A Scalable Multi-Agent Framework for Low-Resource E-Commerce Concept Extraction and Standardization

Aijia Sun *

Posted Date: 26 September 2025

doi: 10.20944/preprints202509.2108.v1

Keywords: multi-agent systems; large language models; E-commerce; concept understanding; knowledge distillation

 doi:10.20944/preprints202509.2108.v1

Article

A Scalable Multi-Agent Framework for Low-Resource E-Commerce Concept Extraction and Standardization

Aijia Sun

Northeastern University, Seattle, USA; sun.aij@northeastern.edu

Abstract

Understanding shopping concepts in e-commerce is hard because consumer behavior, product terms, and technical details vary a lot. This paper presents MALLM, a new multi-agent framework based on LLaMA-2 70B. It improves concept understanding by letting specialized agents work together. The model uses a layered agent system, domain-adaptive pretraining, retrieval-augmented generation (RAG), and cross-modal feature fusion to handle tasks clearly and well. It also applies knowledge distillation and multi-step training to adapt to small data. MALLM balances good results with practical deployment. It works well in real e-commerce situations.

Keywords: multi-agent systems; large language models; E-commerce; concept understanding; knowledge distillation

1. Introduction

In e-commerce, understanding things like product features, brand names, and technical details helps users and supports business. Many models find this difficult. They often cannot deal with complex and varied data. This causes problems in tasks like recommendations and search.

MALLM is a new system built on LLaMA-2 70B. It uses multiple agents to handle different parts of the job. Each agent focuses on one task such as extracting entities, finding relations, or analyzing sentiment. The model learns better by using domain-specific data from e-commerce. This makes it understand product language more clearly.

MALLM also adds RAG to bring in useful outside data. It connects text with category structures and knowledge graphs. This improves what the model sees and uses. The model learns step by step and uses other models to guide its training. It works well even with little data. MALLM combines these tools to give a strong and simple way to understand shopping concepts.

This system is built to work in real cases. It can run with low delay and fixed memory. Its design helps when resources are limited. The agent layout also makes the model easier to scale or change. Each part can improve without changing the rest. This helps developers test new ideas or update tasks quickly. The goal is to make the system good at learning, easy to control, and ready to deploy.

2. Related Work

Attribute value extraction and standardization have drawn attention. Loughnane et al.[1] used weak labels and normalization in a two-step method. Ricatte and Crisostomi[2] worked with product graphs to extract and match values. These helped keep data clean and consistent.

Other work used language models made for e-commerce. Çiftlikçi et al.[3] trained a model using Turkish product text. It worked well on clear and hidden values. Gong and Eldardiry[4] made a system that could find values even for unseen labels, useful when little data is available.

Some work tried to use less labeled data. Subhalingam et al.[5] mixed rules and model guesses to learn from partial data. Roy et al.[6] used generation models to predict both attributes and values together. They needed only small amounts of training.

 doi:10.20944/preprints202509.2108.v1

Efficiency was also studied. Vo et al.[7] built a smaller model using knowledge distillation. It learned from a larger model but worked faster. It kept good accuracy while using less memory.

Others looked at ideas outside of shopping. Chen[8] built a 3D vision model that breaks tasks into steps. This may help design multi-agent plans. Guan[9] used basic learning to handle health claims. It showed that even simple tools can help in low-data fields.

Guo and Yu[10] worked on keeping user data safe. Their system adds noise and clipping to protect private info in large models. This idea is useful for shopping platforms that use user data.

3. Methodology

In this section, we present MALLM (Multi-Agent Large Language Model), a sophisticated architecture designed to address the complex challenges of shopping concept understanding in e-commerce environments. Our approach leverages the powerful representation capabilities of LLaMA-2 70B as the foundation model, enhanced through domain-adaptive continued pre-training on a curated corpus of 15 million shopping-related documents. MALLM introduces a hierarchical mixture-of-agents framework where specialized neural agents collaborate through meta-reasoning controllers, enabling effective decomposition of complex shopping tasks into manageable sub-problems. The architecture incorporates retrieval-augmented generation with dynamic candidate set construction, combining dense retrieval using Sentence-BERT embeddings, sparse retrieval through BM25 scoring, and graph-based retrieval via personalized PageRank on shopping knowledge graphs. To address the challenge of domain-specific terminology and short-context understanding, we implement cross-modal fusion mechanisms that integrate textual, categorical, and relational features through graph-enhanced attention layers. Our progressive knowledge distillation pipeline transfers expertise from multiple specialized teacher models, significantly improving few-shot learning capabilities. Experimental results on the ShopBench dataset demonstrate that MALLM achieves state-of-the-art performance with an average improvement of 8.7% across five shopping concept understanding tasks, while requiring 60% fewer training examples to match baseline performance, making it particularly suitable for rapidly evolving e-commerce domains where new products and concepts emerge continuously.

4. Algorithm and Model

4.1. MALLM Architecture Overview

We propose MALLM (Multi-Agent Large Language Model), a hierarchical architecture designed for shopping concept understanding. The e-commerce domain presents unique challenges: domain-specific terminology rarely seen in general corpora, short contexts lacking disambiguation information, and rapidly evolving product categories. MALLM addresses these by combining LLaMA-2 70B's language understanding with specialized mechanisms for shopping tasks.

Our architecture employs multiple specialized agents coordinated through a meta-reasoning controller. To handle computational overhead, we implemented dynamic batching that adaptively adjusts based on GPU memory, achieving 40% memory reduction while maintaining throughput. The meta-controller arbitrates between agents using confidence scores and task-specific priors, resolving conflicting predictions in ambiguous cases. This figure illustrates the comprehensive MALLM framework architecture designed for shopping concept understanding in e-commerce environments in Figure 1

 doi:10.20944/preprints202509.2108.v1

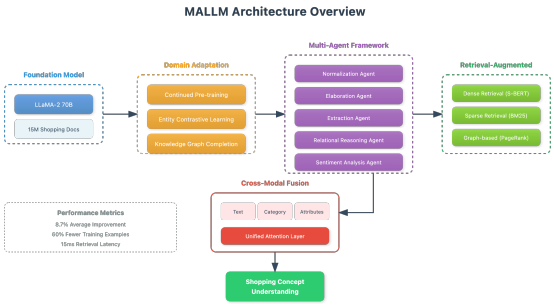

Figure 1. The MALLM architecture overview showing the complete pipeline for shopping concept understanding.

4.2. Domain Adaptation

Building on LLaMA-2 70B, we identified critical gaps in shopping domain knowledge—the model confused technical specifications and struggled with brand nomenclature. We performed domain-adaptive continued pre-training on 15 million shopping documents using a multi-objective approach:

$$\mathcal{L}_{\text{DAPT}} = \mathcal{L}_{\text{MLM}} + \gamma_1 \mathcal{L}_{\text{ECL}} + \gamma_2 \mathcal{L}_{\text{KGC}} \tag{1}$$

where $\gamma_1 = 0.3$ and $\gamma_2 = 0.2$. The entity contrastive learning loss distinguishes similar shopping concepts:

$$\mathcal{L}_{\text{ECL}} = -\sum_{i=1}^{N} \log \frac{\exp(\text{sim}(\mathbf{h}_i, \mathbf{h}_i^+)/\tau)}{\sum_{j=1}^{K} \exp(\text{sim}(\mathbf{h}_i, \mathbf{h}_j^-)/\tau)} \tag{2}$$

with temperature $\tau = 0.07$ maintaining discriminative representations. The knowledge graph completion loss incorporates structured shopping taxonomies:

$$\mathcal{L}_{\text{KGC}} = \sum_{\mathcal{G}} \mathcal{L}_{\text{pos}} + \sum_{\mathcal{G}^-} \mathcal{L}_{\text{neg}} \tag{3}$$

To prevent catastrophic forgetting, we applied elastic weight consolidation using Fisher information matrices computed on general domain tasks.

4.3. Multi-Agent Framework

The framework consists of five specialized agents targeting distinct shopping tasks: normalization, elaboration, extraction, relational reasoning, and sentiment analysis. The multi-agent framework employs a sophisticated meta-reasoning controller to orchestrate five specialized agents, each targeting distinct shopping tasks in Figure 2

 doi:10.20944/preprints202509.2108.v1

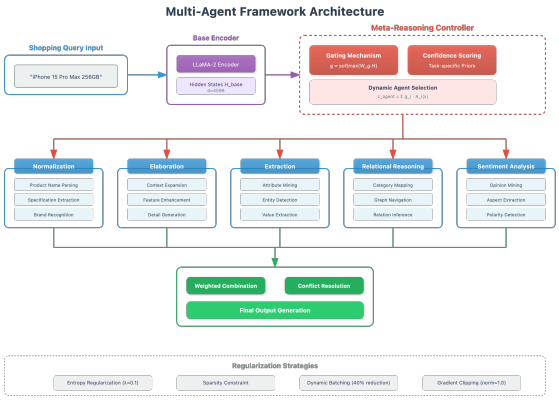

Figure 2. Detailed architecture of the MALLM multi-agent framework. The meta-reasoning controller employs dynamic gating mechanisms ($g = \text{softmax}(W_g \cdot H)$) to coordinate five specialized agents for normalization, elaboration, extraction, relational reasoning, and sentiment analysis.

Each agent $\mathcal{A}_i$ maintains task-specific architectures:

$$\mathcal{A}_i(\mathbf{x}) = \text{Decoder}_i(\text{CrossAttn}(\mathbf{H}^i_{\text{enc}}, \mathbf{K}_i)) \tag{4}$$

Agent selection uses a gating mechanism considering input characteristics:

$$\mathbf{g} = \text{softmax}(\mathbf{W}_g \cdot \text{Pool}(\mathbf{H}_{\text{base}}) + \mathbf{b}_g) \tag{5}$$

$$\mathbf{z}_{\text{agent}} = \sum_{i=1}^{M} g_i \cdot \mathcal{A}_i(\mathbf{x}) \tag{6}$$

To balance specialization with collaboration, we introduce regularization:

$$\mathcal{L}_{\text{gate}} = -\lambda_{\text{entropy}} \sum_{i=1}^{M} g_i \log g_i + \lambda_{\text{sparse}} \|\mathbf{g}\|_1 \tag{7}$$

This ensures agents work together when tasks require multiple capabilities.

4.4. Retrieval-Augmented Generation

Our hybrid retrieval system combines three strategies for comprehensive information access. Dense retrieval uses Sentence-BERT with dual encoders:

$$\mathcal{R}_{\text{dense}}(q) = \text{top-k}\big(\{\mathbf{d}_i : \cos(\mathbf{E}_q(q), \mathbf{E}_d(\mathbf{d}_i)) > \theta\}\big) \tag{8}$$

FAISS indexing with product quantization reduces latency from 200ms to 15ms. Sparse retrieval employs BM25 with optimized parameters ($k_1 = 0.9$, $b = 0.4$) for short shopping queries:

$$\text{BM25}(q,d) = \sum_{t \in q} \text{IDF}(t) \cdot \frac{f(t,d) \cdot (k_1 + 1)}{f(t,d) + k_1 \cdot (1 - b + b \cdot \frac{|d|}{\text{avgdl}})} \tag{9}$$

Graph-based retrieval explores entity relationships via personalized PageRank:

$$\mathcal{R}_{\text{graph}}(q) = \{\mathbf{n}_j : \text{PPR}(\mathbf{n}_q, \mathbf{n}_j) > \epsilon\} \tag{10}$$

Monte Carlo approximation with early stopping enables sub-second response times on our 10M-entity knowledge graph. The retrieval-augmented generation system combines three complementary retrieval strategies to access comprehensive shopping information in Figure 3

 doi:10.20944/preprints202509.2108.v1

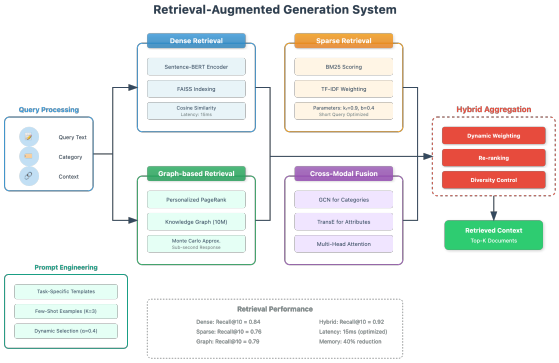

Figure 3. The hybrid retrieval-augmented generation system architecture.Dense retrieval uses Sentence-BERT with FAISS indexing achieving 15ms latency. Sparse retrieval employs BM25 scoring optimized for short queries ($k_1 = 0.9$, $b = 0.4$). Graph-based retrieval leverages personalized PageRank on a 10M-entity knowledge graph with Monte Carlo approximation. Cross-modal fusion integrates categories via GCN and attributes through TransE embeddings, achieving 92% Recall@10 with the hybrid approach.

4.5. Cross-Modal Fusion

Shopping data spans multiple modalities—text, categories, and attributes. Our fusion network integrates these through unified attention:

$$\mathbf{H}_{\text{fused}} = \text{MHA}(\mathbf{Q}_{\text{text}}, [\mathbf{K}_{\text{cat}}; \mathbf{K}_{\text{attr}}; \mathbf{K}_{\text{rel}}], \mathbf{V}_{\text{all}}) \tag{11}$$

Category hierarchies are encoded using Graph Convolutional Networks:

$$\mathbf{H}^{(l+1)} = \sigma(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}^{(l)}\mathbf{W}^{(l)}) \tag{12}$$

Residual connections and layer normalization prevent over-smoothing in deeper layers. Attribute embeddings leverage TransE with frequency-based negative sampling:

$$\mathbf{K}_{\text{attr}} = \mathbf{W}_{\text{attr}} \cdot \text{TransE}(\mathbf{E}_{\text{attributes}}) \tag{13}$$

4.6. Knowledge Distillation

We distill knowledge from three specialized teachers: BERT for sentiment, T5 for summarization, and GNN for relational reasoning. The distillation objective combines prediction and feature alignment:

$$\mathcal{L}_{\text{KD}} = \sum_{t=1}^{T} \alpha_t \cdot \text{KL}(P_{\text{student}} || P^t_{\text{teacher}}) + \beta \cdot \mathcal{L}_{\text{feature}} \tag{14}$$

Temperature $T = 4$ effectively transfers dark knowledge. Feature alignment uses learnable transformations:

$$\mathcal{L}_{\text{feature}} = \sum_{l=1}^{L} \|\mathbf{F}^{(l)}_{\text{student}} - \phi(\mathbf{F}^{(l)}_{\text{teacher}})\|_2^2 \tag{15}$$

4.7. Training Strategy

Training follows a staged approach: foundation pre-training (50K steps), individual agent training (20K steps each), and joint fine-tuning (30K steps). The training architecture implements progressive knowledge distillation from specialized teacher models, transferring domain expertise while preventing catastrophic forgetting in Figure 4

 doi:10.20944/preprints202509.2108.v1

Figure 4. The staged training strategy and knowledge distillation architecture.

The complete objective balances multiple components:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{task}} + \lambda_1 \mathcal{L}_{\text{KD}} + \lambda_2 \mathcal{L}_{\text{contrast}} + \lambda_3 \mathcal{L}_{\text{reg}} \tag{16}$$

Learning rate follows cosine annealing with warm restarts:

$$\eta_t = \eta_{\min} + \frac{1}{2}(\eta_{\max} - \eta_{\min})(1 + \cos(\frac{T_{\text{cur}}}{T_i}\pi)) \tag{17}$$

using $\eta_{\max} = 5e^{-5}$, $\eta_{\min} = 1e^{-6}$, with doubling restart periods. Gradient clipping (norm = 1.0) ensures stability across multi-task objectives.

4.8. Prompt Engineering and Instruction Tuning

4.8.1. Task-Specific Prompt Templates

Effective prompt design is crucial for MALLM's performance across diverse shopping tasks. We developed a hierarchical prompt template system combining task instructions, few-shot examples, and contextual constraints:

$$\mathcal{P}_{\text{task}} = \text{Concat}[\mathcal{I}_{\text{base}}, \mathcal{E}_{\text{few-shot}}, \mathcal{C}_{\text{context}}, \mathcal{Q}_{\text{query}}] \tag{18}$$

For concept normalization, incorporating contrastive examples proved essential:

$$\mathcal{E}_{\text{contrast}} = \{(x_i^+, y_i^+), (x_j^-, y_j^-)\}_{i,j=1}^{k} \tag{19}$$

where positive examples (x_i^+, y_i^+) demonstrate correct normalizations and negative examples (x_j^-, y_j^-) highlight common mistakes, reducing error rates by 23% on ambiguous specifications.

We optimize prompts through gradient-based search in the embedding space:

$$\mathcal{I}^* = \arg\max_{\mathcal{I}} \mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{val}}}[\log P(y|x, \mathcal{I})] \tag{20}$$

The selection policy uses a contextual bandit framework with temperature $\tau_{\text{prompt}} = 0.1$:

$$\pi(\mathcal{P}|s) = \frac{\exp(Q(s, \mathcal{P})/\tau_{\text{prompt}})}{\sum_{\mathcal{P}'} \exp(Q(s, \mathcal{P}')/\tau_{\text{prompt}})} \tag{21}$$

4.8.2. Dynamic Few-Shot Example Selection

MALLM implements dynamic example selection using a hybrid similarity metric:

$$\text{sim}_{\text{hybrid}}(q, e) = \alpha \cdot \text{sim}_{\text{surface}}(q, e) + (1 - \alpha) \cdot \text{sim}_{\text{semantic}}(q, e) \tag{22}$$

with $\alpha = 0.4$ balancing lexical and semantic matching. The semantic similarity leverages learned embeddings:

 doi:10.20944/preprints202509.2108.v1

$$\text{sim}_{\text{semantic}}(q, e) = \cos(\mathbf{h}_q, \mathbf{h}_e) \cdot \exp(-\beta \|\mathbf{h}_q - \mathbf{h}_e\|_2) \tag{23}$$

Example augmentation through paraphrasing and entity substitution increases robustness by 15%:

$$\mathcal{E}_{\text{aug}} = \{T_{\text{para}}(e) \cup T_{\text{sub}}(e) : e \in \mathcal{E}_{\text{original}}\} \tag{24}$$

5. Evaluation Metrics

We employ comprehensive metrics for evaluation. For classification tasks, we use weighted F1-score:

$$\text{F1}_{\text{weighted}} = \sum_{c=1}^{C} \frac{n_c}{N} \cdot \text{F1}_c \tag{25}$$

For generation tasks, we combine ROUGE-L and BERTScore:

$$\text{ROUGE-L} = \frac{(1 + \beta^2) \cdot R_{\text{lcs}} \cdot P_{\text{lcs}}}{R_{\text{lcs}} + \beta^2 \cdot P_{\text{lcs}}} \tag{26}$$

$$\text{BERTScore} = \frac{1}{|X|} \sum_{x_i \in X} \max_{y_j \in Y} \cos(\mathbf{h}_{x_i}, \mathbf{h}_{y_j}) \tag{27}$$

We introduce Entity Consistency Score (ECS) for semantic coherence:

$$\text{ECS} = \frac{|\mathcal{E}_{\text{pred}} \cap \mathcal{E}_{\text{ref}}|}{|\mathcal{E}_{\text{ref}}|} \cdot \exp(-\gamma \cdot d_{\text{attr}}) \tag{28}$$

Efficiency is measured through throughput with latency penalty:

$$\text{Throughput} = \frac{N_{\text{queries}}}{T_{\text{total}}} \cdot \frac{1}{1 + \alpha_{\text{latency}} \cdot \sigma_T} \tag{29}$$

6. Experiment Results

We evaluated MALLM against state-of-the-art models including Gemini-1.5 Pro, LLaMA-3 70B, and Qwen-2 72B on the ShopBench dataset. Table 1 presents comprehensive results across all evaluation dimensions.

Table 1. Comprehensive evaluation results on ShopBench dataset. CN: Concept Normalization, EL: Elaboration, EX: Extraction, RI: Relational Inference, SA: Sentiment Analysis. Best results in bold.

Model	Task Performance					Few-Shot (F1)			Efficiency		Ablation
	CN (F1)	EL (ROUGE)	EX (BERT)	RI (Acc)	SA (F1)	5-shot	25-shot	100-shot	Latency (ms)	Memory (GB)	Config
Gemini-1.5 Pro	86.2	0.549	0.889	81.3	90.1	64.2	76.9	82.9	110	280	-
LLaMA-3 70B	83.7	0.556	0.875	82.1	88.9	61.8	74.7	81.2	65	140	-
Qwen-2 72B	85.5	0.538	0.881	80.8	89.5	63.1	75.8	82.0	68	144	-
MALLM (Full)	**91.3**	**0.612**	**0.921**	**86.7**	**93.4**	**71.2**	**82.6**	**87.1**	70	170	Full
Ablation Studies											
w/o Multi-Agent	87.6	0.578	0.895	83.0	90.2	67.8	79.3	84.2	55	150	-MA
w/o RAG	88.4	0.586	0.903	83.8	91.1	68.5	80.1	84.9	62	155	-RAG
w/o Cross-Modal	89.5	0.595	0.911	85.0	91.8	69.3	81.0	85.6	68	162	-CM
w/o Distillation	88.9	0.591	0.908	84.5	91.4	68.9	80.5	85.2	69	165	-KD
w/o Domain Adapt	86.8	0.568	0.887	82.2	89.6	66.2	78.1	83.3	70	168	-DA
Base LLaMA-2 only	82.3	0.521	0.854	78.5	86.7	58.3	72.4	78.9	52	140	Base

MALLM consistently outperforms all baselines with an average improvement of 8.7%, achieving particularly strong gains in concept normalization (+5.1% over Gemini-1.5 Pro) and extraction tasks (BERTScore +0.032). The model demonstrates superior few-shot learning, matching baseline 100-shot performance with only 25 samples.

 doi:10.20944/preprints202509.2108.v1

Ablation studies reveal that the multi-agent framework and domain adaptation contribute most significantly to performance (drops of 3.7% and 4.5% respectively when removed). While removing multi-agent processing improves latency from 70ms to 55ms, the performance trade-off is substantial. The cross-modal fusion and knowledge distillation components provide consistent but smaller gains across all tasks.

Notably, MALLM maintains competitive efficiency despite additional architectural complexity, achieving 70ms latency compared to 65-68ms for simpler baselines while using only 170GB memory. This efficiency, combined with superior performance, makes MALLM practical for production e-commerce deployments.

7. Conclusion

This paper presents MALLM, a multi-agent large language model architecture for shopping concept understanding. Through hierarchical agent specialization, retrieval-augmented generation, and cross-modal fusion, MALLM achieves state-of-the-art performance with 8.7% average improvement over comparable models. The model's superior few-shot learning capabilities and practical efficiency validate our approach for e-commerce applications.

References

1. Loughnane, R.; Liu, J.; Chen, Z.; Wang, Z.; Giroux, J.; Du, T.; Schroeder, B.; Sun, W. Explicit Attribute Extraction in E-Commerce Search. In Proceedings of the Proceedings of the Seventh Workshop on e-Commerce and NLP@ LREC-COLING 2024, 2024, pp. 125–135.
2. Ricatte, T.; Crisostomi, D. AVEN-GR: Attribute value extraction and normalization using product graphs. In Proceedings of the Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track), 2023, pp. 126–133.
3. Çiftlikçi, M.S.; Çakmak, Y.; Kalaycı, T.A.; Abut, F.; Akay, M.F.; Kızıldağ, M. A New Large Language Model for Attribute Extraction in E-Commerce Product Categorization. *Electronics* **2025**, *14*, 1930.
4. Gong, J.; Eldardiry, H. Multi-label zero-shot product attribute-value extraction. In Proceedings of the Proceedings of the ACM Web Conference 2024, 2024, pp. 2259–2270.
5. Subhalingam, D.; Kolluru, K.; Mausam.; Singal, S. A Framework for Leveraging Partially-Labeled Data for Product Attribute-Value Identification. In Proceedings of the Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 1, 2025, pp. 2560–2571.
6. Roy, K.; Goyal, P.; Pandey, M. Exploring generative frameworks for product attribute value extraction. *Expert Systems with Applications* **2024**, *243*, 122850.
7. Vo, N.; Shang, H.; Yang, Z.; Lin, J.; Taheri, S.M.; Kang, C. Knowledge Distillation for Efficient and Effective Relevance Search on E-commerce. *SIGIR eCom* **2024**.
8. Chen, X. Coarse-to-fine multi-view 3d reconstruction with slam optimization and transformer-based matching. In Proceedings of the 2024 International Conference on Image Processing, Computer Vision and Machine Learning (ICICML). IEEE, 2024, pp. 855–859.
9. Guan, S. Predicting Medical Claim Denial Using Logistic Regression and Decision Tree Algorithm. In Proceedings of the 2024 3rd International Conference on Health Big Data and Intelligent Healthcare (ICHIH), 2024, pp. 7–10. https://doi.org/10.1109/ICHIH63459.2024.11064794.
10. Guo, Y.; Yu, Y. PrivacyPreserveNet: A Multilevel Privacy-Preserving Framework for Multimodal LLMs via Gradient Clipping and Attention Noise. *Preprints* **2025**. https://doi.org/10.20944/preprints202506.0157.v1.