CATSF: LEARNING ADAPTIVE REASONING FOR MULTIMODAL USER FEEDBACK CLASSIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Android users generate large volumes of multimodal feedback, making manual triage impractical at scale and motivating automated classification. While multimodal large language models exhibit strong reasoning capabilities, high computational costs hinder their deployment, and the scarcity of high-quality task-specific data and reasoning supervision limits the adaptation of compact models. Reasoning-depth selection must also account for both text–image complexity and compact-model capabilities. To address these challenges, we introduce the Android User Feedback (AUF) dataset, a multimodal benchmark containing 3,750 expert-labeled entries, and propose CATSF , a Capability-Aware Teacher–Student Framework that separates learning to execute reasoning paths from learning to select them. In the first fine-tuning stage, a compact student learns basic and complex reasoning from teacher-generated trajectories. In the second stage, the same student learns adaptive path selection from teacher targets selected based on whether its first-stage basic predictions match the training labels. Experiments show that CATSF achieves a relative macro-F_1 improvement of up to 63.4% and an accuracy gain of up to 18.3 percentage points over the evaluated baselines. Across teacher–student configurations, CATSF reduces mean output token usage by up to 50.8% relative to matched, separately fine-tuned complex-reasoning baselines.

1 INTRODUCTION

With over 3.8 billion global users by March 2026 (AndroidStatistics, 2026), the Android platform faces immense service quality challenges, exacerbated by ecosystem fragmentation (Wu et al., 2020) and diverse hardware-software compatibility issues. Despite the availability of multimodal user feedback (text and images), the massive daily volume creates an intractable bottleneck for manual triage, leading to delayed responses and resource inefficiency. Therefore, multimodal classification of feedback is essential to ensure timely diagnosis and scalable customer support. The rise of Large Language Models (LLMs) (Brown et al., 2020; Achiam et al., 2023) and their multimodal counterparts (MLLMs) (Hurst et al., 2024; Team et al., 2024) offers transformative reasoning capabilities for complex data. Leveraging these models for automated feedback analysis represents a natural and promising progression in improving service infrastructure.

However, deploying these architectures in the mobile feedback domain faces two critical challenges. First, industrial-scale triage demands a strategic balance between model performance and deployment costs. While compact models are well suited to high-throughput feedback classification, their development is hindered by a scarcity of high-quality, large-scale labeled datasets. Feedback quality is often compromised by mislabeled categories, ambiguous descriptions, and misaligned multimodal evidence. This is primarily caused by a semantic gap between professional technical taxonomies (e.g., “stability” vs. “device protection”) and the limited technical vocabulary of end-users. Without addressing these data quality issues, it is nearly impossible for compact models to internalize the specialized domain expertise required for robust multimodal feedback classification. Second, multimodal feedback exhibits highly heterogeneous reasoning complexity, requiring models to adapt their reasoning depth to different text–image relationships. Simple cases can often be resolved through direct semantic matching, whereas complex cases—such as the complementary evidence illustrated in Figure 1—may require deeper cross-modal correlation analysis. More importantly, the appropriate reasoning depth also depends on models’ ability to classify a given sample using basic reasoning. However, a static reasoning strategy like (Zhang et al., 2024; Mondal et al., 2024) applies

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

a fixed process to all samples without accounting for their differing reasoning requirements. This creates two potential failure modes: insufficient reasoning may overlook subtle multimodal cues in complex cases, while unnecessarily deep reasoning on straightforward samples may introduce overthinking and hallucinations. A basic reasoning trajectory that yields a correct answer for a large Teacher model does not necessarily indicate that basic reasoning is sufficient for a compact Student model. Teacher correctness alone therefore does not provide student-specific supervision for reasoning-path selection. Consequently, learning to select reasoning depth in a way that accounts for both multimodal complexity and Student model capability remains a critical challenge for efficient and robust feedback classification.

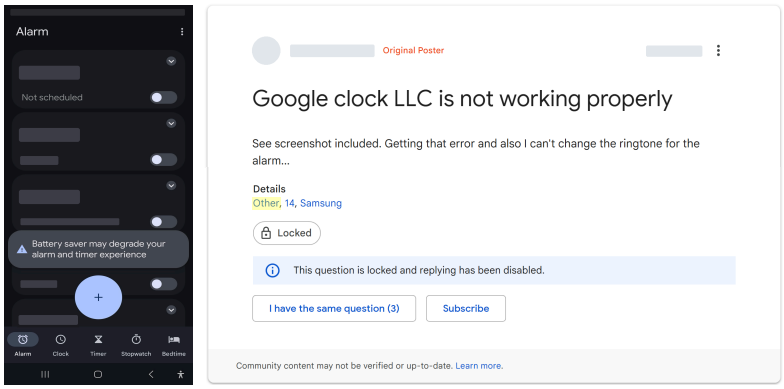


Figure 1: Illustration of complementary information in the user description and screenshot. (The text reports an alarm-related issue, while the screenshot provides an additional battery-saver warning. Personal identifiers have been redacted).

To address these challenges, we introduce a two-fold solution consisting of a new multimodal dataset and an adaptive reasoning framework. To address the first challenge, we present the Android User Feedback (AUF) dataset, a high-quality expert-annotated multimodal benchmark for Android feedback classification. Collected from Android Help Community (Android Help), AUF contains 3,750 manually labeled text–image pairs spanning diverse diagnostic categories. By aligning raw user expressions with professional technical taxonomies, the dataset provides the structured domain supervision necessary for compact models to perform reliable multimodal feedback classification. Detailed data collection, annotation, and cleaning procedures are provided in Appendix A.2.

To address the second challenge, we propose **CATSF** , a Capability-Aware Teacher–Student Framework for learning adaptive reasoning depth through two-stage fine-tuning for a student model. CATSF decomposes expert reasoning into basic and complex trajectories, distinguished by direct evidence assessment and explicit cross-modal correlation analysis, respectively. In the first stage, a compact Student model is fine-tuned on teacher-model-generated trajectories of both types to acquire classification knowledge and multi-depth reasoning capabilities. We then evaluate the Student model's classification correctness under basic reasoning on labeled training samples to construct a selection dataset: a sample is paired with a basic trajectory of a Teacher model if the Student model predicts its category correctly, or a complex trajectory of a Teacher model otherwise, provided that the corresponding Teacher model's trajectory is available and has a correct final label. In the second stage, the same Student model is further fine-tuned on these selection examples under an adaptive prompt. Thus, the Teacher model provides reasoning content, while the Student model's own observed performance guides the supervision for adaptive reasoning-path selection. At inference time, the resulting compact Student model selects between basic and complex reasoning under an adaptive prompt and produces the final classification without querying a Teacher model. This design supports efficient deployment while allowing reasoning depth to vary across feedback samples.

In summary, this work advances automated multimodal feedback classification via the contributions:

- **High-Quality Multimodal Dataset:** We introduce and release AUF, a high-quality expert-annotated multimodal Android User Feedback dataset containing 3,750 manually labeled entries, providing a foundational benchmark for multimodal feedback understanding and classification.

- **Adaptive Reasoning Framework:**We propose CATSF , a capability-aware teacher–student framework that first distills basic and complex reasoning, then fine-tunes the same student for adaptive path selection using its first-stage basic-reasoning correctness to guide supervision.
- **Classification Performance and Output Efficiency:** Experiments on AUF show that CATSF improves F_1 by up to 63.4% and accuracy by up to 18.3 percentage points over evaluated baselines. Across Teacher-Student configurations, it reduces mean output token usage by up to 50.8% relative to matched separately fine-tuned complex-reasoning baselines.

2 Related Work

2.1 Reasoning and Classification with LLMs

Recent research demonstrates the effectiveness of LLMs in classification tasks (Sun et al., 2023; Zhang et al., 2025; Li et al., 2022b; Longpre et al., 2023; Wang et al., 2022b; Mishra et al., 2022). A line of work employs structured CoT reasoning to support classification, including clue-guided reasoning in CARP (Sun et al., 2023), the four-stage framework in QLFR (Wu et al., 2024) and the three-hop reasoning process in THOR (Fei et al., 2023). These approaches organize intermediate reasoning to support the interpretation of linguistic cues and the prediction of class labels.

However, the benefits of explicit reasoning are not uniform across tasks and models. Chen et al. (2025) show that extended reasoning can generate redundant solutions that increase computational overhead with limited accuracy gains. In visual perception tasks, Li et al. (2026) find that reinforcement fine-tuning without explicit thinking can match or outperform its thinking-based counterpart. Similarly, Zhu et al. (2026) report that verbose textual reasoning can reduce accuracy in fine-grained visual classification. These findings caution against assuming that longer reasoning is universally beneficial. This motivates learning not only how to perform structured reasoning, but also when different reasoning processes are appropriate for the model and input at hand.

Motivation. Multimodal user feedback classification requires jointly interpreting textual descriptions and visual evidence. Our goal is to learn an appropriate choice between basic reasoning and more elaborate cross-modal analysis, rather than apply a single reasoning process uniformly. CATSF addresses this through two-stage fine-tuning: a compact Student model learns basic and complex reasoning from teacher-model-generated trajectories and is subsequently fine-tuned for adaptive path selection using supervision derived from its observed classification correctness under basic reasoning.

2.2 Multimodal Reasoning with VLMs and MLLMs

Early vision–language research established approaches to cross-modal alignment and visual–textual representation learning (Li et al., 2021; Radford et al., 2021; Li et al., 2022a; Wang et al., 2022a). BLIP-2 (Li et al., 2023) advances efficient vision–language transfer by connecting frozen image encoders and language models through a lightweight Q-Former. For classification, DORA (Hossain et al., 2024) combines dual co-attention with modality-specific features to identify hateful memes and their targets. Its explicit modeling of image–text interactions offers a complementary, feature-fusion-based approach to multimodal classification. Beyond feature integration, reasoning and contextual knowledge have been used to support multimodal decisions. Multimodal-CoT (Zhang et al., 2024) separates rationale generation from answer inference, allowing predictions to benefit from intermediate reasoning grounded in both modalities. WisdoM (Wang et al., 2024) augments multimodal sentiment analysis with contextual world knowledge generated by large vision–language models and uses contextual fusion to reduce irrelevant information. Although developed for different benchmarks, these methods address related challenges in integrating visual evidence, interpreting textual context, and producing informed predictions. Together with BLIP-2 and DORA, they provide complementary baselines covering vision–language transfer, cross-modal feature fusion, rationale-based inference, and knowledge-enhanced classification.

Recent open MLLMs provide an additional perspective on general-purpose multimodal capability. Qwen3-VL offers both Instruct and reasoning-enhanced Thinking variants (Bai et al., 2025), while Qwen3.5 further develops unified vision–language modeling (Qwen Team, 2026). We include their 8B–9B variants as supplementary references without AUF-specific fine-tuning, complementing the

task-adapted baselines and the text-only Qwen2.5-7B reference. This comparison places specialized feedback classification alongside general-purpose multimodal models of similar parameter scale.

Adaptive reasoning addresses a related question, i.e., which reasoning strategy a model should use. Thinkless (Fang et al., 2026) learns short- and long-form reasoning selection according to task complexity and model ability. TON (Wang et al., 2026) combines thought-dropout fine-tuning with reinforcement learning to select whether to generate explicit reasoning in vision–language tasks. TATA (Xu et al., 2026) instead uses model-aware data selection to learn adaptive switching between CoT and tool-integrated reasoning for mathematical problem solving.

Motivation. CATSF complements multimodal feature fusion and knowledge augmentation with student-specific supervision for reasoning-path selection. Unlike TON's think-or-not formulation, it selects between two explicit multimodal reasoning paths: basic evidence assessment and more elaborate text–image correlation analysis. Selection is learned through supervised fine-tuning rather than the reinforcement-learning optimization used in Thinkless and TON. Whereas TATA evaluates candidate CoT/TIR demonstrations through anchor-set performance, CATSF selects teacher targets according to the task-adapted student's classification correctness under basic reasoning on the corresponding training samples.

2.3 Knowledge Distillation

Knowledge distillation (KD) transfers knowledge from large teacher models to compact students, providing a practical route to efficient deployment (Hinton et al., 2015). Research on KD has explored uncertainty-aware formulations (Vadera et al., 2020; Fang et al., 2024) and logit-level alignment (Li et al., 2025). For language-model reasoning, Distilling Step-by-Step uses teacher-generated rationales as additional supervision for training smaller task-specific models (Hsieh et al., 2023). DeepSeek-R1 (Guo et al., 2025) further demonstrates the transfer of reasoning patterns to smaller dense models through fine-tuning on model-generated data.

Motivation. Learning from teacher rationales and selecting a reasoning path for a particular student are related but distinct learning goals. CATSF couples the distillation of basic and complex reasoning with a subsequent selection-training stage for the same student. While teacher trajectories remain the source of reasoning content, adaptive training targets are selected according to the task-adapted student's observed classification correctness under basic reasoning. Thus, path-selection supervision is grounded in student performance rather than inferred solely from teacher correctness.

3 Approach

3.1 Overview

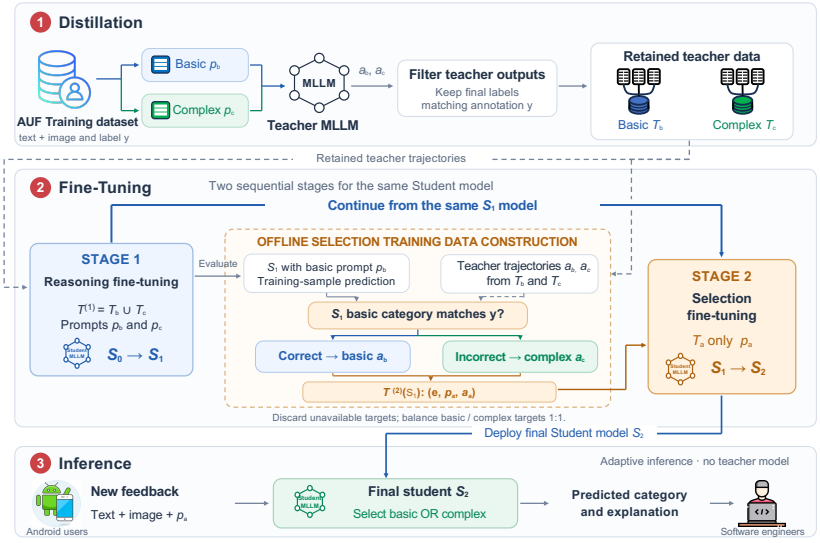


Figure 2: The overview of CATSF .

As shown in Figure 2, CATSF comprises three components: *Distillation*, *Fine-Tuning*, and *Inference*. In Distillation, a teacher model generates basic and complex reasoning trajectories using p_b and p_c. Outputs with correct final labels are retained to form T_b and T_c. In Fine-Tuning, the student model first learns classification and both reasoning paths from $T_b \cup T_c$ (Stage 1). The resulting student model S_1 is evaluated with p_b on labeled training samples, and its classification correctness guides the selection of teacher targets. Stage 2 continues fine-tuning the same student model on these targets under the adaptive prompt p_a. During Inference, the final student S_2 selects a basic or complex reasoning path under p_a and produces a category prediction without querying the teacher. All examples are drawn from the 3,750-entry AUF dataset; dataset construction is detailed in Appendix A.2.

3.2 Distillation

3.2.1 Structured Prompt Design for Coherent Reasoning

The Distillation component extracts task-specific reasoning trajectories from a Teacher MLLM, addressing the classification of Android feedback into *21 fine-grained categories* and the difficulty of resolving domain-specific ambiguities. We leverage the Teacher model's in-context learning by incorporating a "Category Description Section" into the basic and complex prompts, p_b and p_c, to provide structured domain expertise. These descriptions follow three principles: reflecting standard understanding, enhancing category distinctiveness, and explicitly reducing ambiguity. The same category descriptions are included in the adaptive prompt p_a, which is used for subsequent selection fine-tuning and inference.

Developed through iterative refinement by four annotators and a domain expert, the finalized descriptions significantly reduced "Other" misclassifications and improved human inter-annotator agreement. This structured design enables the MLLM to maintain explicit logical boundaries during its CoT process. For example, distinguishing "Security" from "Device Protection" is facilitated by defining a boundary based on whether harm has occurred. By embedding subject, temporal, and feature-specific attributes, we transform general semantic classification into structured, logic-guided reasoning. The finalized category descriptions are provided in Table 8, and the prompt templates are in Appendix A.4.

3.2.2 Multi-Level Logic Formalization for Knowledge Distillation

We categorize text-image relationships as Supportive, Consistent, Contradictory, or Complementary. These relations characterize multimodal evidence rather than determine a fixed reasoning path. Basic reasoning extracts key information from both modalities for classification, while complex reasoning additionally performs explicit cross-modal correlation analysis and evidence weighting. Formal definitions and examples are provided in Appendix A.3.

For each labeled training sample, the Teacher model generates basic and complex trajectories using p_b and p_c. We retain only trajectories whose final prediction matches the annotated label, forming T_b and T_c. Because the two paths are filtered independently, each sample may contribute to either dataset, both, or neither. The retained trajectories supervise multi-depth reasoning in Stage 1 and serve as candidate targets for Stage 2. Adaptive targets are selected according to the first-stage student's basic-reasoning correctness, as detailed in Section 3.3.

3.3 Knowledge Transfer via Parameter-Efficient Fine-Tuning

3.3.1 Model Architecture and Parameter Efficiency

We apply Low-Rank Adaptation (LoRA) (Hu et al., 2022) to the student's language model in both stages, optimizing only adapter parameters θ while freezing the vision encoder and multimodal projector. Stage 2 continues from the Stage 1 checkpoint.

3.3.2 Two-stage Reasoning and Selection Fine-Tuning

Both stages use causal language modeling to supervise the generation of a structured reasoning process followed by a final category prediction. For a training set T containing examples (e_j, p_j, a_j), where $e_j = (t_j, i_j)$ is the text–image input, p_j is the instruction prompt, and a_j is the target output

sequence, the supervised fine-tuning objective is

$$\mathcal{L}_{\text{SFT}}(\theta; T) = -\frac{1}{|T|} \sum_{(e_j, p_j, a_j) \in T} \sum_{k=1}^{|a_j|} \log P_\theta \left(w_{a_j}^k \mid t_j, V(i_j), p_j, w_{a_j}^{<k} \right). \tag{1}$$

Here, $V(i_j)$ denotes the visual representation produced by the frozen visual components, $w_{a_j}^k$ is the k-th target token, and $w_{a_j}^{<k}$ is the preceding target-token sequence. The pretrained model parameters are held fixed, and the loss is used to update the LoRA parameters θ.

In multi-depth reasoning fine-tuning (stage 1), we first train the Student model on the retained basic and complex training datasets from Section 3.2:

$$T^{(1)} = T_b \cup T_c, \qquad \theta_1 = \text{SFT}\Big(\theta_0, T^{(1)}\Big), \tag{2}$$

where SFT denotes supervised fine-tuning using Equation (1). Each example in T_b pairs the multimodal input and basic prompt p_b with a retained basic teacher output a_b; examples in T_c analogously use p_c and a_c. This stage trains classification and reasoning under both prescribed paths. The resulting Student model S_1 provides the performance signal used to construct the selection dataset. In student-performance-guided selection fine-tuning (Stage 2), we construct the selection training dataset by running the fixed first-stage student S_1 with the basic prompt p_b on labeled training samples. Let $\hat{y}_{b,j}^{S_1}$ denote its predicted category for input e_j, and let y_j be the annotated category. The retained teacher outputs $a_{b,j}$ and $a_{c,j}$ provide candidate targets; an unavailable output is denoted by $\varnothing$. The adaptive target is selected as

$$a_{a,j} = \begin{cases} a_{b,j}, & \hat{y}_{b,j}^{S_1} = y_j \ \wedge \ a_{b,j} \neq \varnothing, \\ a_{c,j}, & \hat{y}_{b,j}^{S_1} \neq y_j \ \wedge \ a_{c,j} \neq \varnothing, \\ \varnothing, & \text{otherwise.} \end{cases} \tag{3}$$

Thus, a correct basic prediction selects the basic teacher output, whereas an incorrect basic prediction selects the complex teacher output. A sample is excluded if the required teacher output was not retained during the filtering in Section 3.2. The Student model's prediction determines which path is selected, but the retained teacher output supplies the target reasoning and final category.

Each retained selection example is formed as $(e_j, p_a, a_{a,j})$, using the adaptive prompt p_a rather than the path-specific prompts p_b or p_c. We balance the retained examples with basic and complex targets at a 1:1 ratio to obtain $T^{(2)}$, emphasizing that this dataset is constructed from the performance of the first-stage Student model. The ratio controls the composition of the selection training examples; it does not impose a fixed distribution of reasoning paths at inference. Starting from θ_1, we continue fine-tuning the same Student model using only $T^{(2)}(S_1)$:

$$\theta_2 = \text{SFT}\Big(\theta_1, T^{(2)}(S_1)\Big). \tag{4}$$

The selected teacher outputs retain their basic or complex reasoning formats, providing supervision for path selection under the adaptive prompt. Equation (1) is applied to the complete target sequence, including the reasoning and final category. At inference, the resulting Student model S_2 receives a new text–image input with p_a and generates the selected reasoning path and category without access to the annotated label.

4 Empirical Evaluation

We evaluate CATSF by addressing the following Research Questions:

RQ$_1$ (SOTA Comparison): Does CATSF consistently outperform state-of-the-art baselines (e.g., Multimodal-CoT, WisdoM, DORA) in the specialized task of Android user feedback classification?

RQ$_2$ (Distillation Effectiveness): How does CATSF 's classification performance compare between student and teacher MLLMs, and what factors influence the performance gap?

RQ$_3$ (Ablation Study): What are the individual contributions of basic, complex, and adaptive reasoning capabilities to the overall performance of student models?

4.1 Experiment Setup

4.1.1 Models and Baselines

As shown in Table 1, we adopt two proprietary state-of-the-art multimodal large language models (GPT-4o Mini and Gemini 2.5 Flash) as teachers and two open-source models (Qwen2.5-VL-7B-Instruct and InternVL3-8B-hf) as students. We further compare CATSF with fine-tuned state-of-the-art models, including text-centric (Qwen2.5-7B), a vision-language backbone (BLIP-2) and reasoning-enhanced baselines (Multimodal-CoT, WisdoM, and DORA). In addition, we include comparisons against strong vanilla MLLMs with parameter sizes comparable to the student models, namely Qwen3-VL-8B-Instruct, Qwen3-VL-8B-Thinking, and Qwen3.5-9B.

Table 1: Summary of Models and Baselines.

Roles	Models	Comments
Teacher	GPT-4o Mini / Gemini 2.5 Flash	gpt-4o-mini-2024-07-18 / gemini-2.5-flash-2025-06
Student	Qwen2.5-VL-7B-Instruct/InternVL3-8B-hf	Reasoning-based Distillation
Baselines	Qwen2.5-7B (Yang et al., 2025)	Text-only LLM Baseline
	BLIP-2 (Li et al., 2023)	Q-Former Backbone
	M-CoT (Zhang et al., 2024)	Rationale Generation and Answer Inference
	WisdoM (Wang et al., 2024)	World Knowledge Integration
	DORA (Hossain et al., 2024)	Dual Co-attention for Multimodal Classification
Vanilla	Qwen3-VL-8B-Instruct (Bai et al., 2025)	Multimodal Instruction Following
	Qwen3-VL-8B-Thinking (Bai et al., 2025)	Reasoning-enhanced Multimodal Reasoning
	Qwen3.5-9B (Qwen Team, 2026)	Unified Vision-Language Model

We evaluate the performance via Accuracy (A), Precision (P), Recall (R), and Macro-F1 (F_1). To quantify reasoning distillation quality, we employ BLEU-4 (B4), ROUGE-L (RL), and BERTScore (BS) to measure the semantic alignment between Teacher and Student reasoning trajectories.

4.1.2 Dataset and Implementation

For CATSF , GPT-4o Mini and Gemini 2.5 Flash provide 2,899/2,738 Stage-1 samples and 406/274 and 292/576 Stage-2 samples for Qwen/InternVL, respectively. All models are evaluated on the same 400-sample test set. Qwen-2.5, Multimodal-CoT, WisdoM, DORA, and BLIP-2 follow their original implementations and are fine-tuned on AUF (Yang et al., 2025; Zhang et al., 2024; Wang et al., 2024; Hossain et al., 2024; Li et al., 2022a). Main tables report a single run; repeated-inference statistics are provided in Appendix A.6. Student models are trained with Adam (1×10^{-4}) on 2 A100 GPUs.

4.2 Comparison with Baselines (RQ_1)

Table 2 compares Student models against state-of-the-art baselines including text-only LLMs (Qwen2.5-7B), VLMs (BLIP-2, DORA), and MLLMs (WisdoM, Multimodal-CoT). Qwen-2.5-VL (GPT) outperformed the text-only Qwen2.5-7B by 9.7 percentages points in accuracy and 29.1% in F_1, emphasizing the necessity of multimodal integration and CoT reasoning. BLIP-2 and DORA significantly underperformed, trailing Qwen-VL(GPT) by 17.7/18.3 percentage points (accuracy) and 43.1%/63.4% (F_1). Despite their multimodal design, their limited text comprehension and unsuitability for complementary text-image relationships caused them to trail even the text-only baseline.

Furthermore, Qwen-VL outperformed Multimodal-CoT by 15.7 percentage points in accuracy and 45.4% F_1, as the latter's T5 architecture lacks sufficient comprehension and truncates long category descriptions.WisdoM achieved the closest performance, trailing by 3.5 percentage points in accuracy and 9.8% in F_1. Although WisdoM utilizes a 13B model for supplementary reasoning and complex inference fusion, our 7B student model achieves superior results with significantly lower inference complexity, demonstrating the efficiency and effectiveness of the CATSF framework.

4.3 Distillation Effectiveness (RQ_2)

Table 3 shows that CATSF substantially improves the original student models. Taking each backbone model's best original score for each metric as the reference, adaptive students gain 10.0–16.5

Table 3: Performance of Teacher and Student Models on AUF dataset.

Roles	Models	Metrics			
		A(%)↑	**P**(%)↑	**R**(%)↑	F_1↑
Original	Qwen-VL$_{basic}$	69.5	45.2	48.3	44.0
	Qwen-VL$_{comp}$	67.8	51.3	48.8	46.8
	Qwen-VL$_{adap}$	69.5	41.9	47.3	43.1
	InternVL$_{basic}$	71.5	53.6	56.3	52.6
	InternVL$_{comp}$	71.3	61.2	61.7	58.1
	InternVL$_{adap}$	72.8	59.2	62.7	58.5
Teacher	GPT$_{basic}$	82.5	78.4	75.0	75.8
	GPT$_{comp}$	**85.3**	**79.8**	76.8	77.3
	GPT$_{adap}$	83.3	79.5	75.2	75.9
Student	Qwen-VL(ft)$_{basic}$	**83.8**	**80.6**	**78.9**	**78.9**
	Qwen-VL(ft)$_{comp}$	83.5	79.2	77.8	77.2
	Qwen-VL(ft)$_{adap}$	**86.0**	**81.4**	**81.5**	**80.4**
	InternVL(ft)$_{basic}$	81.3	75.0	73.9	72.6
	InternVL(ft)$_{comp}$	82.0	77.2	77.2	75.4
	InternVL(ft)$_{adap}$	85.0	81.7	**79.4**	**78.8**
Teacher	Gemini$_{basic}$	78.8	73.2	69.1	69.5
	Gemini$_{comp}$	81.5	**81.9**	**76.0**	**77.4**
	Gemini$_{adap}$	79.3	68.4	64.1	64.9
Student	Qwen-VL(ft)$_{basic}$	81.8	79.7	74.4	75.3
	Qwen-VL(ft)$_{comp}$	**82.0**	**82.5**	74.0	75.1
	Qwen-VL(ft)$_{adap}$	**83.0**	77.8	**74.6**	**75.1**
	InternVL(ft)$_{basic}$	80.5	78.0	73.3	73.4
	InternVL(ft)$_{comp}$	81.5	77.8	74.8	74.4
	InternVL(ft)$_{adap}$	**82.8**	**76.9**	**74.4**	**73.9**

Table 2: Performance comparison of our best-performing model against state-of-the-art baselines for classification tasks.

Baselines	Metrics			
	A(%)↑	**P**(%)↑	**R**(%)↑	$\mathbf{F_1}$↑
Qwen2.5-7B	**76.3**	**66.0**	**64.9**	**62.3**
BLIP-2	68.3	59.1	60.1	56.2
DORA	67.7	52.3	49.4	49.2
Multimodal-CoT	70.3	61.6	53.8	55.3
WisdoM	**82.5**	**75.8**	**72.7**	**73.2**
Qwen-2.5-VL(GPT)	**86.0**	**81.4**	**81.5**	**80.4**

Table 4: The results of vanilla MLLMs with different reasoning strategies.

Models	Metrics			
	A(%)↑	**P**(%)↑	**R**(%)↑	F_1↑
Qwen3-VL-8B$_{basic}$	**70.8**	**72.1**	65.9	**64.7**
Qwen3-VL-8B$_{comp}$	**70.8**	65.4	**67.0**	64.0
Qwen3-VL-8B$_{adap}$	68.8	71.1	64.6	63.8
Qwen3-VL-8B-Thinking$_{basic}$	76.5	**79.8**	74.0	**74.2**
Qwen3-VL-8B-Thinking$_{comp}$	76.5	72.6	70.7	70.0
Qwen3-VL-8B-Thinking$_{adap}$	**78.0**	79.8	**75.1**	73.6
Qwen3.5-9B$_{basic}$	75.5	**72.0**	**65.2**	**66.3**
Qwen3.5-9B$_{comp}$	77.0	64.9	64.6	63.9
Qwen3.5-9B$_{adap}$	**78.0**	66.3	61.8	61.8
Qwen2.5-VL(ft)$_{adap}$	**86.0**	**81.4**	**81.5**	**80.4**

Note: GPT: GPT-4o Mini; Gemini: Gemini 2.5 Flash. Model variants: *basic* (basic reasoning), *comp* (complex reasoning), *adap* (adaptive reasoning), *ft* (fine-tuned). Qwen-VL(GPT) indicates Qwen-VL-7B-Instruct fine-tuned via GPT-4o Mini dataset. **Metrics:** A (Accuracy); P (Precision); R (Recall); F_1 (Macro F_1 measurement). **Bold text** indicates the top three models (or the best result in specific model groups; Total running time for Table 3/ Table 2/Table 4 is 35/52/10 hours).

percentage points in accuracy and 26.3%–71.8% in F_1. GPT-supervised Qwen-VL achieves 86.0% accuracy and 80.4 F_1, exceeding GPT's best reported scores by 0.7 percentage points and 4.0%, respectively. Gemini-supervised student models also surpass their teacher models' best accuracy, but retain lower F_1 scores than its complex-reasoning variant. Thus, CATSF transfers useful task-specific supervision, while the remaining student–teacher gap depends on the metric and teacher source.

Complex reasoning yields the highest accuracy for both teacher models, whereas adaptive variants achieve the highest student accuracy in all four configurations, exceeding the best fixed-path variant by 1.0–3.0 percentage points. Adaptive reasoning also achieves the highest F_1 for GPT-supervised students. Under Gemini supervision, however, Qwen-VL's basic variant scores slightly higher in F_1 than its adaptive variant (75.3 versus 75.1), as does InternVL's complex variant (74.4 versus 73.9). The accuracy gains are therefore more consistent than the F_1 gains, indicating that the benefit of adaptive reasoning depends on both the supervision source and evaluation criterion.

Table 4 shows that GPT-supervised Qwen2.5-VL outperforms comparable vanilla MLLMs by 8.0 percentage points in accuracy and 8.4% in F_1, highlighting the benefit of AUF-specific adaptation over general-purpose multimodal capability. Table 5 shows consistent gains in BLEU-4, ROUGE-L, and BERTScore after fine-tuning, indicating stronger alignment with teacher trajectories. These metrics measure output similarity rather than reasoning or path-selection correctness; the latter, together with token usage, is analyzed in Appendix A.5.

4.4 Ablation Study (RQ_3)

Table 6 evaluates CATSF's complete two-stage procedure (**Full ft**) against two ablated variants. **B&C w/o** omits first-stage basic/complex reasoning fine-tuning. Instead, the vanilla student's basic-reasoning predictions are used to construct its selection dataset according to Equation 3, and the student is fine-tuned directly on the resulting selection examples. Thus, this variant preserves the selection rule but derives its supervision from the vanilla rather than the task-adapted student. **Adap w/o** retains the first-stage student without subsequent selection fine-tuning.

Table 5: Assessment of output similarity between teacher and student models. (BLEU: BLEU-4; ROUGE: ROUGE-L; BERTS: BERTScore).

Student	Teacher: GPT			Teacher: Gemini		
	BLEU↑	ROUGE↑	BERTS↑	BLEU↑	ROUGE↑	BERTS↑
Qwen-VL(orig)$_{basic}$	22.7	43.4	92.0	16.6	32.0	88.3
Qwen-VL(orig)$_{comp}$	**30.2**	45.2	92.5	**24.4**	32.8	88.6
Qwen-VL(orig)$_{adap}$	9.5	37.7	88.8	17.4	22.4	86.5
Qwen-VL(ft)$_{basic}$	27.7	**50.0**	**93.5**	22.6	**40.3**	90.1
Qwen-VL(ft)$_{comp}$	**31.7**	**48.9**	**93.4**	**27.6**	**40.9**	**90.8**
Qwen-VL(ft)$_{adap}$	29.3	43.6	92.1	9.2	26.7	87.4
InternVL(orig)$_{basic}$	20.3	42.7	92.1	15.6	32.2	88.5
InternVL(orig)$_{comp}$	19.8	41.3	91.4	23.0	31.1	88.6
InternVL(orig)$_{adap}$	9.6	37.6	88.7	16.2	22.2	86.6
InternVL(ft)$_{basic}$	25.6	**48.8**	**93.4**	21.8	**40.2**	**90.2**
InternVL(ft)$_{comp}$	**29.7**	47.4	93.2	**26.5**	40.1	**90.7**
InternVL(ft)$_{adap}$	29.5	43.4	92.1	8.0	25.8	87.3

Full ft achieves higher accuracy and F_1 than **B&C w/o** in all four teacher–student configurations. The improvements for GPT-supervised Qwen-VL reach 8.7 percentage points in accuracy and 14.5% in F_1. The corresponding gains for InternVL are 6.5 percentage points and 11.0%. Under Gemini supervision, the gains are 7.7 percentage points and 37.6% for Qwen-VL, and 3.5 percentage points and 9.5% for InternVL. These results favor adapting the student to both reasoning paths before constructing and learning from selection supervision. Because this ablation also changes the student whose predictions determine the selection targets, the observed gains reflect the contribution of first-stage adaptation to the overall pipeline, including its influence on subsequent supervision.

Compared with **Adap w/o, Full ft** improves accuracy/F_1 by 3.5 percentage points/10.4% for GPT-supervised Qwen-VL and 1.7/5.8% for InternVL; under Gemini supervision, the gains increase to 5.5/17.0% and 4.5/11.6%, respectively. These results confirm the added value of selection fine-tuning. Qwen-VL benefits more from this stage under both teachers, reversing the pre-selection ranking relative to InternVL, although the final Gemini accuracy gap is only 0.2 percentage points.

Table 6: Ablation of CATSF's two-stage fine-tuning. ("B&C w/o" and "Adap w/o" omit reasoning and selection fine-tuning, respectively; "Full ft" includes both stages.)

Teacher	GPT-4o Mini							
Students	Qwen-VL				InternVL			
Metrics	**A**(%)↑	**P**(%)↑	**R**(%)↑	**F$_1$**↑	**A**(%)↑	**P**(%)↑	**R**(%)↑	**F$_1$**↑
B&C w/o	77.3	72.0	71.8	70.2	78.5	74.1	72.4	71.0
Adap w/o	82.5	77.8	72.6	72.8	83.3	78.7	74.0	74.5
Full ft	86.0	81.4	81.5	80.4	85.0	81.7	79.4	78.8
Teacher	Gemini 2.5 Flash							
Student	Qwen-VL				InternVL			
Metrics	**A**(%)↑	**P**(%)↑	**R**(%)↑	**F$_1$**↑	**A**(%)↑	**P**(%)↑	**R**(%)↑	**F$_1$**↑
B&C w/o	75.3	59.5	54.5	54.6	79.3	67.4	69.0	67.5
Adap w/o	77.5	65.8	65.0	64.2	78.3	68.4	66.9	66.2
Full ft	83.0	77.8	74.6	75.1	82.8	76.9	74.4	73.9

5 CONCLUSIONS

In this study, we present AUF, the first high-quality, manually annotated multimodal Android user feedback dataset, and CATSF , a cost-effective teacher–student framework for adaptive multimodal classification. By distilling basic and complex reasoning into compact models, CATSF enables dynamic reasoning selection and achieves performance comparable to or exceeding Teacher MLLMs at significantly lower cost. Extensive experiments validate the effectiveness of adaptive reasoning for accurate user feedback classification.

AI USE STATEMENT

Generative AI models were used to produce teacher reasoning trajectories for distillation, as described in Section 3.2. Generative AI tools also assisted with translation, language editing, and preparation of the manuscript's LaTeX source. The authors are responsible for the final content of the paper, including all claims and artifacts produced with AI assistance.

ETHICS STATEMENT

This study uses publicly accessible posts and screenshots from the Android Help Community for technical-issue classification. Direct personal identifiers have been removed or redacted from the AUF dataset and the examples presented in this paper. The analysis concerns technical issues rather than identifying or profiling users. Nevertheless, retained content may allow the original public posts to be located.

Category labels were assigned by four annotators with Android development experience, with disagreements reviewed by a domain expert. The relevant institutional office determined that formal ethics review was not required for this study under the applicable institutional policy. Teacher-generated reasoning trajectories are distinguished from human category annotations and should not be treated as verified diagnoses.

CATSF is intended to assist technical-support triage. Incorrect classifications or unsupported explanations may misdirect support requests; deployment should therefore retain human oversight and include validation in the intended setting.

REPRODUCIBILITY STATEMENT

Our anonymous repository for the AUF dataset, student model artifacts, and code is `https://anonymous.4open.science/r/Anonymization_a1b2-0x3F`. Sections 3.2–3.3 describe teacher-trajectory filtering, selection-data construction, and two-stage fine-tuning. Table 1 and Section 4.1 identify the evaluated models and summarize the experimental setup. Appendix A.2 documents data collection and annotation, and Appendix A.4 provides the prompts. Appendix A.5 defines the routing diagnostics and output-token measurement, while Appendix A.6 reports three repeated inference evaluations per trained configuration. These repetitions assess inference variability rather than variability across independently trained models.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Android Help. Android help. https://support.google.com/android/.

AndroidStatistics(2026). Androidstatistics(2026). URL `https://www.businessofapps.com/data/android-statistics/`.

Anonymous. Android multimodal user feedback dataset. `https://anonymous.4open.science/r/r8f3k29`, 2026.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, et al. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Xingyu Chen, Jiahao Xu, Tian Liang, Zhiwei He, Jianhui Pang, Dian Yu, Linfeng Song, Qiuzhi Liu, Mengfei Zhou, Zhuosheng Zhang, et al. Do not think that much for 2+ 3=? on the overthinking of long reasoning models. In *Forty-second International Conference on Machine Learning*, 2025.

Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

Gongfan Fang, Xinyin Ma, and Xinchao Wang. Thinkless: Llm learns when to think. *Advances in neural information processing systems*, 38:151268–151295, 2026.

Luyang Fang, Yongkai Chen, Wenxuan Zhong, and Ping Ma. Bayesian knowledge distillation: a bayesian perspective of distillation with uncertainty quantification. In *Forty-first international conference on machine learning*, 2024.

Hao Fei, Bobo Li, Qian Liu, Lidong Bing, Fei Li, and Tat-Seng Chua. Reasoning implicit sentiment with chain-of-thought prompting. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 1171–1182, 2023.

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

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Eftekhar Hossain, Omar Sharif, Mohammed Moshiul Hoque, and Sarah Masud Preum. Deciphering hate: Identifying hateful memes and their targets. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 8347–8359, 2024.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alex Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. In *Findings of the Association for Computational Linguistics: ACL 2023*, pp. 8003–8017, 2023.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

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

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705, 2021.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, pp. 12888–12900. PMLR, 2022a.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pp. 19730–19742. PMLR, 2023.

Minchong Li, Feng Zhou, and Xiaohui Song. Bild: Bi-directional logits difference loss for large language model distillation. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 1168–1182, 2025.

Ming Li, Jike Zhong, Shitian Zhao, Yuxiang Lai, Haoquan Zhang, Wang Bill Zhu, and Kaipeng Zhang. To think or not to think: A study of thinking in rule-based visual reinforcement fine-tuning. *Advances in Neural Information Processing Systems*, 38:167356–167407, 2026.

Shiyang Li, Jianshu Chen, Yelong Shen, Zhiyu Chen, Xinlu Zhang, Zekun Li, Hong Wang, Jing Qian, Baolin Peng, Yi Mao, et al. Explanations from large language models make small reasoners better. *arXiv preprint arXiv:2210.06726*, 2022b.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. The flan collection: Designing data and methods for effective instruction tuning. In *International Conference on Machine Learning*, pp. 22631–22648. PMLR, 2023.

Jianguo Lu, Yan Wang, Jie Liang, Jessica Chen, and Jiming Liu. An approach to deep web crawling by sampling. In *2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, volume 1, pp. 718–724. IEEE, 2008.

Jayant Madhavan, David Ko, Łucja Kot, Vignesh Ganapathy, Alex Rasmussen, and Alon Halevy. Google's deep web crawl. *Proceedings of the VLDB Endowment*, 1(2):1241–1252, 2008.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3470–3487, 2022.

Debjyoti Mondal, Suraj Modi, Subhadarshi Panda, Rituraj Singh, and Godawari Sudhakar Rao. Kam-cot: Knowledge augmented multimodal chain-of-thoughts reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 18798–18806, 2024.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Xiaofei Sun, Xiaoya Li, Jiwei Li, Fei Wu, Shangwei Guo, Tianwei Zhang, and Guoyin Wang. Text classification via large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 8990–9005, 2023.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*, 2024.

Meet Vadera, Brian Jalaian, and Benjamin Marlin. Generalized bayesian posterior expectation distillation for deep neural networks. In *Conference on Uncertainty in Artificial Intelligence*, pp. 719–728. PMLR, 2020.

Jiaqi Wang, Kevin Qinghong Lin, James Cheng, and Mike Zheng Shou. [think or not? selective reasoning via reinforcement learning for vision-language models]. *Advances in Neural Information Processing Systems*, 38:111196–111234, 2026.

Peng Wang, An Yang, Rui Men, Junyang Lin, Shuai Bai, Zhikang Li, Jianxin Ma, Chang Zhou, Jingren Zhou, and Hongxia Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework. In *International conference on machine learning*, pp. 23318–23340. PMLR, 2022a.

Wenbin Wang, Liang Ding, Li Shen, Yong Luo, Han Hu, and Dacheng Tao. Wisdom: Improving multimodal sentiment analysis by fusing contextual world knowledge. In *Proceedings of the 32nd ACM international conference on multimedia*, pp. 2282–2291, 2024.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. *arXiv preprint arXiv:2204.07705*, 2022b.

Hui Wu, Yuanben Zhang, Zhonghe Han, Yingyan Hou, Lei Wang, Siye Liu, Qihang Gong, and Yunping Ge. Quartet logic: A four-step reasoning (qlfr) framework for advancing short text classification. *arXiv preprint arXiv:2401.03158*, 2024.

Xi Wu, Subodha Kumar, and Min-Seok Pang. Tackling android fragmentation: Mobile apps' dilemma and the platform's strategies. 2020.

Xin Xu, Yan Xu, Tianhao Chen, Yuchen Yan, Chengwu Liu, Zaoyu Chen, Yufei Wang, Yichun Yin, Yasheng Wang, Qun Liu, et al. Teaching llms according to their aptitude: Adaptive switching between cot and tir for mathematical problem solving. In *The Third Conference on Parsimony and Learning (Proceedings Track)*, 2026.

An Yang, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoyan Huang, Jiandong Jiang, Jianhong Tu, Jianwei Zhang, Jingren Zhou, et al. Qwen2. 5-1m technical report. *arXiv preprint arXiv:2501.15383*, 2025.

Yazhou Zhang, Mengyao Wang, Qiuchi Li, Prayag Tiwari, and Jing Qin. Pushing the limit of llm capacity for text classification. In *Companion Proceedings of the ACM on Web Conference 2025*, pp. 1524–1528, 2025.

Zhuosheng Zhang, Aston Zhang, Mu Li, Hai Zhao, George Karypis, and Alex Smola. Multimodal chain-of-thought reasoning in language models. *Transactions on Machine Learning Research*, 2024.

Jie Zhu, Yiyang Su, and Xiaoming Liu. Can textual reasoning improve the performance of mllms on fine-grained visual classification? *arXiv preprint arXiv:2601.06993*, 2026.

A APPENDIX

A.1 LIMITATIONS

While our work demonstrates the effectiveness of adaptive reasoning for multimodal user feedback classification, several limitations remain.

Dependency Limitation: The framework's performance is inherently dependent on the quality of the teacher models used for knowledge distillation, as their reasoning trajectories can be suboptimal or exhibit hallucinations. Although our proposed adaptive selection mechanism partially mitigates this, enhancing the robustness and reliability of teacher reasoning presents a key direction for future research.

Modality Limitation: Our framework currently uses static screenshots as visual input. Temporal visual information, such as screenshot sequences or screen recordings, may provide additional context for understanding user feedback. Future work will explore incorporating these inputs to support more fine-grained multimodal reasoning.

A.2 DATASET CONSTRUCTION

In this section, we elaborate on the systematic methodology employed to construct our high-quality, multimodal Android User Feedback (AUF) dataset. We first detail the procedures for *Data Acquisition and Preprocessing* (Section A.2.1), including the collection strategy and initial data cleaning rules. Following this, we describe the rigorous *Data Classification and Labeling* (Section A.2.2), outlining the steps taken by domain experts to ensure label accuracy and inter-annotator agreement. We then present the *Dataset Statistics and Misclassification Analysis* (Section A.2.3), which confirms the widespread issue of user mislabeling and demonstrates the representativeness of our final corpus. The AUF dataset is available at an anonymous repository (Anonymous, 2026).

A.2.1 DATA ACQUISITION AND PREPROCESSING

We collect Android user feedback from the Google Help website for Android (Android Help), which serves as Google's official technical support platform for Android systems, addressing technical issues users encounter with Android devices or applications. Users submit feedback through a structured form that is posted to the website to receive professional assistance. The form captures an issue title, issue description (which may include multimodal content), and category (group) selection. Given that posted user feedback is accessible to other users only through the website's search interface, we employed a deep web crawler (Lu et al., 2008; Madhavan et al., 2008) (deepCrawler) to systematically collect user feedback data.

The collection algorithm operates as follows: Algorithm 1 begins by randomly selecting a token from the main page to serve as the initial query q_0, which is then submitted to the search interface I_s. The interface returns a set of user feedback entries F. Each user feedback entry is converted to $MD5$ encoding and cross-referenced with previously collected feedback; any duplicates are removed. For each new user feedback entry in F, we identify the top 5 tokens with the highest term frequency-inverse document frequency (tf-idf) values and add them to the query pool Q. A query is then randomly selected from Q to serve as the subsequent query q_1, which is submitted to I_s to retrieve additional user feedback entries. The collection process concludes when the returned set F contains no more new user feedback entries. This Algorithm 1 ensures broader data coverage while collecting more feedback entries.

We collected a total of 4,251 user feedback entries spanning from April 2019 to September 2025. During data collection, we retained entries that simultaneously met the following criteria: (i) inclusion of both image(s) and textual information; (ii) issue titles and descriptions exceeding 3 tokens in length; (iii) no duplication with other entries; (iv) content relevance to Android-related topics. Criteria (i) and (ii) were automatically validated through programmatic verification. For criterion (iii), we generated unique MD5 identifiers for each entry, identifying duplicates through identifier matching. For criterion (iv), we conducted a manual review process to evaluate each entry on an individual basis.

Algorithm 1 Data Collection Algorithm

INPUT: An initial query q_0
OUTPUT: A user-feedback dataset D
PROCEDURE:

Initialize $D = \{\}$ and a query pool $Q = \{\}$
$F = I_s(q_0)$
while $|F \cap D| < |F|$ **do**
  for each entry $f \in F \setminus D$ **do**
    add(f, D) #add new entries
    $T_5 = tfidf(f)$ #get top 5 tokens
    $Q = Q \cup T_5$ #add new queries
  end for
  $F = I_s(Random(Q))$ #get random query
end while
return D

A.2.2 Data Classification and Labeling

Classification Categories. The collected user feedback entries use the 9 groups from Table 8 as classification labels. In practice, using groups as labels simplifies the selection process for users—they only need to review 9 groups instead of 21 categories. Additionally, categories within each group are related (e.g., Wi-Fi Connectivity and Bluetooth both involve network connectivity concepts), helping users locate their issues more easily. However, for our classification task, using 21 categories as labels for each entry is significantly more effective than using 9 groups. The main reason is that while categories within a group share some similarities, the root causes of issues across different categories can vary greatly (e.g., Wi-Fi Connectivity issues may stem from router problems, while Bluetooth connectivity issues may result from device pairing difficulties). In this study, we aim for MLLMs to identify the root cause of each entry through reasoning to achieve accurate classification. Fine-grained categories align more naturally with the root causes of individual entries.

Annotation Process. To complete the annotation process, we recruited four annotators, all of whom were graduate students with experience in Android application development. Additionally, to resolve any disagreements among annotators, we engaged a programmer with extensive experience in software engineering and natural language processing to serve as an annotation expert. We divided the annotators into two groups of two, with each group annotating different subsets of user feedback cases. During the initial phase, we provided training on the definitions of 21 problem categories (shown in Table 8), distinguishing features between similar categories, and relevant examples. Our primary objective was to ensure that the annotators understood the training content and could accurately identify the root causes of user feedback issues from multimodal inputs. Two annotators independently labeled each case, with the final label determined through consensus. On average, an annotator required three minutes to determine the label for a user feedback case. In cases of disagreement, the expert joined the discussion to make the final decision. For a small number of cases ($< 3\%$), we found that user feedback entries could relate to multiple categories. When annotators and the expert could not reach consensus on label assignment, the category was determined by vote. Ultimately, we obtained a multimodal user feedback dataset containing approximately 3,750 cases with problem categories.

We computed the inter-annotator agreement in terms of Cohen's Kappa (k) score (Cohen, 1960) to check the validity, i.e., we compute the category-specific k scores and then take their mean across the 21 categories. Table 7 presents the Kappa scores for each category. The arithmetic mean of the 21 category-specific κ scores is 0.7253. Most categories achieved k scores exceeding 0.6, with Camera (0.9338), Battery (0.9170), and Bluetooth (0.9033) demonstrating particularly high consistency. While Other, Device Protection, and Accessibility represent the three categories with the lowest k-scores, all remain above 0.5. The top three categories exhibit distinct textual characteristics that minimize confusion with other categories. In contrast, the latter three categories encompass broader content, with textual expressions that may overlap with terminology commonly used in other categories. For instance, "unable to enable screen reader function" functionally belongs to Accessibility (as it aims to enhance device accessibility), yet from a textual standpoint, it also relates to system permission management and could reasonably be interpreted as a Permissions issue. We

Table 7: Kappa Coefficients for User Feedback Categories

Category	k-score	Category	k-score
GPS	1.0000	Malware	0.6670
Camera	0.9338	Forgot PIN/Pattern/Passcode	0.6505
Battery	0.9170	Privacy & Permissions	0.6132
Bluetooth	0.9033	Performance	0.6093
Location	0.8993	Google Play Services	0.6079
Backup & Restore	0.8328	Stability	0.5863
Notifications	0.8275	Syncing	0.5549
Find My Device	0.8225	Other	0.5379
Wi-Fi Connectivity	0.7888	Device Protection	0.5248
Settings	0.7365	Accessibility	0.5156
Security	0.7020	—	
Mean across categories: 0.7253			

note that all cases of annotator disagreement underwent expert review, with final labels established through majority voting.

A.2.3 Dataset Statistics and Misclassification

After the comprehensive expert annotation process, our final dataset comprises 3,750 high-quality, manually labeled user feedback entries. Our analysis confirms that the issue of user misclassification is rampant, validating the urgent need for this expert-annotated dataset. The official Google Android Community has categorized user feedback into 21 categories, which are further organized into 9 major groups (shown in Table 8).

The phenomenon of mislabeling is pervasive rather than an anomaly. The example presented in Figure 1 illustrates this common pattern: the user mistakenly categorized an alarm malfunction under the 'Other' group, yet expert analysis of the accompanying image revealed the issue stemmed from a depleted battery, warranting classification under the 'Battery/Performance/Stability' group. This misclassification appears to have resulted from the user's limited familiarity with battery-related functionality.

Figure 3 presents the distribution of incorrect label counts for each group based on manual annotation of a subset of 3,750 samples. All groups exhibited labeling errors. Specifically, 1,200 samples examined were assigned to incorrect groups, representing a widespread 32.0% error rate in user-submitted labels. We observed that users made the fewest mistakes in the "Find My Device/GPS/Location" group due to its clear characteristics, while the "Other" group exhibited the highest error rate as it frequently became the default selection for uncertain users. This widespread mislabeling underscores the significance of the first challenge and highlights the necessity of our expert-corrected dataset.

To compare data distributions before and after expert annotation, the subfigures (A) and (B) of Figure 4 present the distribution of entries across different groups. Analysis of these subfigures indicates that the overall data distribution demonstrated consistency, with an average variation of 2.3 percentage points across the 9 groups, suggesting our dataset is representative. The "Battery/Performance/Stability" and "Other" groups exhibited the largest and smallest variations, respectively. The former showed greater variation due to having the highest number of entries among all groups, while the latter demonstrated minimal variation because the volume of entries migrating from "Other" to other groups approximately equaled those migrating into "Other"—representing a balanced exchange. Furthermore, Figure 5 also illustrates the distribution of data across all 21 fine-grained categories, which serves as our final classification labels.

A.3 Formal Definitions of Multimodal Interdependencies

To rigorously define the interaction between textual and visual evidence, we represent a user feedback entry as a set of propositions derived from text t and image(s) i. Let $\{p_1^t, \ldots, p_n^t\}$ be the n propositions from the text and $\{p_1^i, \ldots, p_m^i\}$ be the m propositions from the image. The underlying issue or root

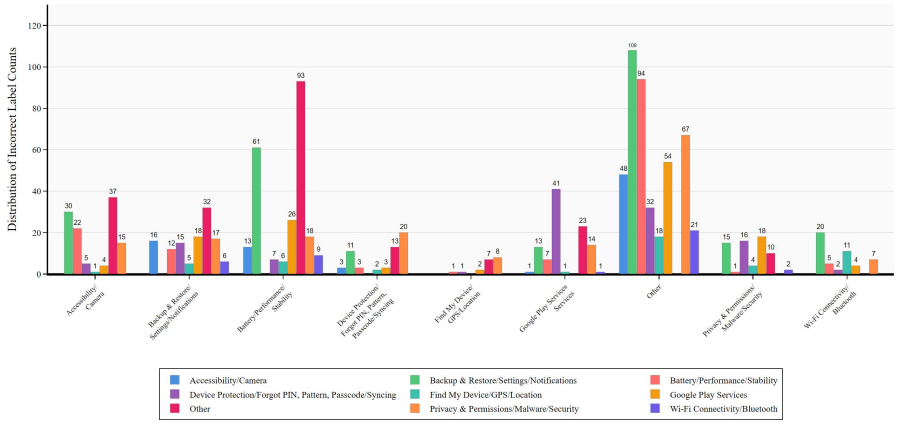
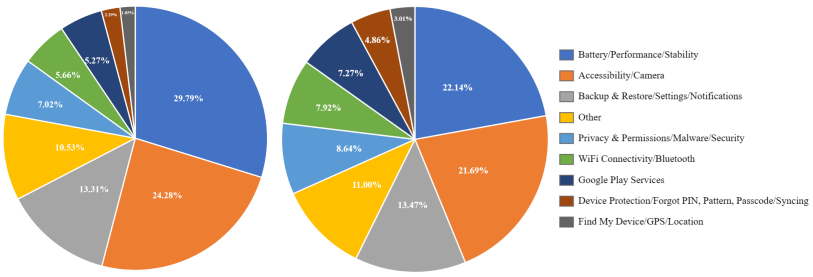


Figure 3: The distribution of incorrect label counts for each group based on manual annotation of 3,750 samples.



(a) Distribution before annotation (b) Distribution after annotation

Figure 4: Distribution of data across groups before and after the annotation process.

Table 8: Groups and categories (Google Android Community categorizes user feedback into 21 categories organized into 9 major groups)

Groups	Categories	Definition
Accessibility / Camera	Accessibility	Problems related to phone accessibility features or those solvable through accessibility settings, including visual or auditory assistance, accessibility apps, or shortcut failures.
	Camera	Issues involving the device camera usage, photo/video capture, or problems such as blurry or corrupted images.
Wi-Fi Connectivity / Bluetooth	Wi-Fi Connectivity	Any issues related to the network, including Wi-Fi connectivity, configuration doubts, disconnections, or slow speeds.
	Bluetooth	Problems involving Bluetooth device pairing, connectivity, or stability.
Privacy & Permissions / Malware / Security	Privacy & Permissions	Issues involving app permission management, privacy settings, or measures taken to protect user privacy/data before an attack occurs. Includes concerns about data protection and user privacy.
	Malware	Any issues caused by malicious software, or user suspicions about unknown or potentially harmful apps.
	Security	Problems where system protection fails, the device is under attack, remotely managed/locked, subject to security restrictions, hacked accounts, or other consequences of system intrusion.
Google Play Services	Google Play Services	Problems related to Google Play Services, including failed system updates, download errors, or related malfunctions.
Backup & Restore / Settings / Notifications	Backup & Restore	Problems related to device data backup, data restoration, or issues involving factory reset.
	Settings	Issues where the user is dissatisfied with interface icons or other non-functional aspects (such as layout or icon placement), or encounters problems when making personalized adjustments or changes to settings. Only personalized adjustments count, not all problems solvable via system settings.
	Notifications	Issues related to notification functionality but not the content of the notifications themselves, or cases where the user receives abnormal or unknown notifications, such as misdelivered notifications or misconfigured notification settings.
Find My Device / GPS / Location	Find My Device	Malfunctions or issues related to the "Find My Device" feature.
	GPS	Any GPS-related problems, including inaccurate positioning or location errors.
	Location	Issues involving all general location services and settings.
Device Protection / Forgot PIN, Pattern, Passcode / Syncing	Device Protection	Device protection issues before an actual attack occurs, including user concerns about mobile security and Factory Reset Protection (FRP)-related problems.
	Forgot PIN, Pattern, Passcode	Problems caused by the user forgetting their phone unlock method (PIN, pattern, or passcode).
	Syncing	Errors in data synchronization between devices or Google account services.
Battery / Performance / Stability	Battery	Problems related to the battery, including battery life, charging faults, or abnormal power consumption suspected to be battery-related.
	Performance	Problems with degraded device responsiveness or slowness, such as lagging, delayed input, or unresponsive apps.
	Stability	Problems involving unexpected system crashes, repeated restarts, or other abnormal behaviors for unknown reasons, as well as overheating or other system stability issues.
Other	Other	Issues not included in the above categories, such as storage problems, physical device damage, or vendor/merchant-related issues.

cause is represented by proposition q. The four categories of interdependencies are formalized as follows:

- **Supportive**: The text provides a claim that is directly evidenced by the image, or vice versa. Formally: $\exists j, k$ such that $p_j^t \vdash p_k^i$.
- **Consistent**: Both modalities convey the same information independently, reinforcing each other. Formally: $\exists j, k$ such that $p_j^t \vdash p_k^i$ and $p_k^i \vdash p_j^t$.
- **Contradictory**: The information provided by the text and image is mutually exclusive or conflicts. Formally: $\exists j, k$ such that $p_j^t \vdash \neg p_k^i$ or $p_k^i \vdash \neg p_j^t$.
- **Complementary**: Neither modality contains sufficient information to identify the root cause q in isolation, but their intersection provides the necessary evidence. Formally: $\forall j, p_j^t \nvdash q$ and $\forall k, p_k^i \nvdash q$, yet $\exists j, k, p_j^t \cap p_k^i \vdash q$.

A.3.1 Illustrative Case Study

To illustrate the **Complementary** relationship, consider the following example from the AUF dataset:

- **Text Proposition (p^t):** "I cannot change the ringtone for the alarm in the settings menu."
- **Image Proposition (p^i):** A screenshot of the battery settings showing a notification: "Battery saver may degrade your alarm and timer experience."
- **Root Cause (q):** The active battery saving mode restricts system-level alarm modifications.

In this case, the text only describes a symptom (cannot change ringtone), and the image only describes a state (battery saver active). Only by correlating these two can the model infer the logical link q.

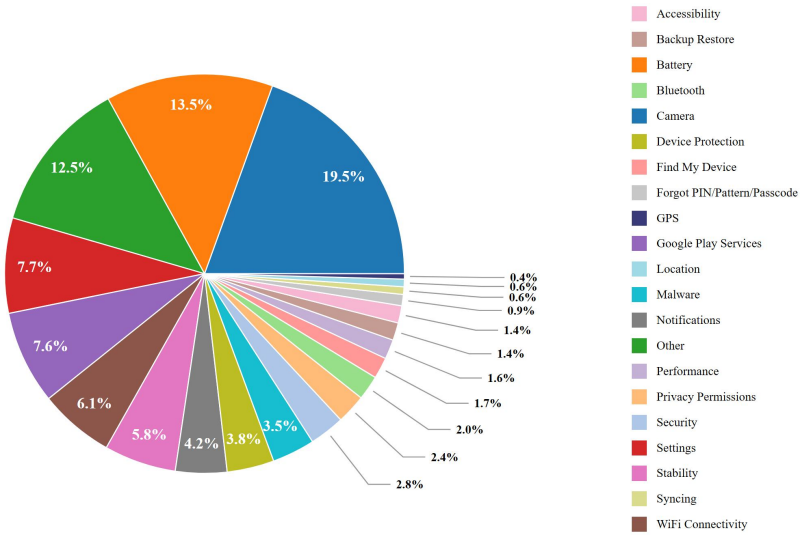


Figure 5: The distribution of data across categories following annotation.

A.4 Prompts

In this section, we present all prompts used in this paper, including Prompt P_b for basic reasoning in the distillation and fine-tuning stages, Prompt P_c for complex reasoning in the distillation and fine-tuning stages, and Prompt P_a designed for adaptive reasoning in the fine-tuning stage.

Input Data

User Description	User Image: ⟨*image_placeholder*⟩	Category Description

Instructions

With the given user feedback case and a predefined set of classification categories, the task is to analyze the case step-by-step and determine the most appropriate category.

```
<Quick Assessment>
   1. Collect and analyze key information from User Description and User Image to understand
   the issue.
   2. State the root cause and determine the correct category based on the evidence.
</Quick Assessment>
<FinalCategory>[Only write the final category name here]</FinalCategory>
```

Figure 6: Prompt P_b for the basic Reasoning in the distillation and fine-tuning stages.

A.5 Adaptive Routing Analysis

We analyze path selection, classification accuracy, and generated output length on the 400-sample AUF testing dataset. The table 9 and 10 report single-run diagnostics for base models, Teacher models, and the final CATSF Student models.

Selection accuracy measures agreement with a model-specific basic-correctness reference: basic reasoning path is the reference path when the corresponding basic-prompt prediction is correct, and complex otherwise. For fine-tuned student models, these reference predictions are obtained from final model S_2. Under-routing selects basic against a complex reference, whereas over-routing selects complex against a basic reference. Selection accuracy is $100[1 - (U + O)/N]$, where U and O are the respective disagreement counts and $N = 400$. Paths are identified from the generated reasoning format. This reference is a post-hoc diagnostic: basic failure does not establish that complex reasoning would succeed.

Output usage is the mean token count of the recorded response, including reasoning and the final category but excluding input prompts and images. All recorded responses are retokenized with the same Qwen2.5 tokenizer to obtain a common output-length measure. For base and Teacher models, the Basic, Complex, and Adaptive columns of Table 10 correspond to different prompting modes. For student models, Basic and Complex are separately fine-tuned fixed-path baselines matched by Teacher and base models, whereas Adaptive denotes the final CATSF Student models.

Table 9 shows higher reference agreement after fine-tuning. With GPT supervision, selection accuracy increases from 68.0% to 74.5% for Qwen-VL and from 63.50% to 80.00% for InternVL; the Gemini-supervised Student models reach 65.8% and 79.5%, respectively. The reference labels themselves vary across models, however, so these scores do not compare selectors against an identical set of routing labels. Basic is the reference path for 81.0–83.5% of samples in the fine-tuned student configurations. An always-basic policy would therefore achieve higher reference agreement than all four CATSF students. Similarly, Gemini obtains 78.3% agreement while never selecting complex reasoning. Raw selection accuracy alone consequently does not establish effective adaptive routing.

The CATSF students nevertheless exhibit different path-usage patterns: InternVL selects complex reasoning for 3.5–6.3% of samples, compared with 12.8–30.3% for Qwen-VL. Their final classification accuracies range from 82.8% to 86.0%. Reference disagreement and classification error are distinct outcomes, since changing the prompt and reasoning process can change the generated prediction. We therefore examine classification performance and output length alongside reference agreement, rather than treating the reference rule as an oracle for optimal routing.

Input Data

User Description	User Image: ⟨*image_placeholder*⟩	Category Description

Instructions

With the given user feedback case and a predefined set of classification categories, the task is to analyze the case step-by-step and determine the most appropriate category.

<Comprehensive Assessment>
1. Collect and analyze key information from User Description and User Image to understand the issue.
2. Correlation Analysis & Evidence Weighting
Determine which type of correlation exists:
- *Consistent*: Both convey the same information or confirm each other, with high semantic overlap.
  → *Strategy*: Formulate the root cause primarily based on the **User Description** (text), treating the User Image as supporting verification.
- *Complementary*: Both provide different but mutually supportive information about the same issue, where combining both leads to a more complete understanding.
  → *Strategy*: **Synthesize** information from both sources. Combine text details and visual cues to construct a complete understanding.
- *Contradictory*: Both point to different root causes or contain conflicting information, requiring further analysis to identify the true issue.
  → *Strategy*: Prioritize the **User Image** (visual evidence) as the ground truth, or focus on the evidence that most clearly aids in correct classification, disregarding the conflicting text.

3. Root Cause Formulation & Category Justification
- *Derive Root Cause*: Based on the strategy in Step 2, clearly define the specific issue.
- *Justify Category*: Compare this Root Cause against the category definitions. Explain **why** this issue belongs to the chosen category.

</Comprehensive Assessment>
<FinalCategory>[Only write the final category name here]</FinalCategory>

Figure 7: Prompt P_c for the complex Reasoning in the distillation and fine-tuning stages.

Input Data

User Description	User Image: ⟨*image_placeholder*⟩	Category Description

Instructions

With the given user feedback case and a predefined set of classification categories, the task is to analyze the case step-by-step and determine the most appropriate category.
And based on the user description and image(s), select either the basic reasoning process or the complex reasoning process for classification. Begin by assessing whether the basic reasoning process adequately addresses the complexity of the user images and description to ensure accurate classification. If the basic reasoning process proves sufficient, apply it; if not, proceed with the complex reasoning process.

Basic reasoning process:

<Quick Assessment>
1. Collect and analyze key information from User Description and User Image to understand the issue.
2. State the root cause and determine the correct category based on the evidence.
</Quick Assessment>
<FinalCategory>[Only write the final category name here]</FinalCategory>

Complex reasoning process:

<Comprehensive Assessment>
1. Collect and analyze key information from User Description and User Image to understand the issue.
2. Correlation Analysis & Evidence Weighting
Determine which type of correlation exists:
- *Consistent*: Both convey the same information or confirm each other, with high semantic overlap.
  → *Strategy*: Formulate the root cause primarily based on the **User Description** (text), treating the User Image as supporting verification.
- *Complementary*: Both provide different but mutually supportive information about the same issue, where combining both leads to a more complete understanding.
  → *Strategy*: **Synthesize** information from both sources. Combine text details and visual cues to construct a complete understanding.
- *Contradictory*: Both point to different root causes or contain conflicting information, requiring further analysis to identify the true issue.
  → *Strategy*: Prioritize the **User Image** (visual evidence) as the ground truth, or focus on the evidence that most clearly aids in correct classification, disregarding the conflicting text.

3. Root Cause Formulation & Category Justification
- *Derive Root Cause*: Based on the strategy in Step 2, clearly define the specific issue.
- *Justify Category*: Compare this Root Cause against the category definitions. Explain **why** this issue belongs to the chosen category.

</Comprehensive Assessment>
<FinalCategory>[Only write the final category name here]</FinalCategory>

Figure 8: Prompt P_a design for adaptive reasoning in the fine-tuning stage.

Table 9: Path selection and classification on AUF. (All models use adaptive inference. Selection accuracy measures agreement with the basic-correctness reference; Under and Over count disagreements with that reference. GPT: GPT-4o Mini; Gemini: Gemini 2.5 Flash.)

Model	Sel. acc. (%)	Complex (%)	Under (#)	Over (#)	Class. acc. (%)
Base models (without task fine-tuning)					
Qwen-VL$_{\text{adap}}$	68.0	4.8	115	12	69.5
InternVL$_{\text{adap}}$	63.5	27.5	75	71	72.8
Teacher models					
GPT$_{\text{adap}}$	71.8	13.8	64	49	83.3
Gemini$_{\text{adap}}$	78.3	0.0	87	0	79.3
Fine-tuned students (GPT supervision)					
Qwen-VL(ft)$_{\text{adap}}$	**74.5**	12.8	59	43	86.0
InternVL(ft)$_{\text{adap}}$	**80.0**	3.5	66	14	85.0
Fine-tuned students (Gemini supervision)					
Qwen-VL(ft)$_{\text{adap}}$	65.8	30.3	46	91	83.0
InternVL(ft)$_{\text{adap}}$	79.5	6.3	65	17	82.8

Table 10: Mean generated output tokens per AUF sample with single-run diagnostics and $N = 400$, counted with a common Qwen2.5 tokenizer. (Input prompts and images are excluded. Bold marks CATSF output lengths).

	Output tokens / sample		
Model	Basic	Complex	Adaptive
Base models (without task fine-tuning)			
Qwen-VL	137.5	240.6	154.8
InternVL	137.6	214.7	181.2
Teacher models			
GPT	139.5	275.6	146.0
Gemini	292.0	604.0	423.6
Fine-tuned students (GPT supervision)			
Qwen-VL(ft)	137.1	274.8	**157.7**
InternVL(ft)	145.9	283.5	**143.8**
Fine-tuned students (Gemini supervision)			
Qwen-VL(ft)	198.6	428.4	**253.6**
InternVL(ft)	223.3	449.7	**221.1**

Note. For base and Teacher models, columns represent prompting modes. For students, Basic and Complex are separately fine-tuned fixed-path baselines; Adaptive is the final CATSF student. GPT: GPT-4o Mini; Gemini: Gemini 2.5 Flash.

Table 10 shows that CATSF students generate 40.8–50.8% fewer output tokens than their matched complex-reasoning baselines. The reductions are 42.6% and 49.3% for GPT-supervised Qwen-VL and InternVL, and 40.8% and 50.8% under Gemini supervision. InternVL's adaptive outputs are approximately as short as those of its basic baselines (approximately 0.99 times their length), whereas Qwen-VL uses 1.15–1.28 times as many output tokens. Path usage does not uniquely determine response length: Gemini selects basic throughout but produces 423.6 tokens under adaptive prompting, compared with 292.0 under basic prompting.

A.6 REPEATED-INFERENCE EVALUATION

To ensure the reliability and statistical rigor of our findings, we conducted multiple independent evaluation runs for the two main experimental settings: the main performance comparisons (Table 3) and the ablation studies (Table 6). Each configuration was evaluated three times with independent inference runs. The aggregated results are reported in Table 11 and Table 12, where all metrics are presented as mean ± standard deviation.

Table 11: Average Performance of Teacher and Student Models on AUF dataset. (GPT: GPT-4o Mini; Gemini: Gemini 2.5 Flash; com: complex; adap: adaptive; A: Accuracy; P: Precision; R: Recall; F_1: Macro F_1 measurement)

Roles	Models	Metrics			
		A(%)↑	**P**(%)↑	**R**(%)↑	F_1↑
Original	Qwen-VL$_{basic}$	68.8±0.8	46.3±1.3	48.1±0.2	44.5±0.5
	Qwen-VL$_{comp}$	67.2±0.4	49.3±1.9	49.5±0.7	46.7±0.3
	Qwen-VL$_{adap}$	70.5±0.9	44.0±2.7	50.7±2.9	45.7±2.4
	InternVL$_{basic}$	71.8±0.4	53.9±0.4	57.7±1.2	53.4±0.8
	InternVL$_{comp}$	70.2±0.9	63.7±2.2	62.3±0.5	59.6±1.3
	InternVL$_{adap}$	71.7±0.9	53.5±4.9	55.0±6.7	51.7±5.9
Teacher	GPT$_{basic}$	83.2±0.8	78.3±3.5	75.9±1.6	75.8±2.1
	GPT$_{comp}$	84.5±1.0	78.5±1.0	76.5±1.3	76.3±1.0
	GPT$_{adap}$	83.3±0.7	79.4±1.4	76.6±1.8	76.4±1.8
Student	Qwen-VL(ft)$_{basic}$	83.1±0.5	80.2±0.3	78.6±0.3	78.3±0.5
	Qwen-VL(ft)$_{comp}$	82.9±0.4	79.9±1.2	77.3±0.4	77.3±0.5
	Qwen-VL(ft)$_{adap}$	85.3±0.6	81.0±0.6	81.0±0.6	79.9±0.4
	InternVL(ft)$_{basic}$	81.2±0.1	77.1±1.8	75.1±0.9	74.2±1.4
	InternVL(ft)$_{comp}$	83.2±0.8	78.9±1.2	78.4±0.8	76.9±1.1
	InternVL(ft)$_{adap}$	85.1±0.1	81.8±0.1	79.5±0.1	78.9±0.2
Teacher	Gemini$_{basic}$	77.7±0.8	73.4±0.1	69.9±1.7	69.9±0.8
	Gemini$_{comp}$	79.6±1.4	78.6±3.4	73.5±1.8	74.3±2.4
	Gemini$_{adap}$	75.2±2.9	71.7±3.7	64.5±0.7	66.6±1.4
Student	Qwen-VL(ft)$_{basic}$	80.4± 0.9	76.7±4.3	73.0±1.8	73.4±2.5
	Qwen-VL(ft)$_{comp}$	81.7±0.2	80.1±1.6	73.3±0.5	74.0±0.8
	Qwen-VL(ft)$_{adap}$	82.4±0.5	77.7±0.2	74.2±1.3	74.7±1.2
	InternVL(ft)$_{basic}$	79.6±1.2	77.0±2.2	71.8±1.5	71.8±1.9
	InternVL(ft)$_{comp}$	79.4±1.6	74.0±3.1	71.3±3.4	71.1±3.3
	InternVL(ft)$_{adap}$	82.6±0.1	76.8±0.1	74.4±0.1	73.8±0.1

Table 12: Average Performance of Ablation study on the mixed training process. (B&C w/o: training without basic and complex samples; Adap w/o: training without the samples for adaptive reasoning; Full ft: full fine-tuning; A: Accuracy; P: Precision; R: Recall; F_1: Macro F_1 measurement).

Teacher	GPT-4o Mini							
Students	Qwen-VL				InternVL			
Metrics	**A**(%)↑	**P**(%)↑	**R**(%)↑	**F$_1$**↑	**A**(%)↑	**P**(%)↑	**R**(%)↑	**F$_1$**↑
B&C w/o	77.3±0.3	70.9±1.9	70.7±1.8	69.3±1.8	78.7±0.1	73.9±0.1	72.5±0.1	70.9±0.1
Adap w/o	82.3±0.3	77.8±0.5	73.2±1.1	73.2±0.4	83.2±0.1	78.6±0.2	74.0±0.1	74.4±0.1
Full ft	85.3±0.6	81.0±0.6	81.0±0.6	79.9±0.4	85.1±0.1	81.8±0.1	79.5±0.1	78.9±0.2
Teacher	Gemini 2.5 Flash							
Student	Qwen-VL				InternVL			
Metrics	**A**(%)↑	**P**(%)↑	**R**(%)↑	**F$_1$**↑	**A**(%)↑	**P**(%)↑	**R**(%)↑	**F$_1$**↑
B&C w/o	76.1±0.4	58.2±1.0	53.6±1.2	53.8±0.9	79.6±0.1	67.5±0.2	69.1±0.2	67.6±0.2
Adap w/o	77.6±0.6	66.1±0.5	65.5±0.5	64.8±0.6	78.1±0.1	68.4±0.1	66.9±0.1	66.1±0.1
Full ft	82.4±0.5	77.7±0.2	74.2±1.3	74.7±1.2	82.6±0.1	76.8±0.1	74.4±0.1	73.8±0.1