MEA: A Reward-Driven Multi-Agent System for Faithful Model Explanations

Anonymous ACL submission

Abstract

Recent years have seen the employment of a plethora of machine learning (ML) models in high-stakes domains, but they remain largely opaque to the practitioners who act on their predictions. While post-hoc explanation methods offer a lens into this model behavior, wielding them effectively demands expertise most domain experts lack: navigating high-dimensional outputs, selecting the best explanations, and synthesizing evidence across disparate tools. To this end, we present MEA, a multi-agent framework that removes the explanation knowledge barrier entirely: a *Proposer* agent selects and configures explanation tools based on the question and modality, while an *Actor* agent is optimized end-to-end against faithfulness, transforming the outputs into natural language explanations grounded in model behavior across tabular, text, and vision modalities. Further, we introduce diverse question types spanning feature attribution, counterfactual reasoning, and spurious feature detection, each paired with a perturbation-based faithfulness metric. We find that frontier agents systematically produce unfaithful explanations. By optimizing against faithfulness rewards augmented with a modality-adaptive penalty, MEA consistently outperforms post hoc explainers, agentic, and closed-source baselines across six datasets, with reward-driven optimization yielding average faithfulness gains of 40%.

1 Introduction

While machine learning (ML) models are increasingly deployed in high-stakes domains, such as healthcare, finance, and legal decision-making (Rajpurkar et al., 2022; Cary Jr et al., 2023; Cao and You, 2024; Flores et al., 2016), their opacity has become a critical concern for practitioners, regulators, and end users, hindering their trustworthiness. This gap between predictive performance and human interpretability is a central challenge that explainability research seeks to address.

Prior works have introduced a plethora of post-hoc explanation methods (Ribeiro et al., 2016; Sundararajan et al., 2017; Selvaraju et al., 2019; Lundberg and Lee, 2017; Pichery, 2014; Smilkov et al., 2017; Springenberg et al., 2015) that probe trained models to generate explanations. However, each method implements a fixed procedure yielding a single output type, *i.e.,* an attribution vector, a heatmap, or a set of super-pixels, independent of what the user actually wants to know. Further, deploying them correctly requires *selecting among competing methods, interpreting high-dimensional outputs, and synthesizing results across tools*, imposing a prohibitive knowledge barrier for domain practitioners without ML expertise.

Conversational explanation frameworks (CEF) address this barrier by allowing users to interrogate models in natural language (Slack et al., 2023; Vanbrabant et al., 2025; Kim et al., 2025a; He and Martens, 2026), mapping free-form queries to structured explanation operations without requiring users to select or configure individual methods. However, these systems share three limitations that remain unaddressed. First, CEF are largely *training-free*: they rely on hand-crafted parsing rules or prompt engineering to route user queries to explanation tools, *without any learned component that adapts to explanation quality or user intent*, resulting in unfaithful explanations. Second, CEF have not been evaluated across large-scale datasets across different modalities. Kroeger et al. (2024) found that LLM-generated explanations fall behind post-hoc explainers and are susceptible to alphabetical bias, questioning whether LLMs can genuinely outperform existing explainers. Third, existing CEF lack a principled, quantitative evaluation framework: they either assess quality through single explanation types (Li et al., 2023; Rawal et al., 2025) or focus on simple linear models, where generated explanations are treated as final outputs with no verification step against model behavior.

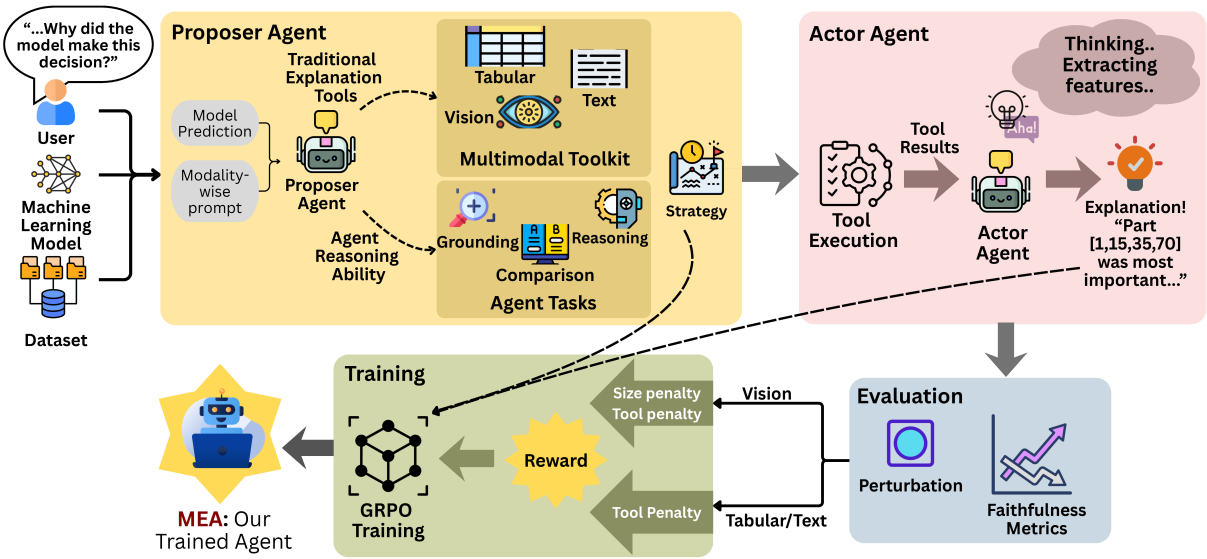

Figure 1: **Overview of MEA.** Given a user question about model behavior, the Proposer agent reasons over the model prediction and input modality to select an explanation strategy, choosing from a multimodal (tabular, text, and vision) toolkit and agent reasoning tasks (grounding, reasoning, and comparison), the Actor agent then executes the strategy, processes raw explanation outputs, and produces a structured natural-language explanation. The explanation is evaluated via input perturbation against faithfulness metrics. In training, both agents are jointly optimized end-to-end via GRPO (Shao et al., 2024), using a reward combining faithfulness and tool penalties.

Present work. We address the above gaps by introducing MEA, a novel multi-agent framework (see Fig. 1) for *prediction-level explainability across diverse modalities*. MEA comprises of a Proposer agent that selects and configures an appropriate combination of tools given a question and data modality, and an Actor agent that executes the tools and synthesizes a natural-language explanation. To scale across diverse explanation difficulty and data modalities, MEA optimizes a multi-modal agent backbone end-to-end via reinforcement learning, with faithfulness scores to directly incentivize explanations that are verifiably grounded in model behavior rather than merely plausible. Since explanation complexity and scale differ fundamentally across modalities, we further introduce a *modality-adaptive reward* that pairs a universal tool-count penalty with modality-specific constraints, preventing reward hacking while encouraging concise, targeted explanations uniformly across all modalities. Further, we introduce the first comprehensive, model- and modality-agnostic question **taxonomy**, comprising diverse question types that spans feature attribution, counterfactual reasoning, and spurious feature detection with their respective faithfulness metrics to quantify alignment between the agent's explanation and the model's actual behavior via perturbation-based testing. To enable large-scale training and evaluation, we construct MEA-BENCH, a dataset covering *three data modalities (tabular, text, and vision), six model architectures, and six datasets, comprising 13K training and 3.5K test instances*.

Our main contributions are as follows: 1) we present MEA, a multi-agent framework that selects, executes, and synthesizes explanation tools into faithful natural-language explanations across tabular, text, and vision modalities; 2) we introduce MEA-BENCH, comprising diverse faithfulness-verified question types, **12** dataset-model combinations, and 13k training and 3.5k test instances across three modalities; 3) we demonstrate that frontier agents systematically hallucinate on explainability tasks and address this by optimizing MEA end-to-end via reinforcement learning, using faithfulness as the reward augmented by a modality-adaptive penalty that prevents reward hacking across modalities; and 4) our trained MEA outperforms post-hoc explainers, closed-source frontier models, and advanced agentic frameworks, yielding average faithfulness gains of **40%**.

2 Related Works

Post-hoc Explainability. Post-hoc explainers attribute a model's prediction to input features without modifying the model itself. Representative approaches include LIME (Ribeiro et al., 2016),

which fits a locally linear surrogate around the instance of interest, and SHAP (Lundberg and Lee, 2017), which provides Shapley-value-based attributions with consistency guarantees. For vision models, Grad-CAM (Selvaraju et al., 2019) and gradient techniques (Sundararajan et al., 2017; Smilkov et al., 2017) supply pixel-level attribution signals.

Research Gap. Existing faithfulness evaluation (Li et al., 2023; Zhang et al., 2025) is conducted *offline* more as a post-hoc assessment, and no prior work adapts it into an *online training signal* to optimize an agentic framework to generate explanations across a comprehensive, multi-question setup spanning multiple modalities.

Conversational and Agentic XAI. Early interactive works framed explanations as dialogues: TalkToModel (Slack et al., 2023) enables natural-language exploration of model behavior, and ECHO (Vanbrabant et al., 2025) grounds answers in computed attributions rather than parametric knowledge. FAX (Kim et al., 2025b) integrates agentic tool use with faithfulness evaluation, though it primarily targets open-ended questions and relies on an LLM-as-a-judge evaluation pipeline. A parallel line of work applies multimodal agents to *mechanistic* interpretability to understand internal model representations, including MAIA (Shaham et al., 2025), OPENMAIA (Camuñas et al., 2025), NEURONSCOPE (Liu et al., 2026); however, these systems are **not** designed for the prediction-level questions that domain practitioners ask.

Research Gap. There are no existing works that jointly optimize a multi-agent pipeline end-to-end with RL across diverse modalities and question types. Our framework addresses this by decomposing the task into a Proposer–Actor structure trained via GRPO with faithfulness as the reward signal.

3 Our Framework: MEA

Problem Formulation. We study natural-language explanation generation for black-box classifiers. Given a trained classifier $f : \mathcal{X} \to \mathcal{Y}$, a question q drawn from a predefined taxonomy $\mathcal{Q}$, and one or more input instances $x \in \mathcal{X}$, MEA generates a structured explanation e that both directly answers q and remains faithful to the model's underlying decision process. MEA supports three input modalities: vision, text, and tabular data, where x is represented as an image, a token sequence, and a feature vector, respectively. Across these modalities, MEA produces explanations that localize the relevant evidence in the input–a bounding box for images, token spans for text, or feature keys for tabular data,thereby identifying the subset of x that accounts for the behavior of f with respect to the question q.

XAI Question Taxonomy. Drawing on prior work (Vanbrabant et al., 2025; Li et al., 2023), we define ten question types organized into three semantic categories (see details in Appendix Table 2): 1) *Feature Attribution* questions ask which input parts drove or suppressed a prediction, *e.g.,* identifying the most responsible region or contrasting why two instances received different predictions; 2) *Counterfactual questions* probe prediction sensitivity to hypothetical modifications, *e.g.,* asking what change would flip the model's decision; and 3) *Spurious feature questions* examine whether the model relies on irrelevant cues, *e.g.,* identifying shared features that cause systematic misclassification across multiple instances. Notably, we have also excluded few questions from training and used them as *out-of-distribution* (OOD) evaluation questions. Unlike other questions, the OODs are only defined over *misclassified* instances, *i.e.,* where $\hat{y} \neq y^*$, which substantially reduces the number of eligible training samples and makes stable gradient estimation difficult. We therefore treat them as a held-out probe of the agent's ability to generalize to unseen question types without task-specific supervision.

3.1 MEA Overview

MEA operates as a two-stage framework built on a shared multimodal LLM backbone θ. In the first stage, the *Proposer Agent* π_θ^{Prop} receives the question q, the input instance $\mathbf{x}$, and the model prediction $\hat{y} = f(\mathbf{x})$, and produces a structured explanation strategy S that specifies which XAI tools to invoke and which autonomous reasoning tasks to perform (refer to Appendix A.2). In the second stage, the *Actor agent* π_θ^{Act} executes the strategy produced by the Proposer, processes the raw attribution outputs returned by each tool, and synthesizes a structured natural-language explanation e. Formally, the two-stage generation process is written as:

$$S \sim \pi_\theta^{\text{Prop}}(S \mid q, \mathbf{x}, \hat{y}, \mathcal{T}_m), \quad (1)$$

$$e \sim \pi_\theta^{\text{Act}}(e \mid q, \mathbf{x}, \hat{y}, S, \{o_t\}_{t \in \mathcal{T}_{\text{sel}}}), \quad (2)$$

where $\mathcal{T}_m$ is the set of XAI tools available for modality m, $\mathcal{T}_{\text{sel}} \subseteq \mathcal{T}_m$ is the subset selected

by the Proposer, and o_t denotes the summarized attribution output of tool t.

3.2 Proposer Agent: Intent-Aware Strategy Formation.

Prior conversational systems rely on fixed attribution pipelines or hand-crafted routing rules to map user queries to explanation tools, with no learned component that adapts to the question semantics. To address this limitation, we introduce the Proposer Agent π_θ^{Prop} which is a trainable agent that performs strategy formation before any tool is executed. Given question type Q_i, input modality m, and model prediction $\hat{y} = f(\mathbf{x})$, it generates a structured strategy $S = (s_{\text{type}}, \mathcal{T}_{\text{sel}}, \mathcal{A}_{\text{auto}})$:

$$S \sim \pi_\theta^{\text{Prop}}(S \mid q, \mathbf{x}, \hat{y}, \mathcal{T}_m), \tag{3}$$

where $s_{\text{type}} \in \{\texttt{autonomous}, \texttt{tools}, \texttt{hybrid}\}$ selects the reasoning mode, $\mathcal{T}_{\text{sel}} \subseteq \mathcal{T}_m$ specifies which tools to invoke from the modality-compatible toolkit, and $\mathcal{A}_{\text{auto}}$ enumerates grounding or reasoning subtasks for the Actor.

3.3 Actor Agent: Tool-Grounded Explanation Synthesis

Even when a well-formed strategy is available, raw outputs are not directly consumable as explanations. Kroeger et al. (2024) demonstrate that LLMs systematically bypass tool evidence in favor of parametric knowledge, generating plausible-sounding but unfaithful conclusions. To address this, the Actor Agent π_θ^{Act} operates in two steps that enforce a structural dependency on attribution evidence.

Step 1: Tool Execution and Summarization. For each tool $t \in \mathcal{T}_{\text{sel}}$, the Actor invokes t against model f and input $\mathbf{x}$, obtaining heatmaps, per-token importance scores, or feature importance vectors. These are converted into structured text summaries recording execution status, a natural-language description, and key attribution statistics, rather than passed as raw tensors. To standardize granularity across modalities, the Actor returns the top 25% of features or tokens for tabular and text inputs, and aggregates top-1% attribution coordinates into a single bounding box for vision inputs.

Step 2: Evidence-Grounded Explanation Generation. The per-tool summaries are assembled into a structured prompt over which the Actor reasons to identify the most relevant features. Given strategy S and outputs $\{o_t\}_{t \in \mathcal{T}_{\text{sel}}}$, the Actor generates:

$$e \sim \pi_\theta^{\text{Act}}(e \mid q, \mathbf{x}, \hat{y}, S, \{o_t\}_{t \in \mathcal{T}_{\text{sel}}}), \quad e \in \mathcal{E}_m, \tag{4}$$

where $\mathcal{E}_m$ is the modality-appropriate output space: a bounding box for vision, a token span for text, or a feature list for tabular data.

3.4 Perturbation-based Faithfulness Evaluation

Evaluating explanation correctness without ground-truth labels is inherently difficult. Prior work either relies on LLM-as-a-judge pipelines susceptible to the same hallucinations they assess (Kroeger et al., 2024; Rawal et al., 2025), or evaluates a single explanation type in isolation (Li et al., 2023; Kim et al., 2025b). No prior work adapts faithfulness evaluation into an online training signal across a multi-question, multi-modal setup. We address this with a unified perturbation-based protocol: an explanation e identifying region r is faithful if and only if masking r produces the model behavior change that q predicts. For each Q_i, metric m_i is computed by masking r in $\mathbf{x}$ and re-querying f, where $\mathbf{x}^{-r}$ denotes the masked input (zero-filled for tabular, grey-filled for vision, span-deleted for text) and $P(y \mid \mathbf{x})$ the predicted class probability.

Feature Attribution (Q1-Q4). Q1-Q2 form a complementary pair: masking the most responsible region should drop confidence; masking the least responsible should not: $m_1 = \max(0, P(\hat{y} \mid \mathbf{x}) - P(\hat{y} \mid \mathbf{x}^{-r}))$ and $m_2 = 1 - |P(\hat{y} \mid \mathbf{x}) - P(\hat{y} \mid \mathbf{x}^{-r})|$. While Q3 checks whether masking the distinctive region collapses the top-1/top-2 decision margin, Q4 measures cross-instance contrastiveness via $\text{sim}(F_A, F_B)$, requiring different attributions for instances with different predictions.

Counterfactual (Q5–Q7). Among the counterfactual questions, Q5 verifies that the predicted direction of change agrees with the observed probability shift, and Q6-Q7 measure whether the proposed modification raises the target-class probability and whether the predicted new class becomes likely: $m_6 = \max\big(0, P(\hat{y} \mid \mathbf{x}^{+\Delta}) - P(\hat{y} \mid \mathbf{x})\big)$ and $m_7 = P(\hat{y}' \mid \mathbf{x}^{-r})$, where $\mathbf{x}^{+\Delta}$ is the agent-proposed modification and $\hat{y}'$ is the predicted class.

Spurious Features. Q8 measures the normalized gain in correct-class probability after masking: $m_8 = \max(0, \min(1, D(\mathbf{x}, r)))$ and $D(\mathbf{x}, r) = \frac{P(y^* \mid \mathbf{x}^{-r}) - P(y^* \mid \mathbf{x})}{1 - P(y^* \mid \mathbf{x})}$, where the denominator normalizes by remaining headroom for fair cross-instance comparison. Q9 averages m_8 over N misclassified instances; Q10 mirrors the contrastive intuition of Q4 on misclassified

instances. Full definitions are in Appendix A.3.

3.5 End-to-End Training

Relying exclusively on raw faithfulness scores to train multi-agent explanation systems frequently precipitates reward hacking. This vulnerability typically manifests through two degenerate strategies: trivial probability-drop maximization (*e.g.,* masking an entire image in vision applications) and indiscriminate tool invocation, which artificially inflates faithfulness without isolating genuinely informative features (Shao et al., 2024). Because naive training objectives inevitably converge on these unfaithful reasoning traces, we employ GRPO (Guo et al., 2025) to jointly optimize both agents end-to-end. During training, each episode captures two distinct transitions: strategy generation by the Proposer and explanation synthesis by the Actor. The faithfulness reward, m_i, is evaluated strictly after the Actor produces its final output, allowing the trajectory-level group-normalized advantage $\hat{A}_k = \frac{R_k - \text{mean}\left(\{R_j\}_{j=1}^{G}\right)}{\text{std}\left(\{R_j\}_{j=1}^{G}\right)}$, is applied uniformly to both transitions, treating the Proposer-Actor as jointly responsible for the final faithfulness outcome. This joint credit assignment is deliberate: a misaligned strategy from the Proposer cannot be recovered by the Actor, so the optimization signal must flow to both agents equally.

Size Penalty (Vision). To foreclose the large-mask shortcut, we penalize large bounding boxes: $m_i^{\text{vision}} = \max(0,\ m_i - \lambda_{\text{size}} \cdot \rho)$, where $\rho \in [0, 1]$ is the bounding-box-to-image area ratio and $\lambda_{\text{size}} = 0.3$. For tabular and text modalities, the fixed top-25% output constraint (Section 3.3) already governs explanation scope, rendering an explicit size penalty unnecessary.

Tool Penalty. To discourage indiscriminate tool invocation, we apply an ℓ_1 tool-count penalty across all modalities: $r = \max\Big(0,\ \min\Big(1,\ m_i - \lambda_{\text{tool}} \cdot \frac{n_{\text{tools}}}{n_{\max}}\Big)\Big)$, where n_{tools} is the number of invoked tools and autonomous tasks, $n_{\max}=9$ and $\lambda_{\text{tool}}=0.05$. The coefficient is kept small to avoid suppressing gradients on question types with inherently low pre-training faithfulness scores (*e.g.,* tabular Q1 and Q8, with mean scores of 0.01 and 0.04 at initialization). Together, these penalties create a dynamic in which the Proposer learns to select informative tools while the Actor learns to ground explanations faithfully in their outputs, with each agent's improvement reinforcing the other through the shared gradient signal.

4 Experiments

We address the following key research questions: RQ1) Does MEA generates faithful explanation across all three modalities? RQ2) Does MEA outperform widely used post hoc explainers? RQ3) Is the prompting style and agentic architecture necessary for MEA's performance? and RQ4) Does MEA generalize to OOD explanation questions, unseen datasets, and novel model architectures?

4.1 Datasets and Experimental Setup

We first describe datasets designed to study the faithfulness of explanations generated by MEA and then outline experimental setup. Figure 2 illustrates MEA's end-to-end pipeline on representative samples from each modality, showing how the Proposer selects appropriate tools and the Actor synthesizes faithful explanations across diverse question types.

Dataset. We construct a multi-modal dataset for analyzing agent behavior across text, tabular and vision modalities (refer to Appendix A.4). The benchmark is generated using predictions from ML models trained on standard supervised learning tasks. For each dataset-model pair, we generate structured questions (see Table 2) covering feature attribution, counterfactual analysis, spurious feature identification, and reasoning failure analysis. In particular, the **tabular** datasets include Adult Census Income (Becker and Kohavi, 1996) and Breast Cancer (Zwitter and Soklic, 1988), evaluated using a two-layer network and TabNet; the **text** datasets include SNLI (Bowman et al., 2015) for natural language inference and IMDb (Maas et al., 2011) for sentiment classification, evaluated using a lightweight neural network and a CNN; and the **vision** datasets include CUB-200-2011 (Welinder et al., 2010) and STL-10 (Coates et al., 2011), evaluated using ResNet-50 (He et al., 2015) and DenseNet-201 (Huang et al., 2018). Each benchmark instance contains the original input, ground-truth label, model prediction, and a natural-language analytical prompt, enabling unified analysis of model behavior, interpretability, and reasoning failures across multiple modalities.

Training Setup. We fine-tune Qwen3.6-35B-A3B (Qwen Team, 2026) using LoRA (r=32) (Hu et al., 2021) for 3 epochs. Following Shao et al. (2024), we set the learning rate to 1×10^{-5}, batch size to 32, and KL coefficient to 0. During training, we generate K=4 rollouts per question at temperature T=1.0 (increased to K=8 for harder

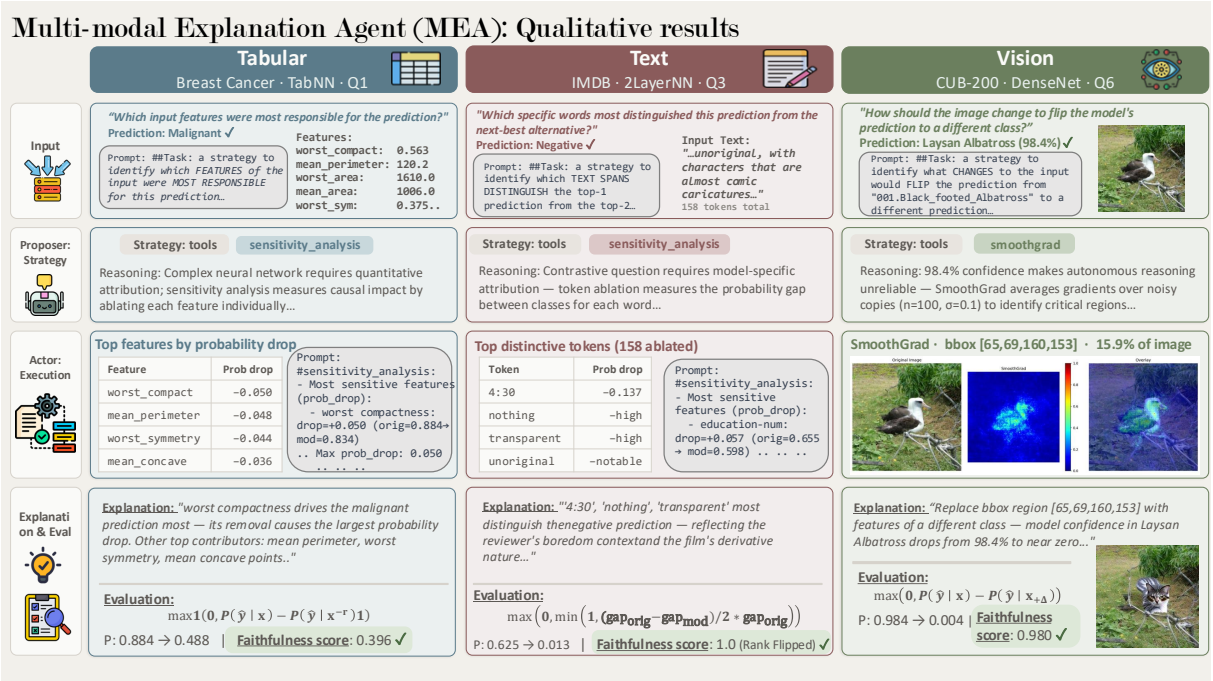

Figure 2: **Case study of MEA across three modalities.** Each column shows a complete pipeline execution for MEA: the input, Proposer's strategy and tool, Actor's execution output, and final explanation with faithfulness evaluation.

types to stabilize advantage variance); evaluation uses T=0.0. Training data is stratified by question type. Advantages are group-normalized strictly over each question's K rollouts, precluding direct cross-type comparison of absolute faithfulness scores. For training across three modalities, we explore *Sequential* and *Hybrid* strategies. In sequential training, MEA is trained first on tabular data, then fine-tuned on text, and finally on vision, each stage initialized from the previous checkpoint. In hybrid, samples from all three modalities are interleaved within each batch, grouped by question type. Our results show that sequential training suffers from catastrophic forgetting: *performance on earlier modalities degrades as training shifts* and the strong prior accumulated in previous stages *reduces behavioral diversity* on the final modality (see Appendix A.5 for more details). We therefore adopt MEA trained on the hybrid strategy for all our main results. Details of computational cost is discussed in Appendix A.10.

Baselines. To isolate the impact of collaborative, iterative refinement, we benchmark MEA against four prompting paradigms: *Zero-Shot* (control), *Chain-of-Thought (CoT)* (Wei et al., 2023), *ReAct* (Yao et al., 2023b), and *Tree of Thoughts (ToT)* (Yao et al., 2023a). All configurations share identical model infrastructure, execution environments, and XAI toolsets, ensuring performance differences stem exclusively from the reasoning methodology. Finally, we compare MEA against standard post-hoc explainers (Ribeiro et al., 2016; Lundberg and Lee, 2017; Smilkov et al., 2017; Selvaraju et al., 2019; Springenberg et al., 2015; Pichery, 2014; Sundararajan et al., 2017).

Evaluation. We follow previous seminal works in explainability (Agarwal et al., 2022) and evaluate explanation quality through *faithfulness*: whether the agent's claimed explanation is actually verified by the target model's behavior when the identified region is perturbed. All metrics Sec. 3.4 follow a common *perturb-and-measure* protocol. Given an agent explanation that identifies a region r of the input (a list of feature keys for tabular data, a set of text spans for text, or a bounding box for vision), we apply a modality-appropriate perturbation with a new input x' and re-run the prediction, and evaluate on the proposed metrics (3.4). For vision, salient regions are replaced with a gray fill, for counterfactual, we inpaint with Stable Diffusion (Rombach et al., 2022); for text, spans are deleted or replaced with agent-provided counterfactual text; for tabular data, features are mean-filled or set to agent-specified values.

5 Results

Here, we discuss experimental results that answer key questions (RQ1-RQ4) highlighted before.

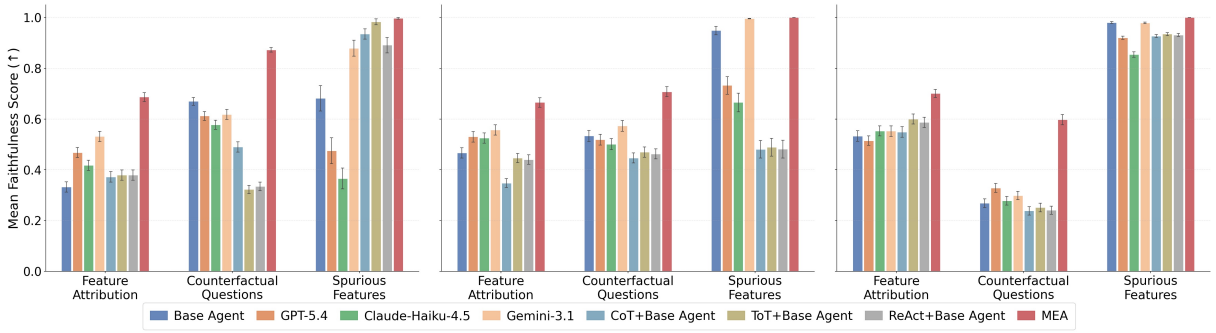

Figure 3: **MEA's performance.** Faithfulness scores of MEA vs. baselines across three modalities: **(A)** tabular, **(B)** text, and **(C)** vision. Higher mean faithfulness scores indicate more faithful explanations. Results show that MEA outperform all baselines, including closed-source models and techniques like CoT, ToT, and ReACT.

RQ1) MEA improves explanation faithfulness across all modalities. A key challenge in XAI is that different explanation needs (feature attribution, counterfactual reasoning, and spurious feature detection) traditionally require separate tools. MEA addresses this by serving as a unified framework that handles diverse question types across all three modalities without modality-specific engineering. Results in Fig. 3 show that MEA achieves the highest faithfulness scores across all three modalities. On tabular data, MEA scores 0.873 on counterfactual questions, outperforming the best baseline (Base Agent, 0.670) by **30%**, and achieves 0.997 on spurious features vs. Gemini-3.1 at 0.879. On text, MEA scores 0.708 on counterfactual questions versus Gemini-3.1 at 0.574 (**+23%**). The gap is largest on vision counterfactual questions, where MEA achieves 0.598 versus GPT-5.4 at 0.328 (**+82%**). Across all three modalities, MEA attains a perfect or near-perfect spurious feature detection score (0.997/1.000/1.000). To verify that similarities between MEA and baselines reflect genuine statistical equivalence, we conducted two-sided Welch's t-tests (WELCH, 1947) on all targeted comparisons. On spurious-feature questions (Q10), MEA achieves ceiling performance (1.000) on both text and vision; the strongest baselines (Gemini-3.1: 0.996 text, Base Agent: 0.981 vision) differ by at most 0.02, a gap that is statistically significant due to MEA's zero variance at ceiling. See Appendix A.9 for complete statistics.

These results confirm that MEA acts as a `one-stop-shop` delivering strong, reliable faithfulness across all ten question types and all three modalities. MEA also outperforms all closed-source models and alternative agentic frameworks (CoT, ToT, ReAct) on all question types across all modalities, confirming its role as a unified solution for diverse explainability needs.

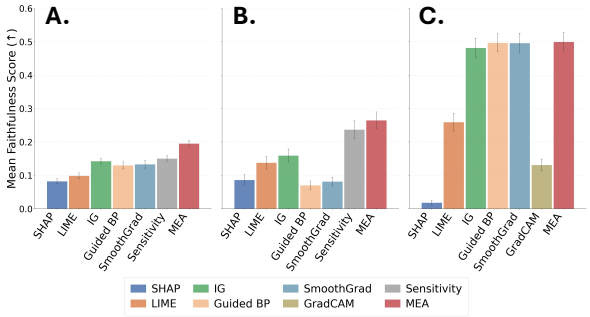

Figure 4: **MEA vs. Explainers.** Mean faithfulness scores of identifying most salient input features across **(A)** tabular, **(B)** text, and **(C)** vision modalities. MEA outperforms all baselines on tabular and text inputs, and matches the best-performing method on vision inputs.

RQ2) MEA outperforms post-hoc explainers. We compare MEA against widely-used post-hoc explainers to understand its effectiveness. Since post-hoc explainers produce modality-specific outputs (*e.g.,* heatmaps, feature importance scores), a fair cross-method comparison is only possible for Q1: *Which parts of the instance are most responsible for the model's prediction?*. Following the same output standardization used in MEA, we extract the top-25% of features for tabular and text inputs, and the top-1% of pixels aggregated into a bounding box (Sec. 3.3) for vision inputs. Results in Fig. 4 show the mean faithfulness scores across all three modalities. On tabular data, MEA scores 0.195, outperforming the best post-hoc explainer (sensitivity analysis, 0.150) by **30%**. On text, MEA scores 0.265 vs. sensitivity analysis at 0.237 (**+12%**). On vision, MEA (0.499) matches the best single-tool baseline Guided Backpropagation (0.497). Appendix A.9 shows that vision feature-attribution (Q1), MEA achieves faithfulness scores

statistically indistinguishable Guided Backpropagation ($p = 0.944$) and other baselines. These results demonstrate that MEA, despite operating as a **general-purpose agent** across all question types, is competitive with or superior to specialized single-purpose explainers on feature attribution.

RQ3) MEA is robust to different prompt styles and outperforms critic-augmented agents. Here, we first test sensitivity to question phrasing by evaluating three paraphrased variants of the question taxonomy (see A.7) and compare mean faithfulness against the original. Next, we investigate whether augmenting a base agent with a Critic (Shinn et al., 2023), which receives the original strategy and explanation, critiques them, and regenerates improved outputs, can close the gap with MEA. Results in Appendix A.8 Fig. 8 show that paraphrase variants score within $< \mathbf{1}\%$ of the original across all modalities and question groups (*e.g.,* tabular counterfactual: paraphrase 0.934 vs. MEA 0.939; vision feature attribution: 0.686 vs. 0.692), confirming that MEA is robust to prompt rephrasing. For the critic ablation (Fig. 5), MEA consistently outperforms Agent+Reflection despite the added inference step. On text counterfactual questions, Agent+Reflection scores 0.644 while MEA reaches 0.876 (**+36%**); on vision counterfactual questions, 0.439 vs. 0.569 (**+30%**). Strikingly, on tabular spurious features the critic step even *degrades* performance (0.514) relative to the base agent (0.677), while MEA achieves 0.997. Appendix A.9 shows paraphrase robustness tests results, which demonstrate no significant performance degradation across 8 of 9 question-type×modality conditions ($p > 0.05$ in all but vision counterfactual). These results demonstrate that RL-based end-to-end training provides substantially stronger and more reliable gains than inference-time critique.

RQ4) MEA generalizes to OOD datasets and model architectures. We evaluate MEA on unseen datasets and model architectures across all modalities (Yelp Review Polarity dataset (Zhang et al., 2016) with a BERT-based classifier (Devlin et al., 2019) for text; German Credit dataset (Dua and Graff, 2019) with a three-layer neural network for tabular; CIFAR-10 (Krizhevsky et al., 2009) with ResNet-18 (He et al., 2015) for vision). Results in Table 1 show that MEA achieves the highest overall faithfulness on all three modalities (0.581 vs. 0.379 on tabular, 0.689 vs. 0.625 on text, 0.576 vs. 0.448 on vision). The largest gains appear on tabular inputs (+**53%**), with feature attribution alone rising from 0.447 to 0.783 (**+75%**). On vision, counterfactual faithfulness improves from 0.246 to 0.521 (**+112%**). MEA only underperforms the base agent on text counterfactual questions (0.770 vs. 0.860) and vision feature attribution (0.583 vs. 0.621), which we attribute to further hyperparameter tuning needed for better modality-specific trade-offs. Overall, these results confirm that MEA **generalizes** to unseen datasets and model architectures (see Appendix Table 5 for detailed results and Appendix A.6 for an ablation study of OOD question types).

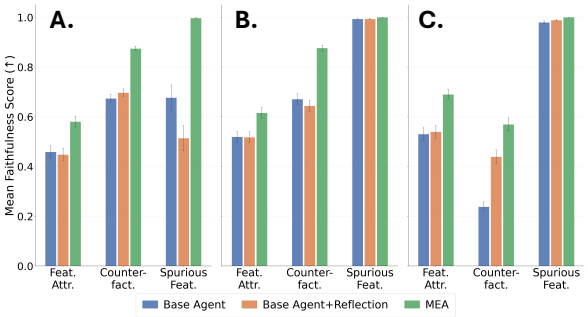

Figure 5: **Ablation on critic agent.** Mean faithfulness scores across three que1on categories for the base agent, base agent augmented with a critic agent, and MEA, evaluated on **(A)** tabular, **(B)** text, and **(C)** vision modalities. MEA consistently outperforms both baselines, demonstrating that RL-based training yields stronger improvements than inference-time critique.

Table 1: OOD faithfulness scores on different datasets and model architectures. Best performance is **bolded**.

Modality	Base Agent	MEA
Tabular	0.379	**0.581**
Text	0.625	**0.689**
Vision	0.448	**0.576**

6 Conclusion

We presented MEA, a multi-agent framework for faithful, prediction-level explanation across tabular, text, and vision modalities. To support training and evaluation, we introduced MEA-BENCH, a model- and modality-agnostic benchmark of 16k instances each paired with a faithfulness metric. Optimizing MEA end-to-end via RL with a modality-adaptive reward yields a trained agent that consistently outperforms post-hoc explainers, frontier models, and advanced agentic frameworks by an average faithfulness margin of 40%, demonstrating that **learning to explain** is a more principled path toward trustworthy AI than prompting to explain.

7 Limitations

The limitations of our work can be summarized in the below points:

i) **Restricted to classification tasks.** MEA is formulated and evaluated exclusively for black-box classifiers. Regression models, generative systems, and structured prediction architectures fall outside the current scope, as the perturbation-based faithfulness protocol relies on class probability signals that are not directly applicable to continuous or Autoregressive output spaces. Generalization to other learning tasks would require task-specific faithfulness metrics.

ii) **Predefined and closed tool library.** MEA selects from a fixed set of established attribution methods covering the major gradient-based and perturbation-based paradigms. Generalization to emerging or domain-specific explanation tools would require new reward signals and faithfulness metrics, which were computationally prohibitive in this work.

iii) **Single backbone evaluation.** All experiments use Qwen3.6-35B-A3B as the policy backbone due to our computational constraints. Extending the framework to larger models is an important direction for future work.

iv) **Perturbation-based faithfulness.** The masking strategies used for evaluation produce inputs that may induce model behavior driven by distributional shift, a known limitation shared by all perturbation-based XAI evaluation protocols.

8 Ethics Statement

MEA is intended to make ML model behavior more accessible to domain practitioners in high-stakes settings such as healthcare, finance, and legal decision-making, but this goal comes with important responsibilities. Users without ML expertise may treat MEA's explanations as ground truth rather than as descriptions of model behavior, so deployment in sensitive contexts must be accompanied by clear communication that faithfulness does not equal causal correctness. Since MEA faithfully reflects whatever the underlying classifier has learned, it will also faithfully explain biased or discriminatory patterns without flagging them — users should understand that a faithful explanation of a biased model is not a safe one. More broadly, easy access to automated explanations may reduce the depth of human scrutiny applied to model decisions, and MEA should therefore be treated as a tool that supports human judgment, not one that replaces it. On the security side, MEA's counterfactual and spurious feature capabilities could in principle be repurposed to craft adversarial inputs or exploit model weaknesses, and we encourage appropriate access controls before deployment in sensitive systems. Finally, since MEA requires query access to the target classifier, any deployment involving personal data such as medical records or financial transactions must comply with applicable privacy regulations, including HIPAA and GDPR.

References

Chirag Agarwal, Satyapriya Krishna, Eshika Saxena, Martin Pawelczyk, Nari Johnson, Isha Puri, Marinka Zitnik, and Himabindu Lakkaraju. 2022. Openxai: Towards a transparent evaluation of model explanations. *Advances in neural information processing systems*, 35:15784–15799.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*.

Barry Becker and Ronny Kohavi. 1996. Adult. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5XW20.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. *Preprint*, arXiv:1508.05326.

Josep Lopez Camuñas, Christy Li, Tamar Rott Shaham, Antonio Torralba, and Agata Lapedriza. 2025. OpenMAIA: a multimodal automated interpretability agent based on open-source models. In *Mechanistic Interpretability Workshop at NeurIPS 2025*.

Kai Cao and Haifeng You. 2024. Fundamental analysis via machine learning. *Financial Analysts Journal*, 80(2):74–98.

Michael P Cary Jr, Anna Zink, Sijia Wei, Andrew Olson, Mengying Yan, Rashaud Senior, Sophia Bessias, Kais Gadhoumi, Genevieve Jean-Pierre, Demy Wang, and 1 others. 2023. Mitigating racial and ethnic bias and advancing health equity in clinical algorithms: a scoping review: scoping review examines racial and ethnic bias in clinical algorithms. *Health Affairs*, 42(10):1359–1368.

Adam Coates, Andrew Ng, and Honglak Lee. 2011. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 215–223, Fort Lauderdale, FL, USA. PMLR.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *Preprint*, arXiv:1810.04805.

Dheeru Dua and Casey Graff. 2019. Uci machine learning repository. `http://archive.ics.uci.edu/ml`.

Anthony W Flores, Kristin Bechtel, and Christopher T Lowenkamp. 2016. False positives, false negatives, and false analyses: A rejoinder to machine bias: There's software used across the country to predict future criminals. and it's biased against blacks. *Fed. Probation*, 80:38.

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

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2015. Deep residual learning for image recognition. *Preprint*, arXiv:1512.03385.

Yifan He and David Martens. 2026. An agentic approach to generating xai-narratives. *Preprint*, arXiv:2603.20003.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. 2018. Densely connected convolutional networks. *Preprint*, arXiv:1608.06993.

Haechang Kim, Hao Chen, Can Li, and Jong Min Lee. 2025a. Talktoagent: A human-centric explanation of reinforcement learning agents with large language models. *Preprint*, arXiv:2509.04809.

Jaechang Kim, Sunung Mun, Seungjoon Lee, Jaewoong Cho, and Jungseul Ok. 2025b. Towards faithful agentic XAI.

Alex Krizhevsky, Geoffrey Hinton, and 1 others. 2009. Learning multiple layers of features from tiny images.(2009).

Nicholas Kroeger, Dan Ley, Satyapriya Krishna, Chirag Agarwal, and Himabindu Lakkaraju. 2024. In-context explainers: Harnessing llms for explaining black box models. *Preprint*, arXiv:2310.05797.

Thinking Machines Lab. 2025. Tinker.

Xuhong Li, Mengnan Du, Jiamin Chen, Yekun Chai, Himabindu Lakkaraju, and Haoyi Xiong. 2023. $\mathcal{M}^4$: A unified XAI benchmark for faithfulness evaluation of feature attribution methods across metrics, modalities and models. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Weiqi Liu, Yongliang Miao, Haiyan Zhao, Yanguang Liu, and Mengnan Du. 2026. Neuronscope: A multi-agent framework for explaining polysemantic neurons in language models. *Preprint*, arXiv:2601.03671.

Scott Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. *Preprint*, arXiv:1705.07874.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

C. Pichery. 2014. Sensitivity analysis. *Encyclopedia of Toxicology*.

Qwen Team. 2026. Qwen3.6-35B-A3B: Agentic coding power, now open to all.

Pranav Rajpurkar, Emma Chen, Oishi Banerjee, and Eric J Topol. 2022. Ai in health and medicine. *Nature medicine*, 28(1):31–38.

Kaivalya Rawal, Zihao Fu, Eoin Delaney, and Chris Russell. 2025. Evaluating model explanations without ground truth. In *Proceedings of the 2025 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '25, page 3400–3411. ACM.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should i trust you?": Explaining the predictions of any classifier. *Preprint*, arXiv:1602.04938.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. *Preprint*, arXiv:2112.10752.

Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2019. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359.

Tamar Rott Shaham, Sarah Schwettmann, Franklin Wang, Achyuta Rajaram, Evan Hernandez, Jacob Andreas, and Antonio Torralba. 2025. A multimodal automated interpretability agent. *Preprint*, arXiv:2404.14394.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *Preprint*, arXiv:2402.03300.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Preprint*, arXiv:2303.11366.

Dylan Slack, Satyapriya Krishna, Himabindu Lakkaraju, and Sameer Singh. 2023. Talktomodel: Explaining machine learning models with interactive natural language conversations. *Preprint*, arXiv:2207.04154.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. 2017. Smoothgrad: removing noise by adding noise. *Preprint*, arXiv:1706.03825.

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. 2015. Striving for simplicity: The all convolutional net. *Preprint*, arXiv:1412.6806.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. *Preprint*, arXiv:1703.01365.

Sebe Vanbrabant, Gilles Eerlings, Gustavo Alberto Rovelo Ruiz, and Davy Vanacken. 2025. Echo: Enhancing conversational explainable ai through tool-augmented language models. *Proc. ACM Hum.-Comput. Interact.*, 9(4).

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

B. L. WELCH. 1947. The generalization of 'student's' problem when several different population varlances are involved. *Biometrika*, 34(1-2):28–35.

Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. 2010. Caltech-UCSD birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 2023a. Tree of thoughts: Deliberate problem solving with large language models. *Preprint*, arXiv:2305.10601.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023b. React: Synergizing reasoning and acting in language models. *Preprint*, arXiv:2210.03629.

Shichang Zhang, Tessa Han, Usha Bhalla, and Himabindu Lakkaraju. 2025. Towards unified attribution in explainable ai, data-centric ai, and mechanistic interpretability. *Preprint*, arXiv:2501.18887.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2016. Character-level convolutional networks for text classification. *Preprint*, arXiv:1509.01626.

Matjaz Zwitter and Milan Soklic. 1988. Breast Cancer. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C51P4M.

A Appendix

The Appendix provides supplementary material supporting the main paper, including the complete question taxonomy (A.1), details of the XAI tools employed (A.2), , faithfulness evaluation metrics (A.3), dataset statistics (A.4), descriptions of the sequential and hybrid training procedures (A.5), ablation studies on out-of-distribution (OOD) question types (A.6), prompt robustness (A.7), and the critic agent (A.8), as well as statistical significance analyses (A.9) and computational cost details (A.10).

A.1 Question Taxonomy

The detailed question definition and category is showed in Table 2.

A.2 XAI Tools

To support explanation generation, our framework provides a library of established attribution methods spanning all three modalities:

Gradient-based. GradCAM (Selvaraju et al., 2019) (layer-wise gradient attribution, vision only), Integrated Gradients (Sundararajan et al., 2017), Guided Backpropagation (Springenberg et al., 2015), SmoothGrad (Smilkov et al., 2017) (noise-averaged gradient estimation), and Sensitivity Analysis (Pichery, 2014) (input-gradient saliency, text and tabular only).

Perturbation-based. LIME (Ribeiro et al., 2016) (Local Interpretable Model-agnostic Explanations with superpixel or token-level segmentation for vision, and token-level perturbation for text) and SHAP (Lundberg and Lee, 2017) (Shapley Additive exPlanations via marginal contribution estimation).

A.3 Faithfulness Metrics

Here $\text{gap}_{\text{orig}}, \text{gap}_{\text{mod}}$ are the original gap between top-1 and top-2 predicted classes and the gap after perturbation; y^* is the ground-truth class; $\hat{y}$ is the predicted class after perturbation; δ is the prediction probability drop after perturbation; and $F_{\text{correct}}, F_{\text{wrong}}$ are attribution maps for the correctly and incorrectly classified instances respectively.

A higher metric score indicates a more faithful explanation. For binary metrics (Q3, Q5, Q6, Q7), the score is 1 if the condition holds and 0 otherwise. For Q6, tabular and text inputs are evaluated against

Table 2: Overview of the ten explanation questions organised by category.

ID	Category	Question
Feature Attribution - which parts of the input drove or suppressed the prediction?		
Q1	Most Responsible	Which parts of the instance are most responsible for the model's prediction?
Q2	Least Responsible	Which parts of the instance are least responsible for the model's prediction?
Q3	Distinctive	Which specific parts of the instance distinguish its prediction from the next-best alternative?
Q4	Contrastive	Why are instances A and B given different predictions?
Counterfactual - how sensitive is the prediction to hypothetical modifications?		
Q5	Mask Prediction	If we mask a certain part of this instance, would the prediction change?
Q6	Flip Prediction	How should the instance change to flip the model into a different prediction?
Q7	Change Prediction	If we remove or change one important part of the instance, how would the prediction change?
Spurious Feature - does the model rely on irrelevant cues?		
Q8	Irrelevant Parts	Is there any irrelevant part in this instance that causes the model's wrong prediction?
Q9	Shared Feature	What shared feature makes misclassified instances A, B, C, ... difficult for the model?
Q10	Similar-Different	Why are instances A and B assigned different predictions despite apparent similarity?

a specific target class y_t, whereas vision inputs are evaluated by whether the prediction changes to any alternative class, reflecting the substantially larger label space in vision tasks.

The detailed metric calculation method and intuition is showed in Table 3.

A.4 Dataset Statistics

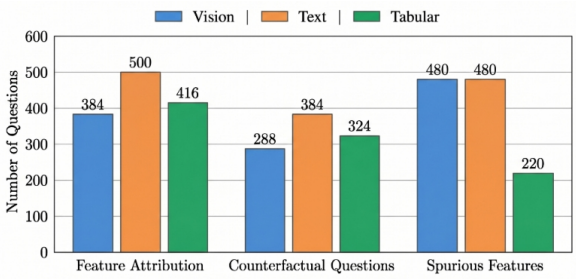

Figure 6: Dsitribution of data across modalities.

Due to the heterogeneous prediction behavior exhibited across models and datasets, the number of generated instances varies across categories (**Q1-Q10**). Different question types require specific prediction configurations, such as correct predictions, informative misclassifications, contrastive prediction pairs, or shared behavioral patterns across models. Consequently, the final instance count for each category is determined by the availability of meaningful and semantically valid examples satisfying these construction criteria. This naturally results in varying sample counts across question types, while ensuring that each generated instance reflects authentic model behavior and preserves the fidelity of the underlying prediction distributions.

A.5 Sequential and Hybrid Training

We compare two multi-modal GRPO training schedules, both using Qwen3-VL-30B-A3B-Instruct(Bai et al., 2023). *Sequential* training fine-tunes one modality at a time, carrying each checkpoint forward to the next phase. *Hybrid* training interleaves all three modalities within every batch, grouped by question type via Q-type stratification.

Sequential Training

Phase 1 - Tabular. Fine-tuning on tabular data across 3 epochs drives the mean batch reward from 0.3 to 0.6. The resulting checkpoint achieves a tabular faithfulness score of 0.632.

Phase 2 - Text. Continuing from the tabular checkpoint drives the batch reward from 0.4 to 0.6 on text examples. However, re-evaluating the new checkpoint on the *tabular* test set reveals measurable catastrophic forgetting. The degradation happens across question types: Q3 drops from 0.326 to 0.264 (-0.06) and Q6 from 0.780 to 0.728 (-0.05). Both Q-types demand accurate numerical reasoning over feature rankings or feature value intervals – skills reinforced during tabular training and partially overwritten as the gradient signal shifts to text-span operations.

Table 3: Summary of evaluation metrics corresponding to different explanation faithfulness questions.

Question	Metric m_i	Intuition
Q1	$\max\big(0, P(\hat{y} \mid \mathbf{x}) - P(\hat{y} \mid \mathbf{x}^{-r})\big)$	Masking the most responsible region should drop predicted-class confidence
Q2	$1 - \big\lvert P(\hat{y} \mid \mathbf{x}) - P(\hat{y} \mid \mathbf{x}^{-r})\big\rvert$	Masking the least responsible region should leave confidence unchanged
Q3	$\max\left(0, \min\left(1, \frac{\text{gap}_{\text{orig}} - \text{gap}_{\text{mod}}}{2 * \text{gap}_{\text{orig}}}\right)\right)$	The top-1/top-2 margin should collapse after masking the distinctive region
Q4	$1 - \text{sim}(F_A, F_B)$	The agent should attribute different features to each instance
Q5	$\begin{cases} \max(0,\ \delta) & \hat{c} = 1 \text{ (predicts change)} \\ \max(0,\ 1 - \lvert\delta\rvert) & \hat{c} = 0 \text{ (predicts stable)} \end{cases}$	Probability change should agree in direction with the agent's yes/no answer
Q6	$\begin{cases} \max(0, P(\hat{y} \mid \mathbf{x}) - P(\hat{y} \mid \mathbf{x}_{+\Delta})) & \text{vision} \\ \max(0, \min(1, P(y_t \mid \mathbf{x}_{+\Delta}))) & \text{text/tab} \end{cases}$	Applying the proposed change should raise the probability of the target class
Q7	$P(\hat{y} \mid \mathbf{x}^{-r})$	The agent's predicted new class should have high probability after masking
Q8	$\max\left(0, \min\left(1, \frac{P(y^* \mid \mathbf{x}^{-r}) - P(y^* \mid \mathbf{x})}{1 - P(y^* \mid \mathbf{x})}\right)\right)$	Masking the spurious region should improve correct-class probability
Q9	$\frac{1}{N}\sum_{i=1}^{N} \max\left(0, \min\left(1, \frac{\Delta_i}{1 - P(y_i^* \mid \mathbf{x}_i)}\right)\right)$	Improvement should hold across all N misclassified instances
Q10	$1 - \text{sim}(F_{\text{corr}}, F_{\text{wrong}})$	Features attributed to the correct vs. wrong instance should differ

Phase 3 - Vision. Vision training for 200 steps shows a markedly weaker learning signal: the batch reward rises only from 0.512 to 0.578, compared to the steep trajectories in Phases 1 and 2 (Figure 7, left). We attribute the limited learning to two compounding effects. First, by Phase 3 the model's policy is already tightly peaked around tabular/text output formats, leaving insufficient entropy to explore the vision-specific bounding-box. Second, the flat reward curve indicates that rollout diversity has collapsed: all sampled trajectories produce similar rewards, so the group-normalized advantage $\hat{A}_k \approx 0$ for most rollouts, suppressing the gradient signal.

Hybrid Training

Hybrid training on all three modalities simultaneously runs for 902 steps across 3 epochs, with the batch reward rising from 0.313 to 0.594 (Figure 7, right). The slower convergence relative to any single-modality phase reflects the wider diversity of the joint training distribution: the model must maintain a flexible policy across tabular, text, and vision inputs simultaneously.

A.6 Ablation Study: OOD Question Types

We use spurious feature questions as OOD question types, as MEA is never trained on them (see Sec. 3). Results in Table 4 show that MEA achieves higher overall faithfulness on tabular and text modalities, with particularly notable gains on Q8 for text. On vision inputs, the base agent slightly outperforms MEA, suggesting that generalization to spurious feature reasoning remains challenging for vision modality without explicit training signal.

Table 4: OOD faithfulness scores on held-out question types Q8 and Q9. Best per row in **bold**.

Modality	Question Type	Base Agent	MEA
Tabular	Q8	0.132	**0.184**
	Q9	**0.024**	0.019
	Overall	0.081	**0.106**
Text	Q8	0.186	**0.353**
	Q9	0.066	**0.075**
	Overall	0.116	**0.186**
Vision	Q8	**0.068**	0.050
	Q9	**0.070**	0.023
	Overall	**0.069**	0.034

A.7 Ablation Study: Prompt Robustness

To assess prompt robustness, we generate three semantically-equivalent paraphrases of every question in the taxonomy. Dynamic values (feature names, row IDs, class labels, masked tokens) are extracted from the original question via regular expressions and re-injected verbatim into each paraphrase template, preserving semantic equivalence. Q1-Q4 receive three lexically distinct vari-

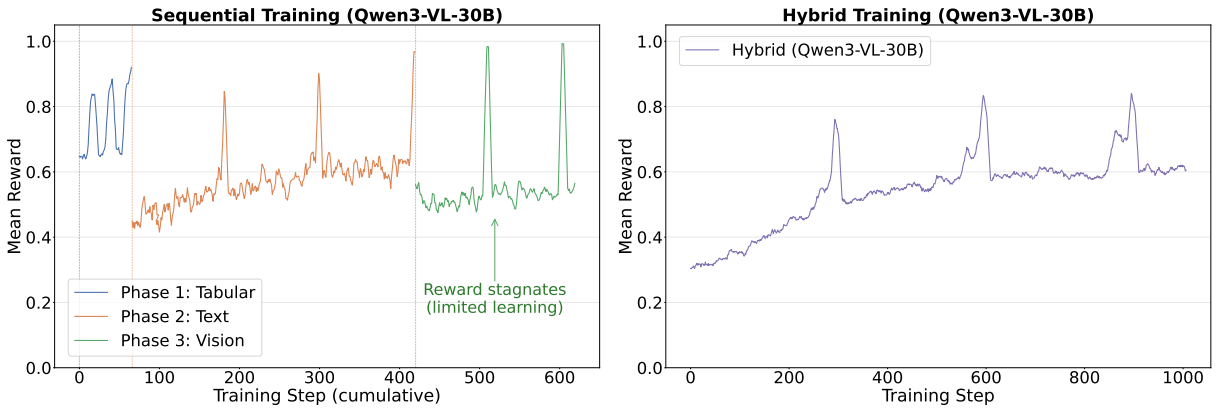

Figure 7: **Training reward curves. (Left)** *Sequential training:* Vertical dashed lines mark phase boundaries (Tab → Text → Vision); the vision reward barely rises, indicating collapsed rollout diversity. **(Right)** *Hybrid training (all modalities interleaved):* The recurring spikes are an artifact of the sampling seed, which concentrates relatively easy questions towards the end of each epoch.

Table 5: OOD faithfulness scores on different datasets and model architectures. Best performance is **bolded**.

Modality	Question Group	Base Agent	MEA
Tabular	Feature Attribution	0.447	**0.783**
	Counterfactual	0.422	**0.499**
	Spurious Features	0.264	**0.349**
	Overall	0.379	**0.581**
Text	Feature Attribution	0.428	**0.563**
	Counterfactual	**0.860**	0.770
	Spurious Features	0.639	**0.778**
	Overall	0.625	**0.689**
Vision	Feature Attribution	**0.621**	0.583
	Counterfactual	0.246	**0.521**
	Spurious Features	0.416	**0.624**
	Overall	0.448	**0.576**

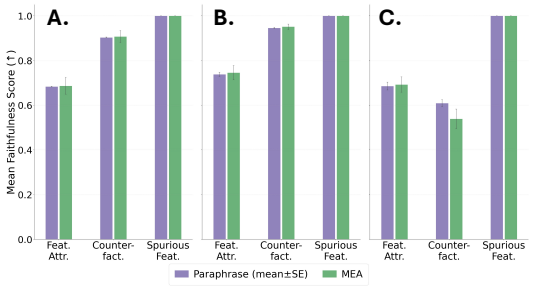

Figure 8: **Ablation on prompt robustness.** Mean faithfulness scores of MEA evaluated on three paraphrased variants of the question taxonomy across **(A)** tabular, **(B)** text, and **(C)** vision modalities. Minimum variance indicates that MEA is robust to changes in question phrasing.

ants; Q5-Q7 and Q10 share a single uniform paraphrase across all three runs (variant-independent templates). Table 6 lists the question-field phrasings for each type, illustrated with tabular-modality examples. Due to computational constraints, paraphrase evaluation is conducted on one dataset per modality.

A.8 Ablation Study: Critic Agent

The Critic Agent augments the base agent pipeline with an inference-time reflection step. After the Actor Agent produces a strategy and explanation and receives a faithfulness score, the Critic receives a structured prompt and returns two critiques: a *Proposer Reflection* (feedback on tool selection) and an *Actor Reflection* (feedback on explanation quality). The actor agent then regenerates its strategy and explanation conditioned on this feedback.

Prompt. Table 10 shows the complete prompt structure passed to the Critic Agent. Each section is populated with runtime values extracted from the pipeline outputs.

Output Schema. The Critc returns two feedbacks, separated by the markers `PROPOSER_REFLECTION` and `ACTOR_REFLECTION`. Table 7 describes each field. The Proposer Reflection is consumed by the Proposer Agent to revise its tool selection; the Actor Reflection is consumed by the Actor Agent to revise its explanation.

A.9 Statistical Significance Tests

We report two-sided Welch's t-tests (unequal variance, unpaired) (WELCH, 1947) for comparisons against independent baselines, and paired t-tests for the paraphrase ablation (where scores are matched by sample ID across conditions). The significance level is $\alpha = 0.05$; "n.s." denotes $p > 0.05$. All values are mean $\pm$ standard error of the mean (SEM).

Vision Feature-Attribution Tool-Only Baselines.

Table 8 compares MEA (Q1, vision; $n = 96$, mean$= 0.499 \pm 0.029$) against six tool-only XAI

Table 6: Three paraphrase variants of each question type (*Q-field only*; tabular modality shown). Dynamic values such as feature names, row IDs, and class labels (denoted {*placeholder*}) are extracted from the original question via regex and re-injected verbatim at generation time.

Q-Type	Variant 1	Variant 2	Variant 3
Q1	What was the most influential feature driving the model's prediction?	Which feature played the biggest role in this prediction?	What did the model rely on most heavily to reach this conclusion?
Q2	Which input feature had the least effect on the model's decision?	Which feature mattered least to the model's prediction?	Which feature is least relevant to the model's conclusion?
Q3	What features were most critical in setting this prediction apart from the runner-up class?	Which features tipped the balance toward this prediction over the runner-up?	What distinguishes this prediction from the second most likely alternative?
Q4	What differences in the inputs account for the model giving instances A and B different predictions?	Why does the model give different predictions to these two instances?	What input differences lead to contrasting predictions for these two instances?
Q5	Would the model's output change if a specific feature were removed from this input?	If a particular feature were excluded, would the model's prediction remain the same?	How sensitive is the model's output to the presence of a specific feature?
Q6	What modifications to this instance would cause the model to output a different class?	How should this input be changed to flip the model's prediction to a different class?	What changes to this instance would lead the model to predict a different outcome?
Q7	How would the model's output be affected if {*feature*} were altered or removed?	What would happen to the model's prediction if {*feature*} were modified or excluded?	In what way would the model's output change if {*feature*} were no longer present?
Q10	What explains the model's success on {*correct instances*} but its failure on {*wrong instance*}?	Why does the model correctly classify {*correct instances*} but fail on {*wrong instance*}?	What distinguishes {*wrong instance*} from {*correct instances*} such that the model predicts it incorrectly?

baselines evaluated on the same vision feature-attribution task (Sensitivity had no evaluation data and is excluded). MEA is statistically equivalent to three gradient-based methods (IG, Guided Backprop, SmoothGrad) and significantly outperforms three others (SHAP, LIME, GradCAM).

Spurious-Feature Faithfulness (Q10). Table 9 reports Welch's t-test results for the spurious-feature question type (Q10) across text and vision modalities. MEA achieves **ceiling performance** (1.000 ± 0.000, $n = 160$) on both modalities–every sample is scored correctly. All Welch's t-tests return $p < 0.05$; however, this should be interpreted in light of the ceiling effect: when one group has zero variance, any non-zero difference in means will be flagged as significant regardless of its practical magnitude. For the strongest baselines the actual performance gap is negligible–Gemini-3.1 reaches 0.996 on text (gap: $+0.004$) and Base Agent reaches 0.981 on vision (gap: $+0.019$)–indicating that top-performing models already saturate Q10 and MEA's advantage is marginal. The significant p-values for weaker baselines (e.g. Claude-Haiku-4.5: 0.665 text, 0.855 vision) do reflect a meaningful improvement. CoT, ToT, and ReAct baselines had no Q10 evaluation data available.

Paraphrase Robustness Ablation. Table 10 presents paired t-tests comparing Paraphrase (mean of three paraphrase runs, matched on common sample IDs) against MEA. In 8 of 9 question-type$\times$modality conditions, there is no statistically significant difference, confirming that MEA's performance is robust to prompt paraphrasing. The single significant difference (vision counterfactual, $p = 0.026$) shows Paraphrase outperforming MEA by 0.075, suggesting that this question type may benefit from additional training diversity.

A.10 Computational Cost

Model architecture and parameters. MEA uses **Qwen3.6-35B-A3B** (Qwen Team, 2026) as its backbone, a Mixture-of-Experts multi-modal language model with **35 billion total parameters** and approximately **3.5 billion parameters activated per forward pass** (A3B). During inference for evaluation, we serve the model via LoRA adapters with rank 32, adding $< 1\%$ additional

Table 7: Output JSON schema of the critic agent.

Object	Field	Description
PROPOSER	analysis	Overall assessment of tool selection
	tool_analysis	Per-tool rating (*high/medium/low*) and recommendation (*keep/remove/replace*)
	tools_to_keep	Tools to retain in the revised strategy
	tools_to_remove	Tools to drop
	tools_to_add	New tools to include
	strategy_suggestions	Free-text suggestions to improve tool selection
ACTOR	analysis	Overall assessment of explanation quality
	identified_issues	Specific errors or weaknesses in the explanation
	region_feedback	Feedback on the accuracy of the identified region or feature
	explanation_feedback	Feedback on clarity and correctness
	improvement_suggestions	Free-text suggestions to improve the explanation

Table 8: Welch's t-test: MEA vs. tool-only XAI baselines (vision, feature attribution Q1). MEA mean $= 0.499 \pm 0.029$ ($n = 96$).

Baseline	n	Mean $\pm$ SEM	Δ (MEA $-$BL)	t	p	Sig.
SHAP	59	0.018 ± 0.007	$+0.481$	16.26	< 0.001	*
LIME	96	0.259 ± 0.027	$+0.240$	6.09	< 0.001	*
Integrated Gradients	**96**	$\mathbf{0.481 \pm 0.029}$	$\mathbf{+0.018}$	**0.44**	**0.661**	**n.s.**
Guided Backprop	**96**	$\mathbf{0.497 \pm 0.027}$	$\mathbf{+0.003}$	**0.07**	**0.944**	**n.s.**
SmoothGrad	**96**	$\mathbf{0.496 \pm 0.030}$	$\mathbf{+0.003}$	**0.08**	**0.935**	**n.s.**
GradCAM	96	0.131 ± 0.017	$+0.368$	10.95	< 0.001	*

parameters relative to the frozen base. Baseline comparisons using the un-finetuned backbone employed `Qwen3.6-35B-A3B`, while closed-source frontier baselines (Gemini-3.1-flash, GPT-5.4, Claude-Haiku-4.5) were accessed via their respective public APIs.

GRPO training. All reinforcement learning training was conducted through *Tinker* (Lab, 2025), a cloud-based training service that manages GPU allocation and distributed rollout execution internally; per-GPU-hour accounting is therefore not directly available from our training logs. The main hybrid training run ran for $\sim$93 GRPO steps with a group size of 128-256 rollouts per step processed per step, covering all three modalities jointly. Each step produced a mean episode length of ≈ 3.6 turns and ≈ 393 action tokens.

Post-hoc explainers. All post-hoc explainers and faithfulness metric computation, and ablations were run on a NVIDIA A100 GPUs (1 GPU per job, 16 CPU cores, 30 GB system RAM). Jobs were parallelized across datasets and question types (up to 4 concurrent jobs per modality). The full test-set evaluation for a single modality (3.5K instances across 4 dataset-model combinations and 10 question types) required approximately 8-12 A100 GPU-hours depending on modality complexity, with vision being the most expensive due to image preprocessing and SmoothGrad/GradCAM computation. Summing across the three modalities, primary test evaluations consumed an estimated 30-40 A100 GPU-hours. Ablation runs (prompt robustness, critic agent, OOD datasets) added approximately an additional 20-30 A100 GPU-hours. Total estimated evaluation compute is therefore on the order of 50-70 A100 GPU-hours.

CoT Actor Preamble

```
## Reasoning Mode: Chain-of-Thought
```

Before generating your explanation, reason through the evidence step-by-step:

1. **Understand the question**: What exactly is being asked?
2. **Analyze the XAI tool results**: What do the attribution maps, importance scores, and statistics tell us?
3. **Identify the key features/regions**: Based on the tool outputs, which specific parts of the input are most relevant?
4. **Synthesize your answer**: Combine the evidence into a clear, specific, and well-supported explanation.

Think step by step, then provide your final answer as valid JSON.

ReAct Actor Preamble

```
## Reasoning Mode: ReAct (Reasoning + Acting)
```

Interleave your reasoning with the evidence to build your explanation:

For each piece of evidence from the XAI tools:

Thought: What does this result tell us about the model's decision?
Observation: Note the key values, features, or regions highlighted.
Reflection: How does this connect to answering the question?

After processing all evidence:

Final Thought: Synthesize all observations into a coherent answer.

Then provide your final answer as valid JSON.

ToT Actor Preamble

```
## Reasoning Mode: Tree of Thought
```

Consider multiple possible interpretations of the evidence before settling on your answer:

Interpretation 1: What does the evidence suggest if you weight the highest-attribution features most heavily?
Interpretation 2: What alternative explanation could account for the same tool results?
Interpretation 3: Which interpretation is most consistent with ALL the evidence, not just the strongest signal?

Evaluate each interpretation against the evidence quality, specificity, and completeness. Select the strongest interpretation and provide your final answer as valid JSON.

Figure 9: Generic actor preambles prepended to the actor prompt for each reasoning baseline across all question types.

Table 9: Welch's t-test: MEA (1.000 ± 0.000, $n = 160$) vs. baselines on spurious-feature questions (Q10).

Modality	Baseline	n	Mean $\pm$ SEM	Δ (MEA $-$BL)	t	p	Sig.
Text	Base Agent	156	0.949 ± 0.017	$+0.051$	3.09	0.002	*
	GPT-5.4	157	0.733 ± 0.035	$+0.268$	7.67	< 0.001	*
	Claude-Haiku-4.5	158	0.665 ± 0.037	$+0.335$	9.11	< 0.001	*
	Gemini-3.1	160	0.996 ± 0.001	$+0.004$	2.92	0.004	*
Vision	Base Agent	148	0.981 ± 0.004	$+0.019$	5.15	< 0.001	*
	GPT-5.4	156	0.921 ± 0.006	$+0.079$	12.53	< 0.001	*
	Claude-Haiku-4.5	160	0.855 ± 0.011	$+0.145$	13.58	< 0.001	*
	Gemini-3.1	159	0.979 ± 0.003	$+0.021$	6.54	< 0.001	*

Table 10: Paired t-test: Paraphrase vs. MEA (matched common samples). Δ = Paraphrase − MEA.

Modality	Question Group	n_{pairs}	Paraphrase ($\mu\pm$ SEM)	MEA ($\mu\pm$ SEM)	Δ	t	p	Sig.
Tabular	Feature Attribution	106	0.683 ± 0.037	0.687 ± 0.037	-0.004	-0.63	0.532	n.s.
	Counterfactual	32	0.934 ± 0.023	0.939 ± 0.021	-0.005	-1.98	0.056	n.s.
	Spurious Features	17	1.000 ± 0.000	1.000 ± 0.000	0.000	–	–	n.s.
Text	Feature Attribution	126	0.738 ± 0.030	0.746 ± 0.031	-0.008	-0.83	0.407	n.s.
	Counterfactual	85	0.947 ± 0.012	0.952 ± 0.012	-0.005	-0.73	0.469	n.s.
	Spurious Features	37	1.000 ± 0.000	1.000 ± 0.000	0.000	–	–	n.s.
Vision	Feature Attribution	96	0.686 ± 0.034	0.693 ± 0.034	-0.007	-0.56	0.575	n.s.
	Counterfactual	**72**	$\mathbf{0.614 \pm 0.041}$	$\mathbf{0.539 \pm 0.044}$	$\mathbf{+0.075}$	**2.27**	**0.026**	*
	Spurious Features	40	1.000 ± 0.000	1.000 ± 0.000	0.000	–	–	n.s.

Note: "–" denotes that both conditions achieve identical ceiling performance (all scores = 1.0), making t-test inapplicable.

Critic Agent Prompt Template

System Role:

You are a critical analyst evaluating an XAI agent's strategy and explanation.

User Input:

Original Question
⟨*question text / example context*⟩

Question Context
Question Type: Q⟨*1-10*⟩
Modality: ⟨*tabular* | *text* | *vision*⟩

Strategy Used
Strategy Type: ⟨*tool-based*⟩
Selected Tools: ⟨*list of tool names*⟩
Strategy Reasoning: ⟨*Proposer's reasoning for tool selection*⟩

Generated Explanation
Output: ⟨*Actor's structured output dict*⟩
Explanation: ⟨*natural language explanation*⟩
Confidence: ⟨*Actor's self-reported confidence*⟩

Faithfulness Evaluation
Score: ⟨*faithfulness score*⟩
Metric: ⟨*metric name and formula*⟩
P_{original}: ⟨*value*⟩ P_{modified}: ⟨*value*⟩
⋮

Task Instruction: Analyze the strategy and results, then provide **two** separate JSON outputs:

1. `PROPOSER_REFLECTION`: feedback on tool selection strategy.
2. `ACTOR_REFLECTION`: feedback on explanation generation.

Consider: what went wrong, which tools were most/least helpful, how the strategy and explanation could be improved.

Figure 10: Complete prompt structure of the critic agent.

Actor Agent Prompt Template

System Role:

You are an XAI expert. Based on the provided analysis, ⟨*execute task-specific directive for Q1-Q10*⟩.

User Input:

Context & Prediction
Question (⟨*Q1-10*⟩): ⟨*specific question text*⟩
Model Prediction: Class: ⟨*predicted class*⟩
Conf: ⟨*score*⟩

Input Data
⟨*raw input text / tabular row / image data*⟩

XAI Analysis
Tool Results Summary:
⟨*status and most influential features per tool*⟩

Detailed Statistics:

- **Integrated Gradients**: ⟨*top positive/negative attributions*⟩
- **SHAP**: ⟨*top positive/negative shap values*⟩
- **Sensitivity**: ⟨*most sensitive prob_drops*⟩

Autonomous Reasoning Results:
GROUNDING: ⟨*extracted features matching query*⟩
REASONING: ⟨*semantic explanation for features*⟩
COMPARISON: ⟨*alignment with model prediction*⟩

Your Task
⟨*Specific directive corresponding to Question Q1-10*⟩

Required Output Format (JSON only)

```
{
"output": {
"⟨modality_key⟩": ⟨expected data structure⟩
},
"explanation": "2-3 sentences explaining your reasoning."
}
```

Critical Requirements

- **For vision:** `bounding_box` as ⟨*[x_min, y_min, x_max, y_max]*⟩
- **For text:** `spans` as a list of ⟨*{start_index, end_index}*⟩
- **For tabular:** `feature_keys` as a list of column names
- **OUTPUT SIZE CONSTRAINT:** ⟨*size limit dictated by question type*⟩
- Base your decision on the attribution analysis.

Final Instruction: Respond with ONLY JSON.

Figure 11: Prompt structure for the Actor Agent. Task directives, constraints, and JSON schema keys are populated dynamically at runtime based on the specific question type (Q1-Q10) and data modality.

Proposer Agent Prompt Template

System Role:

You are an AI explainability expert designing a strategy to answer a user's question about a machine learning model's prediction on ⟨*MODALITY*⟩.

User Input:

Question & Data Context
Question: ⟨*user question text*⟩
Input Content: ⟨*truncated input text, image, or tabular row*⟩

Model Information
Model: ⟨*model name*⟩ Arch: ⟨*architecture*⟩
Prediction: ⟨*class index & name*⟩ (Conf: ⟨*score*⟩)
Top-K Predictions: ⟨*list of dicts containing top probabilities*⟩

Available Methods

1. Autonomous Analysis: ⟨*instructions on using internal reasoning to ground/extract features*⟩

2. External XAI Tools:

- `lime`: ⟨*executes local surrogate explanation*⟩
- `integrated_gradients`: ⟨*computes path attribution*⟩
- `shap`: ⟨*computes shapley values*⟩
- `sensitivity_analysis`: ⟨*leave-one-out ablation*⟩
- `guided_backprop`: ⟨*modified ReLU gradients*⟩
- `smoothgrad`: ⟨*averaged noisy saliency*⟩

Your Task
⟨*Specific strategy design directive corresponding to Question Q1-10*⟩

Required Output Format (JSON only)

```
{
"strategy_type": "autonomous" | "tools" | "hybrid",
"reasoning": "Explain strategy choice (2-3 sentences)",
"autonomous_tasks": [
{
"task_type": "grounding" | "reasoning" | "comparison",
"query": "Specific query for autonomous analysis",
"expected_output": "What should be extracted"
}
],
"tool_selection": {
"selected_tools": ⟨subset of available tools⟩,
"reasoning": "Why these tools for this modality"
}
}
```

Final Instruction: Provide your strategy as a JSON object:

Figure 12: Prompt structure for the Proposer Agent. Task directives are populated dynamically based on the active question type (Q1-Q10), ensuring the strategy generated aligns with the specific objective.