	33.35	36.95	33.94	30.17	27.46	34.01
InternVL3_5-8B	34.81	35.99	49.37	37.75	46.45	2.64	8.13	5.39	5.14	1.97	43.21	50.46	50.24	43.55	45.92	46.37
InternVL3_5-38B	35.16	36.74	49.83	37.16	46.37	2.75	8.55	5.53	4.06	2.16	44.51	50.95	51.32	44.15	46.31	47.39
InternVL3_5-241B-A28B	36.25	35.74	50.14	38.15	47.12	4.85	14.71	6.74	9.12	9.27	43.56	48.53	52.34	45.11	47.10	47.55
Qwen3-VL-8B	30.97	31.31	37.39	34.19	53.18	3.19	11.64	1.92	8.26	2.23	38.26	49.17	46.44	39.77	41.80	42.63
Qwen3-VL-8B-think	30.97	31.31	46.73	34.19	53.18	3.23	11.89	2.13	8.51	2.53	39.62	44.81	44.60	35.70	52.55	43.57
Qwen3-VL-30B-think	31.70	29.82	54.46	33.49	40.30	1.93	13.08	3.29	6.98	2.63	39.54	43.98	40.15	29.49	32.98	45.92
Qwen3-VL-235B-think	36.08	30.36	59.52	39.71	42.72	5.14	15.74	6.82	8.82	9.48	53.23	60.89	58.29	52.44	32.89	55.40
Qwen2.5-VL-72B-Instruct	14.16	22.42	28.38	34.22	31.72	3.15	10.37	5.17	7.26	4.68	21.69	34.75	32.45	16.21	29.32	26.69
XuanYuan4.0-VL-8B	50.66	30.17	61.74	33.25	55.76	4.73	12.36	5.63	4.69	2.91	52.89	54.58	52.90	53.46	39.80	53.07

Table 3: **Performance Comparison of MLLMs Across Different Table Types, Domain Knowledge, and Task Types.** Under Table Type: Credit Trans. = Credit Transaction; Residence + Acct. = Residence and Account; Occupation + Acct. = Occupation and Account. Under Task Type: Repay. Willing. = Repayment Willingness; Debt & Willing. = Debt and Willingness; Repay. Ability = Repayment Ability. Deep red indicates the highest value in each column, light red indicates the second highest.

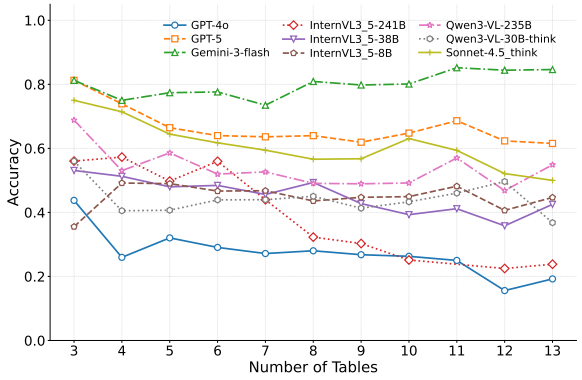

Figure 6: Comparison of Credit Tables Understanding with Number of Tables.

ing required knowledge from tables through visual perception is considerably easier for MLLMs than inferring knowledge through contextual reasoning. This indicates that knowledge analysis and reasoning capabilities constitute the primary bottleneck for MLLMs in table understanding tasks. Additionally, for domain knowledge category, as demonstrated in Figure 3,we observe that different models exhibit varying sensitivity to different knowledge categories. Notably, Gemini-3-Flash shows lower sensitivity to temporal and ratio knowledge despite achieving the highest overall accuracy. **This suggests that knowledge capability optimization should adopt differentiated strategies tailored to each model's knowledge sensitivity profile.**

5 Conclusion

This work introduces MMFCTUB, a novel finance credit table understanding benchmark specifically designed to evaluate the capacities of MLLMs in credit table understanding. MMFCTUB encompass diverse practical credit tables and fine-grained capacities. Extensive evaluations on MMFCTUB allow us to identify significant performance limitations among existing MLLMs in table structure perception, domain knowledge utilization and numerical computation.

6 Limitations

Our evaluation focuses on the most frequently used table types in credit review processes. However, credit reports contain additional information that reflects applicants' economic profiles, and metrics computed solely from our selected tables may introduce bias in comprehensive credit assessment. Furthermore, while MMFCTUB incorporates diverse table types and quantities to measure MLLMs' table structure perception capabilities, finer-grained quantification would provide more interpretable insights. Specifically, operation-level granularity such as cross-row retrieval would offer more actionable guidance for improving MLLMs' true capacities in table understanding.

References

Rawan AlSaad, Alaa Abd-Alrazaq, Sabri Boughorbel, Arfan Ahmed, Max-Antoine Renault, Rafat Damseh, and Javaid Sheikh. 2024. Multimodal large language models in health care: applications, challenges, and future outlook. *Journal of medical Internet research*, 26:e59505.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, and 1 others. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Claire Brennecke, Gohar Siravyan, and B Heath Witzen. 2021. Commercial credit on consumer credit reports. *Consumer Financial Protection Bureau Office of Research Reports Series*, (21-7).

Lang Cao and Hanbing Liu. 2025. Tablemaster: A recipe to advance table understanding with language models. *arXiv preprint arXiv:2501.19378*.

Si-An Chen, Lesly Miculicich, Julian Eisenschlos, Zifeng Wang, Zilong Wang, Yanfei Chen, Yasuhisa Fujii, Hsuan-Tien Lin, Chen-Yu Lee, and Tomas Pfister. 2024. Tablerag: Million-token table understanding with language models. *Advances in Neural Information Processing Systems*, 37:74899–74921.

Yanlin Chen, Chenjia Huang, Shumiao Gao, Yifan Lyu, Xinyuan Chen, Shen Liu, Dat Bao, and Chunli Lv. 2025. A multimodal deep learning approach for legal english learning in intelligent educational systems. *Sensors*, 25(11):3397.

Zhiyu Chen, Wenhu Chen, Charese Smiley, Sameena Shah, Iana Borova, Dylan Langdon, Reema Moussa, Matt Beane, Ting-Hao Huang, Bryan R Routledge, and 1 others. 2021. Finqa: A dataset of numerical reasoning over financial data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3697–3711.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Xiang Deng, Huan Sun, Alyssa Lees, You Wu, and Cong Yu. 2022. Turl: Table understanding through representation learning. *ACM SIGMOD Record*, 51(1):33–40.

Xin Guo, Haotian Xia, Zhaowei Liu, Hanyang Cao, Zhi Yang, Zhiqiang Liu, Sizhe Wang, Jinyi Niu, Chuqi Wang, Yanhui Wang, and 1 others. 2025. Fineval: A chinese financial domain knowledge evaluation benchmark for large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6258–6292.

Raisa Islam and Owana Marzia Moushi. 2025. Gpt-4o: The cutting-edge advancement in multimodal llm. In *Intelligent Computing-Proceedings of the Computing Conference*, pages 47–60. Springer.

Xiaoqiang Kang, Shengen Wu, Zimu Wang, Yilin Liu, Xiaobo Jin, Kaizhu Huang, Wei Wang, Yutao Yue, Xiaowei Huang, and Qiufeng Wang. 2025. Can grpo boost complex multimodal table understanding? In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 12642–12655.

Alex Khang, Rashmi Gujrati, Hayri Uygun, RK Tailor, and Sanjaya Gaur. 2024. *Data-driven modelling and predictive analytics in business and finance: concepts, designs, technologies, and applications*. CRC Press.

Xingzuo Li, Kehai Chen, Yunfei Long, Xuefeng Bai, Yong Xu, and Min Zhang. 2025. Generator-assistant stepwise rollback framework for large language model agent. *arXiv preprint arXiv:2503.02519*.

Yifei Li, Zeqi Lin, Shizhuo Zhang, Qiang Fu, Bei Chen, Jian-Guang Lou, and Weizhu Chen. 2023. Making language models better reasoners with step-aware verifier. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5315–5333.

Ruixue Liu, Shaozu Yuan, Aijun Dai, Lei Shen, Tiangang Zhu, Meng Chen, and Xiaodong He. 2022. Few-shot table understanding: A benchmark dataset and pre-training baseline. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3741–3752.

Lefteris Loukas, Manos Fergadiotis, Ilias Chalkidis, Eirini Spyropoulou, Prodromos Malakasiotis, Ion Androutsopoulos, and Georgios Paliouras. 2022. Finer: Financial numeric entity recognition for xbrl tagging. *arXiv preprint arXiv:2203.06482*.

Dakuan Lu, Hengkui Wu, Jiaqing Liang, Yipei Xu, Qianyu He, Yipeng Geng, Mengkun Han, Yingsi Xin, and Yanghua Xiao. 2023. Bbt-fin: Comprehensive construction of chinese financial domain pre-trained language model, corpus and benchmark. *arXiv preprint arXiv:2302.09432*.

Weizheng Lu, Jing Zhang, Ju Fan, Zihao Fu, Yueguo Chen, and Xiaoyong Du. 2025. Large language model for table processing: A survey. *Frontiers of Computer Science*, 19(2):192350.

Hyejin Park, Jiyoon Lee, and Hayoung Oh. 2025. Fintab-llava: Finance domain-specific table understanding multimodal llm using fintmd. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 235–246. Springer.

Yi Ren, Chenglong Yu, Weibin Li, Wei Li, Zixuan Zhu, TianYi Zhang, ChenHao Qin, WenBo Ji, and Jianjun Zhang. 2025. Tablegpt: a novel table understanding

method based on table recognition and large language model collaborative enhancement. *Applied Intelligence*, 55(5):311.

Ali Salbas and Rasit Eren Buyuktoka. 2025. Performance of large language models in recognizing brain mri sequences: a comparative analysis of chatgpt-4o, claude 4 opus, and gemini 2.5 pro. *Diagnostics*, 15(15):1919.

Alexey Shigarov. 2023. Table understanding: Problem overview. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 13(1):e1482.

Huan-Yi Su, Ke Wu, Yu-Hao Huang, and Wu-Jun Li. 2024. Numllm: Numeric-sensitive large language model for chinese finance. *arXiv preprint arXiv:2405.00566*.

Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. 2024. Table meets llm: Can large language models understand structured table data? a benchmark and empirical study. In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining*, pages 645–654.

Manotar Tampubolon. 2025. Challenges in using multimodal argumentation in legal code. *International Journal for the Semiotics of Law-Revue internationale de Sémiotique juridique*, pages 1–18.

Shansong Wang, Mingzhe Hu, Qiang Li, Mojtaba Safari, and Xiaofeng Yang. 2025a. Capabilities of gpt-5 on multimodal medical reasoning. *arXiv preprint arXiv:2508.08224*.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, and 1 others. 2025b. Internvl3. 5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*.

Zilong Wang, Hao Zhang, Chun-Liang Li, Julian Martin Eisenschlos, Vincent Perot, Zifeng Wang, Lesly Miculicich, Yasuhisa Fujii, Jingbo Shang, Chen-Yu Lee, and 1 others. 2024. Chain-of-table: Evolving tables in the reasoning chain for table understanding. *arXiv preprint arXiv:2401.04398*.

Xianjie Wu, Jian Yang, Linzheng Chai, Ge Zhang, Jiaheng Liu, Xeron Du, Di Liang, Daixin Shu, Xianfu Cheng, Tianzhen Sun, and 1 others. 2025. Tablebench: A comprehensive and complex benchmark for table question answering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 25497–25506.

Guowei Xu, Peng Jin, Ziang Wu, Hao Li, Yibing Song, Lichao Sun, and Li Yuan. 2025. Llava-cot: Let vision language models reason step-by-step. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2087–2098.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Cehao Yang, Chengjin Xu, and Yiyan Qi. 2024. Financial knowledge large language model. *arXiv preprint arXiv:2407.00365*.

Qinghao Ye, Haiyang Xu, Jiabo Ye, Ming Yan, Anwen Hu, Haowei Liu, Qi Qian, Ji Zhang, and Fei Huang. 2024. mplug-owl2: Revolutionizing multimodal large language model with modality collaboration. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 13040–13051.

Jingyi Zhang, Jiaxing Huang, Huanjin Yao, Shunyu Liu, Xikun Zhang, Shijian Lu, and Dacheng Tao. 2025. R1-vl: Learning to reason with multimodal large language models via step-wise group relative policy optimization. *arXiv preprint arXiv:2503.12937*.

Mingyu Zheng, Xinwei Feng, Qingyi Si, Qiaoqiao She, Zheng Lin, Wenbin Jiang, and Weiping Wang. 2024. Multimodal table understanding. *arXiv preprint arXiv:2406.08100*.

Bangbang Zhou, Zuan Gao, Zixiao Wang, Boqiang Zhang, Yuxin Wang, Zhineng Chen, and Hongtao Xie. 2025. Syntab-llava: Enhancing multimodal table understanding with decoupled synthesis. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24796–24806.

Fengbin Zhu, Wenqiang Lei, Youcheng Huang, Chao Wang, Shuo Zhang, Jiancheng Lv, Fuli Feng, and Tat-Seng Chua. 2021. Tat-qa: A question answering benchmark on a hybrid of tabular and textual content in finance. *arXiv preprint arXiv:2105.07624*.

A Dataset Construct Prompts

We introduced the dataset construction process in Section 3.1, which involves both advanced LLMs (GPT-5) and MLLMs (Gemini 2.5-Flash). To illustrate how these models interact with each pipeline component, we detail the prompts used in each generation stage. Given the overall user economic background distributions (average monthly income, credit score), table infrastructure specifications, and inter-table constraints, we employ GPT-5 to generate detailed loan accounts based on this information. As shown in Tables 4,5, and 6, credit reports contain three account types: Quasi-Credit Card, General Loan Account, and Credit Card Account. Each account type includes common attributes (e.g., opening date, repayment records) as well as type-specific fields (e.g., overdraft balance for Credit Card Accounts). We therefore design specialized prompts for each account type. The LLM leverages its domain knowledge to understand the user's economic profile while adhering to account generation rules, producing account details that align with the user's background. All account fields are defined based on real-world credit report schemas. Notably, General Loan Accounts are further subdivided into three subtypes: Non-revolving Loan Account, Sub-account under Revolving Credit Limit, and Revolving Loan Account. Since these subtypes follow different table construction rules, we design subtype-specific prompts and generation procedures to ensure compliance with their respective structural constraints.

B Evaluation Prompts

C Expert Consultation Process Record

The design and validation of our research framework benefited significantly from consultations with senior professionals at one of the largest financial institutions, which serves over 2 billion users globally. Through structured interviews with industry specialists, we gathered critical insights into consumer credit evaluation. These expert perspectives directly shaped the development of this benchmark. Our consultation panel comprised seasoned specialists across credit underwriting, risk control, and lending operations, each with 8 to over 15 years of hands-on experience. Each expert brings unique domain knowledge spanning different aspects of the credit evaluation lifecycle, from initial application review to final credit decision-making.

Key findings from our expert consultations highlighted several critical aspects:

Expert 1 (14 years, Senior Credit Underwriter) Specializes in consumer credit assessment and loan approval decision-making. Extensive experience in interpreting multi-table credit data including credit reports, bank statements, and asset verification documents.

Expert 2 (12 years, Credit Risk Manager) Focuses on credit risk modeling and portfolio risk management. Expert in analyzing credit bureau reports, identifying default indicators across multiple data tables, and developing risk assessment frameworks.

Expert 3 (11 years, Credit Review Specialist) Concentrates on credit application review and financial document verification. Proficient in cross-referencing information across credit tables, detecting inconsistencies, and validating borrower credentials.

Expert 4 (9 years, Lending Operations Manager) Manages end-to-end credit approval workflows and automation systems. Deep understanding of credit table processing requirements, data integration challenges, and operational efficiency optimization.

Expert 5 (10 years, Credit Policy Analyst) Designs and implements credit underwriting policies and scoring models. Expertise in credit scoring methodology, regulatory compliance (Basel III, local lending regulations), and credit table standardization.

Expert 6 (8 years, Credit Data Analyst) Specializes in quantitative credit analysis and tabular data processing. Expert in extracting risk signals from structured credit data, performing multi-table joint analysis, and building predictive models for creditworthiness assessment.

The expert panel provided invaluable guidance on three critical aspects. First, the practical challenges in processing and integrating multi-table credit data, including credit reports and asset statements etc.; Second, the information dependencies and cross-validation requirements across different credit tables that underwriters must consider. Lastly, the real-world task requirements for credit assessment, including question answering about borrower profiles, fact verification across multiple data sources, and comprehensive creditworthiness summarization. Their domain expertise ensured that our benchmark addresses authentic industry

Table 4: Prompt for Quasi-Credit Card Account Data Generation

Section	Content
Role Definition	You are an experienced credit review expert with extensive experience in credit application approval.
Task Objective	Generate the user's loan information based on the following requirements according to the user's credit status, economic situation, residential situation, and employment situation.
Input Parameters	• User's average monthly income: {average_month_income} • User's credit rating: {credit_rating} • Residential situation: {live_tab} • Employment situation: {prof_tab}
Output Requirement	Generate this user's semi-credit card usage situation
Required Fields	Issuing institution, Opening date, Account credit limit, Shared credit limit, Shared used limit, Currency, Business type, Guarantee method, Account status, Due date, Actual repayment this month, Cutoff date, Overdraft balance, Average overdraft balance in recent 6 months, Maximum overdraft balance, Unpaid balance overdue for more than 180 days, Total months of bad debt, Total months of overdue, Total months of settled, Repayment record start month, Repayment record end month, Account type, Total overdraft periods in repayment record, Current overdraft periods in repayment record
Account Status Options	Normal, Overdue, Bad Debt, Settled
Constraint Rule	(total months of bad debt + total months of overdue + total months of settled + n) $\leq$ m Where: • m = total months from repayment record start month to repayment record end month (inclusive) • n = random integer between 5-12
Currency Assignment	Value should be assigned based on user attributes
Field Format Examples	**Date Fields:** Opening date: "2020.05.15" (YYYY.MM.DD); Cutoff date: "2024.09.30"; Due date: "2024.10.25"; Repayment record start month: "2023-01" (YYYY-MM); Repayment record end month: "2023-12". **Institutional Fields:** Issuing institution: "Issuing Institution GQ"; Business type: "Personal Semi-Credit Card"; Guarantee method: "Credit Guarantee"; Account status: "Normal"; Account type: "Semi-Credit Card" (fixed); Currency: "RMB". **Credit Limits:** Account credit limit: 50000; Shared credit limit: 50000; Shared used limit: 30000 ($\leq$ shared credit limit). **Balances:** Overdraft balance: 4500 ($<$ account credit limit); Average overdraft balance (6M): 3800; Maximum overdraft balance: 6200; Actual repayment this month: 3200; Unpaid balance overdue $>$180 days: 0. **Period Counts:** Total months of bad debt: 2; Total months of overdue: 3; Total months of settled: 2; Total overdraft periods: 7; Current overdraft periods: 5.
Output Format	The account situation should be output in JSON object format. The overall output is a JSON object array, where each JSON object corresponds to one account. The keys within the object are the required fields for the account, and the values need to be reasonably assigned by you based on the user's specific situation. Note: Do not output any thinking process or extra content. Output strictly according to the specified format.

Table 5: Prompt For General Loan Account Data Generation

Section	Content
Role Definition	You are an experienced credit review expert with extensive experience in credit application approval.
Task Objective	You need to generate the user's loan information based on the following requirements according to the user's credit status, economic situation, residential situation, and employment situation.
Input Parameters	• User's average monthly income: {average_month_income} • User's credit rating: {credit_rating} • Residential situation: {live_tab} • Employment situation: {prof_tab}
Output Requirement	Please generate this user's loan situation, including non-revolving loan accounts, sub-accounts under revolving credit limit, and revolving loan accounts
Required Fields	Managing institution, Account credit limit, Currency, Business type, Guarantee method, Account status, Repayment periods, Remaining repayment periods, Actual repayment this month, Repayment method, Opening date, Closing date, Repayment record start month, Repayment record end month, Current overdue periods, Repayment due this month, Loan amount, Account type, Total months of bad debt, Total months of overdue, Total months of settled
Account Status Options	Normal, Overdue, Bad Debt, Settled
Business Type Options	Personal Commercial Housing Loan, Personal Housing Provident Fund Loan, Commercial Student Loan, Personal Consumer Loan, Cash Loan
Repayment Method Options	Installment Equal Principal, Installment Equal Principal and Interest, One-time Principal and Interest at Maturity, Periodic Interest Settlement with Principal at Maturity, Periodic Interest Settlement with Flexible Principal Repayment, Equal Principal, Equal Principal and Interest
Account Type Options	Non-revolving Loan Account, Sub-account under Revolving Credit Limit, Revolving Loan Account
Constraint Rule 1	(total months of bad debt + total months of overdue + total months of settled + n) $\leq$ m Where: • m = total months from repayment record start month to repayment record end month (inclusive) • n = random integer between 5-12
Constraint Rule 2	Current overdue periods cannot exceed total months of overdue
Account Credit Limit Rule	Only has value in revolving loan accounts, value should be assigned based on user attributes
Loan Amount Rule	Only has value in non-revolving loan accounts and sub-accounts under revolving credit limit, value should be assigned based on user attributes
Repayment Periods Rule	Empty for revolving loan accounts, a specific number for non-revolving loan accounts and sub-accounts under revolving credit limit, this number should be assigned based on user attributes
Current Overdue Periods Rule	Value is determined based on account status, user attributes, and total months of overdue. Only when account status is Overdue, its value is greater than 0, otherwise it equals 0
Currency Assignment	Value should be assigned based on user attributes
Field Format Examples	**Date Fields:** opening_date="2013.07.22" (YYYY.MM.DD); closing_date="2015-05-05" (YYYY-MM-DD); repayment record start month="2023-01" (YYYY-MM); repayment record end month="2023-12". **Account Type:** account_type="Revolving Loan Account" (options: Non-revolving Loan Account, Sub-account under Revolving Credit Limit, Revolving Loan Account). **Business & Methods:** business_type="Personal Consumer Loan" (options: Personal Commercial Housing Loan, Personal Housing Provident Fund Loan, Commercial Student Loan, Stock Pledge Repo Transaction, Personal Consumer Loan); ...
Output Format	The account situation should be output in JSON object format. The overall output is a JSON object array, where each JSON object corresponds to one account. The keys within the object are the required fields for the account, and the values need to be reasonably assigned by you based on the user's specific situation. Note: Do not output any thinking process or extra content. Output strictly according to the specified format.

Table 6: Prompt Credit Card Account Data Generation

Section	Content
Role Definition	You are an experienced credit review expert with extensive experience in credit application approval.
Task Objective	You need to generate the user's loan information based on the following requirements according to the user's credit status, economic situation, residential situation, and employment situation.
Input Parameters	• User's average monthly income: {average_month_income} • User's credit rating: {credit_rating} • Residential situation: {live_tab} • Employment situation: {prof_tab}
Output Requirement	Please generate this user's credit card usage situation
Required Fields	Issuing institution, Shared credit limit, Shared used limit, Used limit, Average used limit in recent 6 months, Maximum used limit, Remaining installment periods, Opening date, Cutoff date, Repayment record start month, Repayment record end month, Account credit limit, Currency, Business type, Guarantee method, Account status, Due date, Repayment due this month, Actual repayment this month, Current overdue periods, Total periods of bad debt, Total months of overdue, Total months of settled, Business type
Account Status Options	Normal, Overdue, Bad Debt, Settled
Constraint Rule 1	(total months of bad debt + total months of overdue + total months of settled + n) $\leq$ m Where: • m = total months from repayment record start month to repayment record end month (inclusive) • n = random integer between 5-12
Constraint Rule 2	Current overdue periods cannot exceed total months of overdue
Current Overdue Periods Rule	Value is determined based on account status, user attributes, and total months of overdue. Only when account status is Overdue, its value is greater than 0, otherwise it equals 0
Currency Assignment	Value should be assigned based on user attributes
Field Format Examples	**Date Fields:** opening_date="2020.05.10" (YYYY.MM.DD); cutoff_date="2024.10.31"; due_date="2024.10.25"; start_time="2023-01" (YYYY-MM); end_time="2023-12". **Account Type:** account_type="Credit Card" (fixed, can only be Credit Card). **Status & Institution:** issuing_institution="Issuing Institution AQ"; business_type="Credit Card"; guarantee_method="Unsecured"; account_status="Overdue"; currency="RMB". **Credit Limits:** shared_credit_limit="50000"; shared_used_limit="30000" ($\leq$ shared credit limit); credit_limit="50000" (independent); used_limit="32000" (independent usage); avg_used_limit_6m="28000"; max_used_limit="45000". **Repayment Fields:** this_month_due="5000" (estimated based on status, balance, method, overdue periods); actual_repayment="0" (when status is Overdue or Bad Debt: actual $<$ due; otherwise: actual $=$ due); remaining_installment_periods="12". **Period Counts:** b_count=2 (total months of bad debt); num_count=3 (total months of overdue); c_count=2 (total months of settled); current_overdue_periods="3" (must be $\leq$ num_count, $>$0 only when status is Overdue).
Output Format	The account situation should be output in JSON object format. The overall output is a JSON object array, where each JSON object corresponds to one account. The keys within the object are the required fields for the account, and the values need to be reasonably assigned by you based on the user's specific situation. Note: Do not output any thinking process or extra content. Output strictly according to the specified format.

Prompt for Generating User Basic Information (Residence/Occupation)

You are an experienced credit review expert with extensive experience in credit application approval. You need to generate user information based on the following requirements according to the user's credit status and economic situation. You must ensure that the generated information matches the user's credit status and economic situation, and also ensure content diversity. User's average monthly income: {average_month_income}, User's credit rating: {credit_rating}. Please generate this user's residential situation and employment situation. The residential situation should be output in JSON object array format, where each JSON object includes: number, residential address, home phone, residence status, and information update date. The employment situation should be output in JSON object array format, where each JSON object includes: number, employer, employer type, employer address, employer phone, occupation, industry, position, professional title, year of joining current employer, and information update date. The number of JSON objects in the JSON object arrays is not fixed. You can decide based on the user's situation. The information within each JSON object represents this user's residential address and employment situation during that period, showing the overall changes in the user's residential and employment situations. You need to ensure diversity in the user's personal situation - do not concentrate in one residential area or workplace, and both m and n should be considered according to the user's personal situation. Follow the format of the example below. Note: Do not output any thinking process or extra content. Output strictly according to the given format.
The overall structure is a JSON object containing two fields, each with a JSON object array as its value, where n is the number of residential situation JSON objects and m is the number of employment situation JSON objects:
{{ "live_tab": [{{ "number": "1", "residential_address": "Room C5XX, Building 7, Chunxiao Garden North District, Chaoyang District, Beijing", "home_phone": "010-832XX", "residence_status": "Mortgage", "info_update_date": "2015.05.01" }}, ...}, "prof_tab": {{{ "number": "1", "employer": "Corporate Department, Credit Reference Center, People's Bank of China", "employer_type": "Government/Public Institution", "employer_address": "Room 305, Building A, International Enterprise Building, No. XX Financial Street, Xicheng District, Beijing", "employer_phone": "010-83233XX", "occupation": "Clerical and Related Workers", "industry": "Financial Industry", "position": "Mid-level Management", "professional_title": "Intermediate", "year_joined": "2008", "info_update_date": "2015.05.01" }}, ...}}}

Figure 7: Prompt for User Basic Information (Residence/Occupation).

Prompt For Generation of Abstract Definitions for Tables

Generate Python code to define a class for the credit report table in the image.
Requirements:
1. Class Structure
- Class name should reflect the table type.
- Include ALL fields from the target columns: {tar_columns}
- Follow the table metadata specifications: {table_meta_information}
2. Required Methods
a) Constructor (__init__)
- Initialize all fields with appropriate default values
- Include type hints for all parameters
- Validate data types if necessary
b) to_json() method
- Convert the object to JSON format
- Return a dictionary that can be serialized
- Handle nested structures if any
- Include proper encoding for Chinese characters
c) to_json_file(filename)
d) to_latex()
- Generate LaTeX table code for the object
- Include table headers matching the original format
3. Data Type Specifications
- Use appropriate Python types (int, float, str, datetime, etc.)
- Follow the data formats defined in {table_meta_information}
- Add validation for required fields

Figure 8: Prompt for Generation of Abstract Definitions for Tables.

Prompt For Generation of Indicators Corresponding to Tables

Based on the credit report table(s) in the image, generate a comprehensive list of assessment indicators for evaluating loan applicants. Input Information:- Table Details: {table_details}- Assessment Focus: {task_type} Requirements:
1. Indicator Generation
- Analyze the table structure and available data fields
- Generate relevant indicators that focus on {task_type}
- Each indicator should be calculable from the table data
2. Output Format for Each Indicator:
- Initialize all fields with appropriate default values
- Include type hints for all parameters
- Validate data types if necessary
b) to_json() method
- Convert the object to JSON format
- Return a dictionary that can be serialized
- Handle nested structures if any
- Include proper encoding for Chinese characters
c) to_json_file(filename)
d) to_latex()
- Generate LaTeX table code for the object
- Include table headers matching the original format
3. Data Type Specifications
- Use appropriate Python types (int, float, str, datetime, etc.)
- Follow the data formats defined in {table_meta_information}
- Add validation for required fields

Figure 9: Prompt for Generation of Indicators Corresponding to Tables.

Prompt for Generating Calculation Functions and Questions of Indicators Corresponding to Tables

Generate a Python calculation function for the indicator: {tar_indicator}Input Information:- Indicator Category: {task_type}- Target Indicator: {tar_indicator}- Table Class Definition: {table_python_definition} Task: Create a Python function that calculates {tar_indicator} using the provided table class as input parameter.---RETURN VALUE FIELD DEFINITIONS: Before implementing the function, understand what each return field represents:
1. indicator_name (str)
2. answer (float/int/str/None) - The final calculated result of the indicator
3. question (str) - The specific assessment question that this indicator answers - Should be a clear, business-oriented question - Examples: * "What is the applicant's total debt burden?"
4. task_type (str) - The category or type of credit assessment this indicator is {task_type}
5. question_detail (str) - A detailed, step-by-step explanation of the complete calculation process
- Should be written in natural language that non-technical stakeholders can understand
- Must include: * What data was extracted from the table * Each calculation step with actual values * The logic/reasoning behind each step * How the final result was derived * Any assumptions or data quality notes * Technical metadata (formula, input fields, warnings)
- Format as a narrative or numbered steps
- Example format: ``` "CALCULATION OF TOTAL DEBT BURDEN
==========
Step 1: Data Extraction Step 2: Calculation Process Step 3: Validation Step 4: Final Result
```---FUNCTION REQUIREMENTS:
Function Signature: ```python def calculate_{indicator_name}(table_obj) -> dict: """ Returns:
dict: {
'indicator_name': str, 'answer': float/int/str,
'question': str, 'task_type': str,
'question_detail': str
} """
```

Prompt for Generating Calculation Functions with MASK

```
# Prompt Ternplate: Add Masking Logic to Calculation Function.
You have an existing calculation function `{calcu_function}` that contains a question: {question}. From this question, we have extracted: Field names (comple_column_list): `{tar_comple_column_list}` - Data fields strongly related to the calculation target Operators (comple_equ_list): `{tar_comple_equ_list}` - Calculation operators used in the question Important: Both lists are ordered sequentially as elements appear in the question string.
---1. Task Requirements. Modify the calculation function to implement the following masking logic:
1) Field Name Masking- Randomly select [1, len(comple_column_list) - 2] distinct field names from `comple_column_list`- Use string matching to locate each selected field name in the `question` string- Replace the field name at its corresponding position with "XXXX"- Add the replaced field names to `comple_column` (semicolon-separated string)
2) Operator Masking- Randomly select [1, min[4, len(comple_equ_list))] operators from `comple_equ_list`- Use string matching to locate each selected operator in the `question` string- Replace the operator at its corresponding position with "YYYY"- Add the replaced operators to `comple_equ` (semicolon-separated string)
3) Final Output- The final `question` should contain both “XXXX” and “YYYY” replacements- `comple_column`: Semicolon-separated string of masked field names- `comple_equ`: Semicolon-separated string of masked operators.
2. Function Signature
```---FUNCTION REQUIREMENTS:
Function Signature: ```python def calculate_mask_{calcu_function, comple_column_list, comple_equ_list}{table_obj) -> dict:
""" Returns:
dict: {
'indicator_name': str, 'answer': float/int/str, 'question': str, 'task_type': str, 'question_detail': str, 'comple_column': str, 'comple_equ': str,
}
```

Figure 11: Prompt for Generating Calculation Functions with MASK.

needs and accurately reflects the complexity of actual credit evaluation workflows in contemporary consumer lending operations.

### C.1 Dataset Details

### C.2 Question Sample

This benchmark evaluates the model's capabilities in processing credit report tables for credit assessment tasks across three dimensions: table structure perception, domain knowledge application, and numerical computation. The result of each metric serves as part of the basis for the final credit assessment decision. These metrics primarily evaluate loan applicants from four perspectives: repayment capacity, repayment willingness, debt pressure, and cash flow conditions.
```

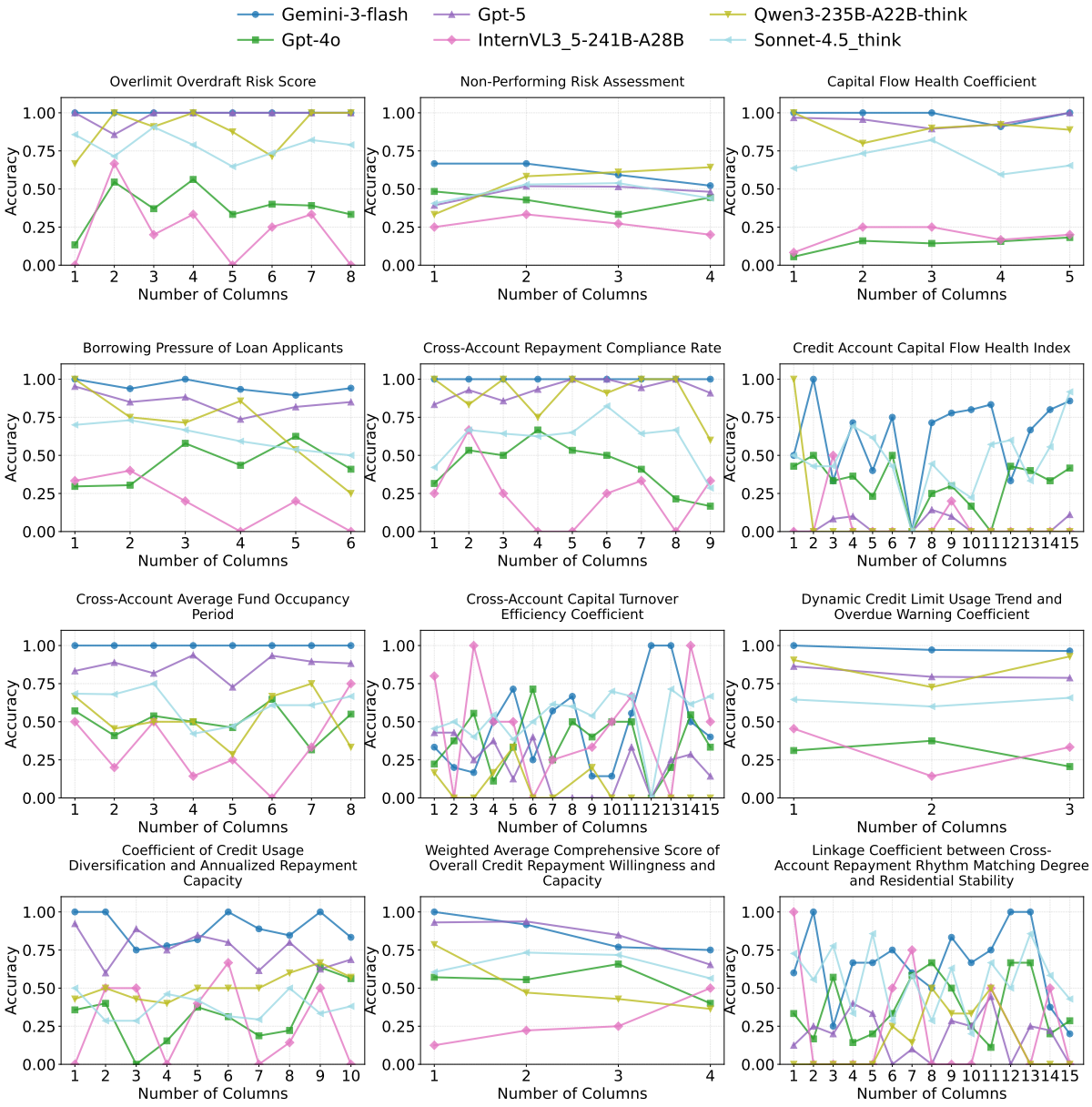

Figure 12: Relationship between the number of knowledge elements to be supplemented for a single metric (denoted as Number of Columns in the prompt) and the final table understanding (TU) accuracy. Here, Number of Columns refers to the quantity of knowledge elements that need to be recovered in the prompt, while Accuracy denotes the resulting TU accuracy across different metrics.

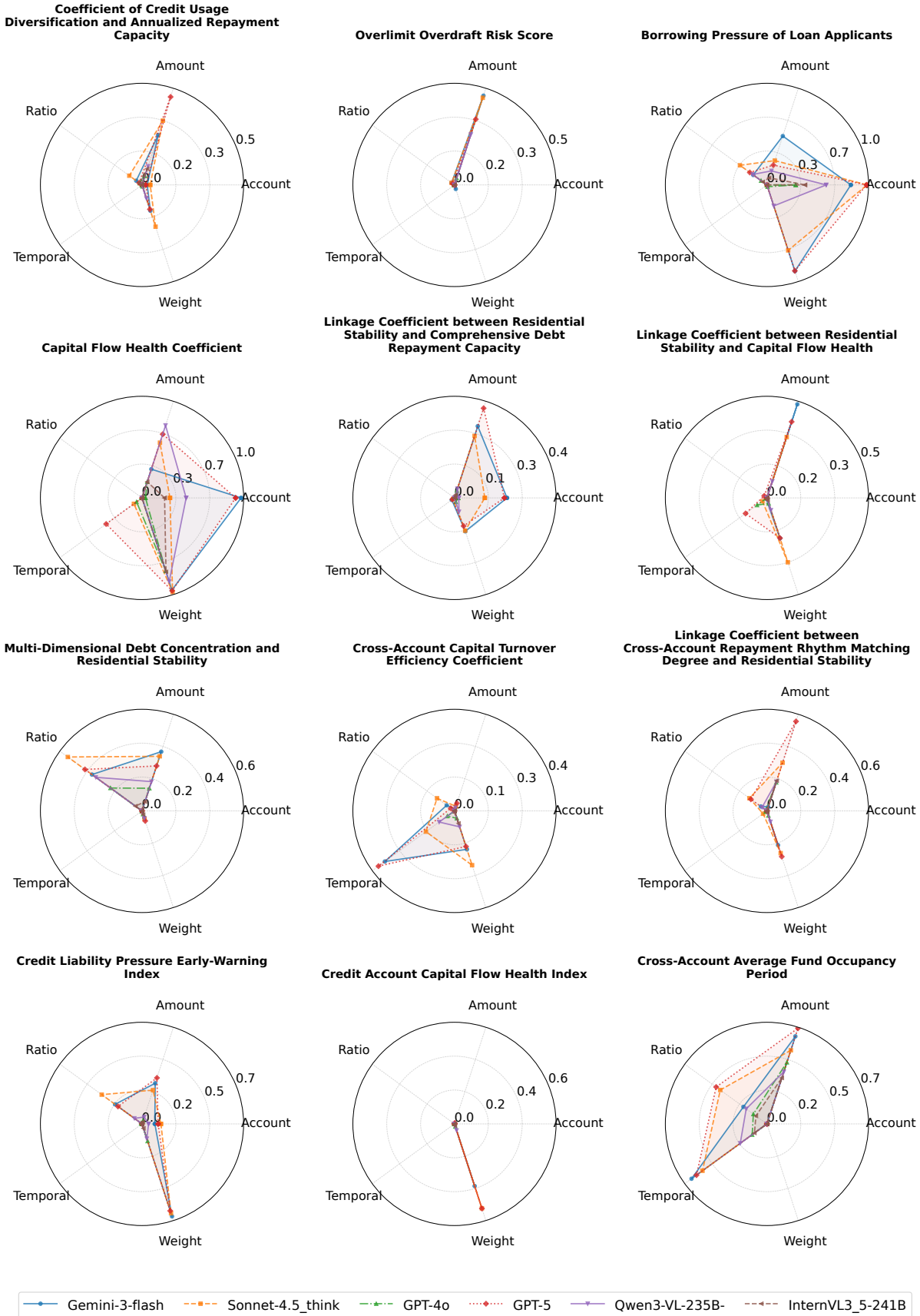

Figure 13: Demonstrate the distribution of knowledge hit rates across different metrics.

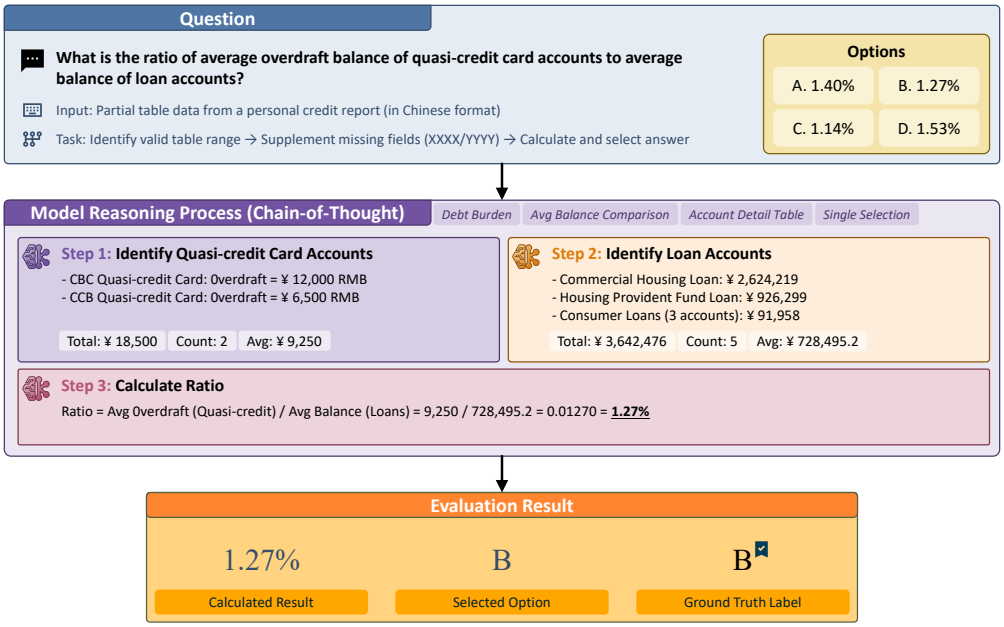

Figure 14: Demonstrate of Table Structure Perception

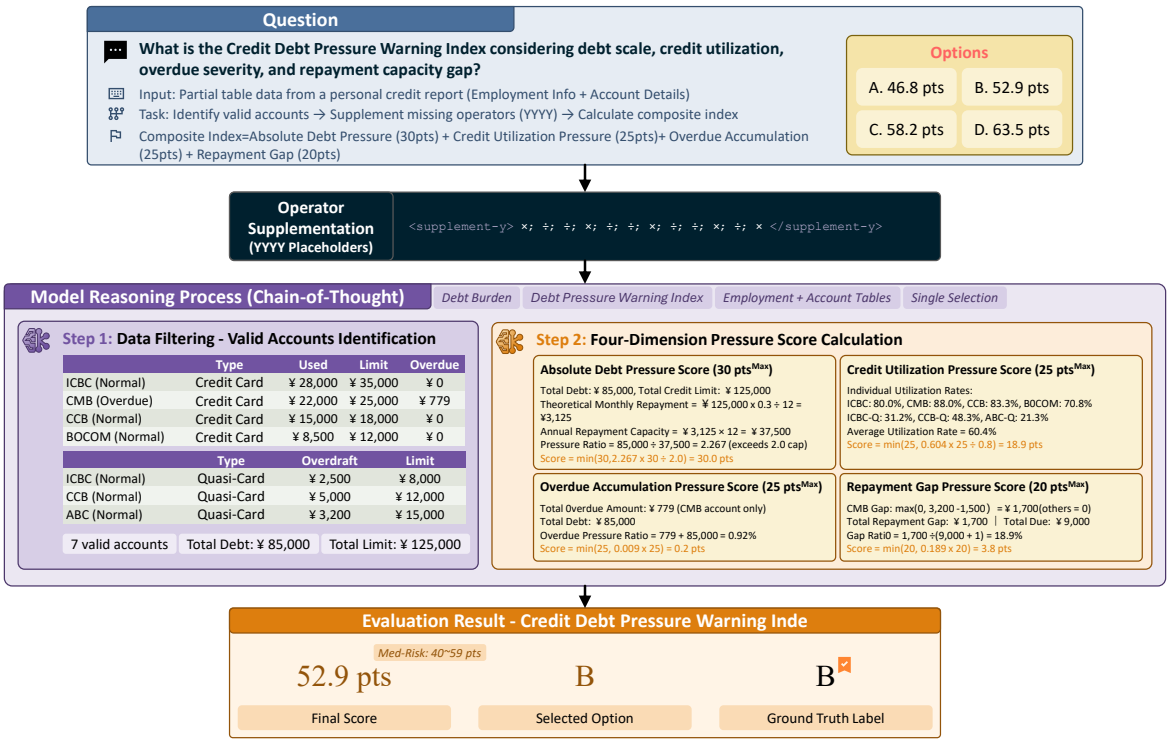

Figure 15: Demonstrate of Domain Knowledge Utilization

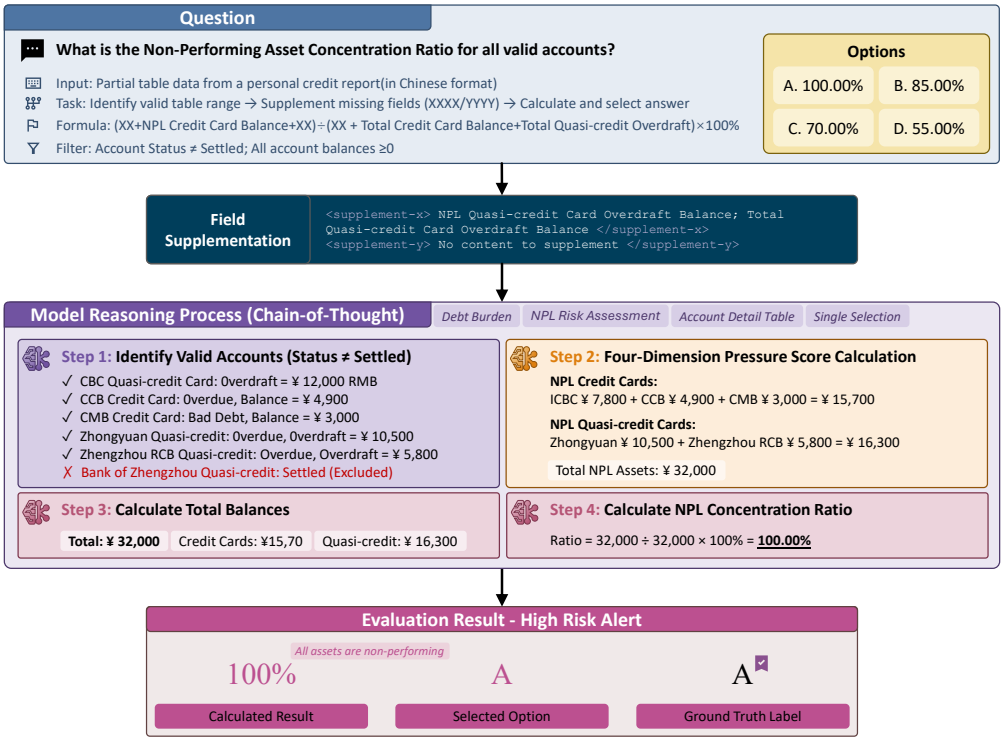

Figure 16: Demonstrate of Numerical Computation

Account (Credit Agreement ID: ZDJK20250015)												
Managing Institution			Opening Date			Account Credit Limit			Loan Amount			
Commercial Bank PD			2020.08.15			2849450			2800000			
Account Type			Currency			Business Type			Guarantee Type			
Non-revolving Loan Account			CNY			Personal Commercial Housing Loan			Mortgage			
Account Status			Five-tier Classification			Payment Due Date			Monthly Payment Due			
Normal			N/A			2023.12.25			15800			
Actual Monthly Payment			Last Payment Date			Total Payment Terms			Remaining Payment Terms			
15800			2023.12.25			360			318			
Maturity Date			Current Overdue Periods			Current Overdue Amount			As of Date			
2050.07.31			0			0			Aug 15, 2050			
Principal Overdue 31-60 Days						Principal Overdue 61-90 Days						
0						0						
Balance: 2624219												
Bad Debt Periods: N/A												
Repayment History | As of 2023-12												
	1	2	3	4	5	6	7	8	9	10	11	12
2023	N	N	N	N	N	N	N	N	N	N	N	N
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2022	N	N	N	N	N	N	N	N	N	N	N	N
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2021	N	N	N	N	N	N	N	N	N	N	N	N
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2020	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N	N	N	N
	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.00	0.00	0.00	0.00
Account Closure Date: N/A												

Table 7: Credit Account Table Details

Account (Credit Agreement ID: ZDJK20250015)												
Managing Institution		Opening Date		Account Credit Limit		Loan Amount						
Commercial Bank XH		2022.05.20		279066		180000						
Account Type		Currency		Business Type		Guarantee Type						
Sub-account under Revolving Credit		CNY		Personal Consumer Loan		Unsecured						
Account Status		Five-tier Classification		Payment Due Date		Monthly Payment Due						
Normal		N/A		2023.12.18		5580						
Actual Monthly Payment		Last Payment Date		Total Payment Terms		Remaining Payment Terms						
5580		2023.12.18		36		16						
Maturity Date		Current Overdue Periods		Current Overdue Amount		As of Date						
2025.04.30		0		0		May 20, 2025						
Principal Overdue 31-60 Days				Principal Overdue 61-90 Days								
0				0								
Balance: 91958												
Bad Debt Periods: N/A												
Repayment History | As of 2023-12												
	1	2	3	4	5	6	7	8	9	10	11	12
2023	N	N	N	N	N	N	N	N	N	N	N	N
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2022	N/A	N/A	N/A	N/A	N/A	N	N	N	N	N	N	N
	N/A	N/A	N/A	N/A	N/A	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Account Closure Date: N/A												

Table 8: Credit Account Details

Occupation Information (ID: 1)			
ID	1	Company	Zhengzhou Shunda Logistics Co., Ltd.
Company Nature	Private Enterprise	Company Address	No. 12, Area B, Logistics Park, Intersection of West 3rd Ring Road and Longhai Road, Zhongyuan District, Zhengzhou City, Henan Province
Company Phone	0371—67123456	Occupation	Commerce and Service Personnel
Industry	Transportation, Storage and Postal Services	Position	General Staff
Title	None	Entry Year	2021
Update Date	2023.08.15		

Table 9: Occupation Information Table Details

Account (Credit Agreement ID: ZDJK20250015)												
Managing Institution		Opening Date		Account Credit Limit		Loan Amount						
Commercial Bank ZJ		2021.11.10		300000		238907						
Account Type		Currency		Business Type		Guarantee Type						
Revolving Loan Account		CNY		Personal Consumer Loan		Unsecured						
Account Status		Five-tier Classification		Payment Due Date		Monthly Payment Due						
Normal		N/A		2023.12.21		1200						
Actual Monthly Payment		Last Payment Date		Total Payment Terms		Remaining Payment Terms						
1200		2023.12.21		N/A		N/A						
Maturity Date		Current Overdue Periods		Current Overdue Amount		As of Date						
2024.03.21		0		0		Nov 10, 2026						
Principal Overdue 31-60 Days				Principal Overdue 61-90 Days								
0				0								
Balance: 0												
Bad Debt Periods: N/A												
Repayment History | As of 2023-12												
	1	2	3	4	5	6	7	8	9	10	11	12
2023	N	N	N	N	N	N	N	N	N	N	N	N
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2022	N	N	N	N	N	N	N	N	N	N	N	N
	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2021	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N
	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	0.00
Account Closure Date: N/A												

Table 10: Revolving Credit Account Details

Residence Information				
ID	Address	Phone	Status	Update Date
1	Room 502, Unit 2, Building 3, No. XXX Zhengzhou City	0371—67845621	Rented	2023.08.15

Table 11: Residence Information Table Details

Credit Agreement 1				
Management Institution	Credit Agreement ID	Effective Date	Maturity Date	Credit Limit Purpose
Commercial Bank ZZ	E705	2021.09.15	2024.08.31	Non-revolving Loan Limit
Credit Limit	Credit Quota	Credit Quota ID	Used Limit	Currency
71747	71747	M806	19831	CNY

Credit Agreement 2				
Management Institution	Credit Agreement ID	Effective Date	Maturity Date	Credit Limit Purpose
Commercial Bank HN	E357	2023.08.20	2025.03.15	Revolving Loan Limit
Credit Limit	Credit Quota	Credit Quota ID	Used Limit	Currency
15000	15000	M953	0	CNY

Credit Agreement 3				
Management Institution	Credit Agreement ID	Effective Date	Maturity Date	Credit Limit Purpose
Commercial Bank JR	E788	2024.11.28	2025.10.31	Revolving Loan Limit
Credit Limit	Credit Quota	Credit Quota ID	Used Limit	Currency
96148	96148	M626	7993	CNY

Credit Agreement 4				
Management Institution	Credit Agreement ID	Effective Date	Maturity Date	Credit Limit Purpose
ICBC Zhengzhou Branch	E680	2021.09.15	2024.12.31	Credit Card Shared Limit
Credit Limit	Credit Quota	Credit Quota ID	Used Limit	Currency
15000	15000	M461	12800	CNY

Credit Agreement 5				
Management Institution	Agreement ID	Effective Date	Maturity Date	Credit Limit Purpose
Zhengzhou Bank	E326	2021.09.15	2024.12.31	Credit Card Independent Limit
Credit Limit	Credit Quota	Credit Quota ID	Used Limit	Currency
15000	15000	M952	12800	CNY

Credit Agreement 6				
Management Institution	Agreement ID	Effective Date	Maturity Date	Credit Limit Purpose
CCB Zhengzhou Jinshui Branch	E287	2023.03.20	2024.12.31	Credit Card Shared Limit
Credit Limit	Credit Quota	Credit Quota ID	Used Limit	Currency
8000	8000	M064	7200	CNY

Credit Agreement 7				
Management Institution	Agreement ID	Effective Date	Maturity Date	Credit Limit Purpose
CCB Jinshui Branch	E468	2023.03.20	2024.12.31	Credit Card Independent Limit
Credit Limit	Credit Quota	Credit Quota ID	Used Limit	Currency
8000	8000	M526	7200	CNY

Table 12: Credit Agreement Tables Details.