MedProb: Probing Internal Representations of Vision-Language Models for Medical Question Answering

Anonymous ACL submission

Abstract

Medical visual question answering (Med-VQA) is often assumed to require medical fine-tuning, large models, or complex multi-agent pipelines. We revisit this assumption with **MedProb**, a lightweight probing framework that predicts multiple-choice Med-VQA answers from frozen VLM representations without free-text generation. Across PATH-VQA, SLAKE, and VQA-RAD, MedProb recovers substantially more answer-relevant signal than prompting and performs stronger than medical VLMs and agentic systems. Probing also reduces the apparent gap between small and large models compared to prompting, suggesting that smaller VLMs contain more recoverable Med-VQA signal than generation-based evaluation reveals. Across 14 matched general-purpose and medical VLM pairs, medical adaptation does not consistently improve this linear decodability. Finally, free-text generation exhibits an answer-position bias of up to 10 percentage points, whereas MedProb is more stable under option reordering. Code is available here: `https://anonymous.4open.science/r/MedProb1`

1 Introduction

Medical visual question answering (Med-VQA) asks models to answer clinical questions about medical images. Med-VQA is widely used to evaluate vision-language models (VLMs) in medical settings (OpenAI et al., 2023; Lu et al., 2024; Chen et al., 2024c) because clinical AI systems must connect visual evidence with medical language (Fries et al., 2022; Moor et al., 2023a). Recent work often improves Med-VQA by adding more resources to the model or inference pipeline. This includes domain-adaptive pretraining on biomedical image-text data (Gururangan et al., 2020; Chen et al., 2024a; Xie et al., 2024; Wu et al., 2025), multi-agent systems (Kim et al., 2024; Li et al., 2024), tool-augmented pipelines (Fallahpour et al., 2025), and large closed-source models that are costly and difficult to inspect in privacy-sensitive clinical settings (Marks and Haupt, 2023; Sellergren et al., 2026). These approaches differ in architecture and training, but they often share a common assumption that better Med-VQA requires larger models, more medical adaptation, or more complex inference.

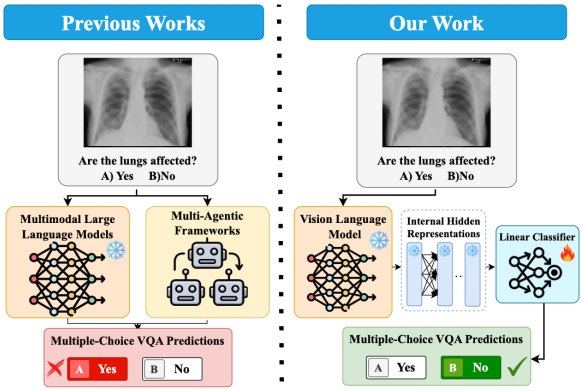

Figure 1: Prior methods (left) give a medical image and question to a VLM or multi-agent system, which then generates a free-text answer. This can lead to unsupported clinical reasoning. MedProb (right) uses the same input, but keeps the VLM frozen, extracts hidden representations from each layer, and predicts the multiple-choice answer with a linear classifier.

We ask the question, when a VLM gives the wrong answer, is the answer-relevant "signal" absent from its internal representations, or is it present but not reflected in the generated answer? Current Med-VQA benchmarks usually evaluate models through free-text generation, which mixes several factors, including what the model represents in their internal features, how the prompt is written, how answer options are ordered, and how the model expresses its answer in language. A model may represent useful information about a medical image but still choose the wrong answer because the prompt is brittle, the answer options appear in a different order, or the generated explanation introduces unsupported reasoning.

This distinction matters for how we interpret

model scale and medical adaptation. Under prompting, large VLMs often appear much stronger than small VLMs, and medically adapted models *may* appear stronger than their general-purpose bases. However, part of these gaps may reflect differences in generation behavior rather than differences in the answer-relevant information available in the hidden states. Prior work supports this concern. Medically adapted models do not always outperform general-domain models under controlled prompting (Jeong et al., 2024), clinical fine-tuning does not always make knowledge more accessible (Zhao et al., 2025), and generalist VLMs can match or outperform specialist medical models on clinical tasks (Zhong et al., 2026). Prompting itself is also unstable, with semantically equivalent prompts changing accuracy by up to 76 points (Sclar et al., 2024).

We address this gap with **MedProb** (Figure 1), a lightweight probing framework for Med-VQA. MedProb freezes the VLM, extracts hidden representations from each transformer layer, and trains a linear classifier to predict the correct multiple-choice answer. MedProb is not intended to replace clinical reasoning systems or provide a fully supervised Med-VQA solution. Instead, it tests whether answer-relevant information is linearly recoverable from frozen VLM representations, separate from the difficulties introduced by free-text generation.

Across PATH-VQA, SLAKE, and VQA-RAD, MedProb recovers substantially more answer-relevant signal than prompting. It also achieves competitive or higher closed-ended accuracy than several generation-based systems, including medically fine-tuned VLMs and multi-agent methods. We further find that the gap between small and large models is much smaller under probing than under prompting, suggesting that generation-based evaluation can underestimate the answer-relevant signal present in smaller models. Across 14 matched general-purpose and medically adapted VLM pairs, medical adaptation does not consistently improve the linear decodability of Med-VQA answer signal. Finally, free-text generation exhibits an answer-position bias of up to 10 percentage points, whereas MedProb is more stable under option reordering.

Our main contributions are as follows:

- We propose **MedProb**, a lightweight linear-probing framework that predicts multiple-choice Med-VQA answers from frozen VLM representations, separating linearly decodable answer signal from free-text generation.
- We show that generation-based evaluation can underestimate the answer-relevant signal in VLM representations. Across three Med-VQA benchmarks, MedProb recovers substantially more task-relevant signal than prompting, and the small-versus-large model gap is much smaller under probing.
- We analyze two evaluation confounds, domain adaptation and answer-option position. Across 14 matched model pairs, medical adaptation does not consistently improve the linear decodability of Med-VQA answer signal, and prompting accuracy changes by up to 10 percentage points depending on answer placement.

2 Related Work

Medical Vision-Language Models. Medical VLMs have developed along two trajectories: adapting general-purpose VLMs to the biomedical domain and building purpose-built systems. Early efforts include Med-Flamingo (Moor et al., 2023b), which extended OpenFlamingo (Awadalla et al., 2023) with few-shot medical reasoning, and LLaVA-Med (Li et al., 2023), which applied visual instruction tuning (Liu et al., 2023) to PubMed image-text pairs. Subsequent domain-adaptive pretraining (DAPT) (Gururangan et al., 2020) approaches include HuatuoGPT-Vision (Chen et al., 2024a) and MedTrinity-25M (Xie et al., 2024), while BioMed variants of InternVL3 (Zhu et al., 2025), Qwen2-VL (Wang et al., 2024), and Llama 3.2 (Grattafiori et al., 2024) follow the domain post-training recipe of Cheng et al. (2025). On the closed-source frontier, Med-Gemini (Yang et al., 2024) and GPT-4 leverage proprietary scale, while reasoning-augmented variants such as MedVL-Thinker (Huang et al., 2026) and MediX-R1 (Mullappilly et al., 2026) demand extensive training resources. Unlike these works, MedProb does not build a new medical VLM; instead, it probes frozen models and shows that medical adaptation does not consistently improve the linear decodability of answer-relevant signal in internal representations.

Prompt Sensitivity in Medical VLM Evaluation. A growing body of work questions the reliability of prompting-based evaluation. Sclar et al. (2024) showed that LLMs are highly sensitive to prompt formatting, with accuracy swings up to 76 points across semantically equivalent formats that persist even under scaling or instruction tuning. In

the medical domain, Jeong et al. (2024) found that domain-adaptive pretraining fails to consistently outperform general-domain baselines under controlled prompting. Pezeshkpour and Hruschka (2024) further demonstrated that LLM performance varies substantially with the ordering of answer options in multiple-choice settings, a positional bias orthogonal to model knowledge. Rather than documenting these instabilities, MedProb sidesteps them entirely by reading directly from frozen internal activations, yielding an evaluation paradigm that is inherently robust to both prompt formatting and option ordering.

Representation Probing in Language Models. Probing has emerged as a principled alternative to generation-based evaluation for diagnosing model knowledge. Maiya et al. (2025) showed that linear probes on frozen activations consistently outperform generation-based judgment across model families and datasets, generalizing under domain shifts. In the medical setting, Berkowitz et al. (2025b) demonstrated that linear classifiers on drug-name embeddings achieve AUC-ROC above 0.95 for adverse drug reaction classification without fine-tuning. Most directly relevant, the PING framework (Berkowitz et al., 2025a) recovers up to 87% of performance lost to safety alignment by probing frozen transformers on clinical QA, indicating that the primary bottleneck is generative rather than representational. MedProb extends this paradigm to the multimodal Med-VQA setting, providing the systematic comparison of probing across 14 matched general-purpose and medically adapted VLM pairs on three clinical benchmarks.

3 Methodology

We study Med-VQA as a multimodal multiple-choice classification task. Our goal is to compare what a VLM can produce through prompting with what can be read from its internal representations using simple linear probes. Figure 2 shows the full setup.

Problem Setup. Each Med-VQA example contains a medical image, a question, a set of answer options, and the correct answer. Let $\mathcal{D}_{\text{train}} = \{(I_i, q_i, \mathcal{O}_i, y_i)\}_{i=1}^{N_{\text{train}}}$ and $\mathcal{D}_{\text{test}} = \{(I_j, q_j, \mathcal{O}_j, y_j)\}_{j=1}^{N_{\text{test}}}$ denote the training and test splits. Here, I_i is the image, q_i is the question, $\mathcal{O}_i = \{o_1, \ldots, o_K\}$ is the set of candidate answers, and $y_i \in \mathcal{Y} = \{1, \ldots, K\}$ is the gold label. We use $K = 2$ for yes/no questions and $K = 4$ for non-binary multiple-choice questions.

Intuitively, prompting asks the VLM to read the image and question, consider the answer choices, and then generate the option it thinks is correct. Given a prompt construction strategy c, we build the multimodal prompt as $p_i = \Phi(I_i, q_i, \mathcal{O}_i; c)$, where Φ combines the image, question, and answer options using template c. The VLM f then produces a free-text response, and we extract the predicted option letter as $\hat{y}_i^{\text{gen}} = \text{Dec}(f, p_i)$, where $\text{Dec}(\cdot)$ maps the generated text to one of the answer choices.

MedProb: Layer-wise Representation Probing. MedProb tests whether the correct answer is already present in the frozen VLM representations, even when the model does not generate the correct answer. Rather than changing the VLM or adding external knowledge, we freeze the model and train simple classifiers on hidden states from each layer.

Intuitively, if a linear probe can predict the correct answer from a layer's hidden state, then that layer contains answer-relevant information in a form that is easy to read out. For each transformer layer $\ell \in \{0, \ldots, L\}$, we extract the hidden state of the last input token as a summary of the full multimodal context (Zhao et al., 2025). We denote this representation as $h_i^{(\ell)} = \psi_\ell(f, p_i) \in \mathbb{R}^{d_\ell}$.

For each layer, we train a separate multinomial logistic regression probe. The probe maps the representation to a distribution over answer choices as $g_\ell(h_i^{(\ell)}) = \text{softmax}(W^{(\ell)} h_i^{(\ell)} + b^{(\ell)})$, and predicts the answer with highest probability as $\hat{y}_i^{(\ell)} = \arg\max_{y \in \mathcal{Y}} [g_\ell(h_i^{(\ell)})]_y$. The parameters $W^{(\ell)} \in \mathbb{R}^{4 \times d_\ell}$ and $b^{(\ell)} \in \mathbb{R}^4$ are learned from the training representations $\{(h_i^{(\ell)}, y_i)\}_{i=1}^{N_{\text{train}}}$.

We standardize features using training-set statistics. We choose the regularization coefficient C with 5-fold cross-validation over a logarithmic grid. The VLM parameters are never updated. At inference time, we report the layer that performs best on a held-out validation split, which is separate from the test set. This layer-wise setup lets us ask two questions. First, does the frozen VLM encode enough information to identify the correct answer? Second, where in the network does that information become linearly decodable?

4 Experiments

Models. To the best of our knowledge, we include all publicly available VLMs, both general-purpose and medically specialized, released up

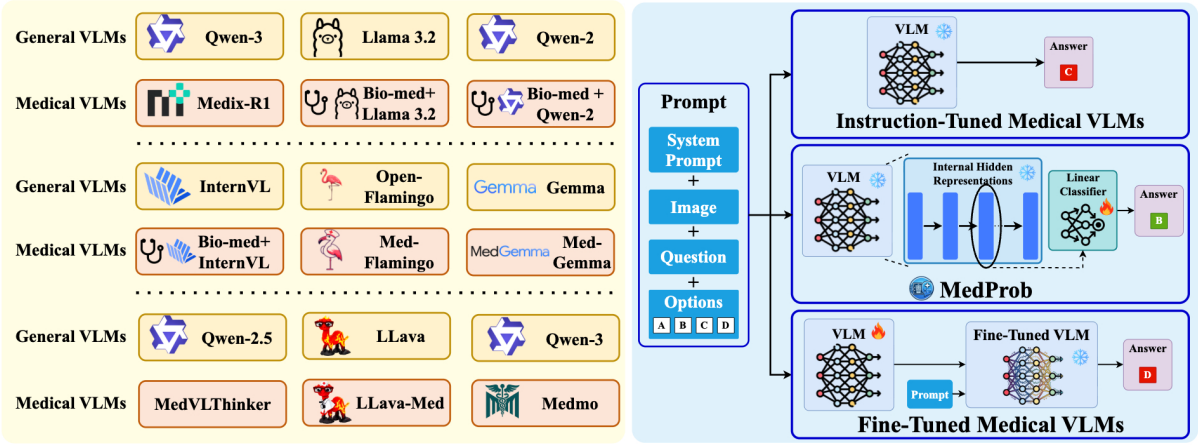

Figure 2: Overview of the MedProb evaluation framework. **Left**: general-purpose and medical VLMs evaluated, grouped by backbone architecture. **Right**: prompts consist of a system prompt, medical image, clinical question, and multiple-choice options (A–D), evaluated under three settings. *Top*: instruction-tuned VLMs are prompted directly (frozen). *Middle*: MedProb extracts hidden representations from each transformer layer and decodes answers via a linear classifier (frozen). *Bottom*: fine-tuned VLMs are trained on medical data, then prompted (frozen).

to March 2026 (Figure 2, left). General-purpose models span: LLaVA (Liu et al., 2023), OpenFlamingo (Awadalla et al., 2023), InternVL3-1B (Zhu et al., 2025), Llama-3.2-11B-Vision (Grattafiori et al., 2024), Qwen2-VL-2B (Wang et al., 2024), Qwen2.5-VL (3B, 7B, 32B) (Bai et al., 2025b), Qwen3-VL (2B, 4B, 8B, 30B) (Bai et al., 2025a), and Gemma (4B, 27B) (Team et al., 2025). Their domain-adapted counterparts include: LLaVA-Med-7B (Li et al., 2023), Med-Flamingo-9B (Moor et al., 2023b), BioMed-InternVL3-1B, BioMed-Llama-3.2-11B, BioMed-Qwen2-VL-2B (Cheng et al., 2025), MedVLThinker-RL (3B, 7B, 32B) (Huang et al., 2026), MedMO (4B, 8B) (Deria et al., 2026), MediX-R1 (2B, 30B) (Mullappilly et al., 2026), and MedGemma (4B, 27B) (Sellergren et al., 2026). A complete listing with medical training datasets is in Table 5. We compare only matched medical and general-purpose models with the same backbone and vision encoder, excluding unmatched cases (e.g., LLaVA-Med v1.5 vs. Mistral-7B) to isolate domain adaptation.

Datasets. We evaluate on VQA-RAD (Lau et al., 2018), PathVQA (He et al., 2020), and SLAKE (Liu et al., 2021). Following Jeong et al. (2024), we restrict to closed-ended questions, convert binary yes/no questions to multiple-choice format, and construct plausible distractors for non-binary questions; samples with missing images or training-set leakage are removed. PathVQA is subsampled to 500 examples via stratified sampling; VQA-RAD and SLAKE use full test sets. Full dataset statistics and preprocessing details are in Appendix A. For broader cross-modal generalization, we additionally evaluate on OmniMed-VQA (Hu et al., 2024) across eight imaging modalities (Section F).

Baselines. We compare MedProb against three categories of baselines, corresponding to the three settings in Figure 2 (right). **Agentic frameworks**: MAM (Zhou et al., 2025), MMedAgent (Li et al., 2024), and UCAgents (Qianhan Feng, 2025) orchestrate multiple models or tools through multi-step reasoning pipelines. **Instruction-tuned medical VLMs** (frozen, directly prompted): HuatuoGPT-Vision-34B (Chen et al., 2024a), Aloe-Vision-72B-AR (Guasch-Martí et al., 2026), and UniMedVL-14B (Ning et al., 2025), instruction-tuned on large-scale biomedical data to generalize to unseen tasks. **Benchmark-trained medical VLMs** (instruction-tuned on benchmark data, then prompted frozen): InfiMed-RL-3B (Liu et al., 2025b), MedVLM-R1 (Pan et al., 2025), MedMO (4B, 8B) (Deria et al., 2026), and MediX-R1 (2B, 30B) (Mullappilly et al., 2026). These models are instruction-tuned with access to training splits of our evaluation benchmarks; they are included as upper-bound references for supervised specialization. The effect of fine-tuning on the probing gap is analyzed separately in Section 5. None of the baseline medical VLMs share backbone architecture with our probed models; they are included to situate MedProb within the broader landscape of specialized systems. All baselines are re-evaluated from scratch on our identical test

Model	Method	PATH-VQA		SLAKE		VQA-RAD		Average	
		F1	Acc	F1	Acc	F1	Acc	F1	Acc
MAM (Zhou et al., 2025)	Agentic	51.85	51.80	39.92	59.51	39.31	59.31	39.62	56.87
MMedAgent (Li et al., 2024)	Agentic	36.11	37.00	28.30	43.13	59.80	58.17	41.40	46.10
UCAgents (Qianhan Feng, 2025)	Agentic	60.55	61.19	78.09	77.10	64.17	71.10	62.36	69.80
HuatuoGPT-Vision-34B (Chen et al., 2024a)	Medical VLM	60.61	61.80	77.89	76.86	83.57	75.66	74.02	71.44
Aloe-Vision-72B-AR (Guasch-Martí et al., 2026)	Medical VLM	67.70	69.20	86.80	80.24	75.73	79.46	76.74	76.30
UniMedVL-14B (Ning et al., 2025)	Medical VLM	52.84	48.00	66.46	58.07	66.24	58.94	61.85	55.00
MedVLM-R1-2B (Pan et al., 2025)	Medical VLM†	46.73	58.00	49.29	61.20	58.71	51.71	51.58	56.97
InfiMed-RL-3B (Liu et al., 2025b)	Medical VLM†	79.79	79.60	88.16	81.92	83.43	74.90	83.79	78.81
MedMo-8B (Deria et al., 2026)	Medical VLM†	65.99	67.80	80.64	72.29	73.39	58.94	73.34	66.34
Medix-R1-30B (Mullappilly et al., 2026)	Medical VLM†	73.00	73.00	82.46	83.61	70.32	72.24	75.26	76.28
InternVL3-1B (Zhu et al., 2025)	Prompting	62.01	62.80	83.68	75.66	76.12	65.02	73.94	67.83
InternVL3-1B (Zhu et al., 2025)	MedProb	82.09	82.20	88.70	83.13	81.51	72.62	84.10	79.32
Qwen3-VL-2B (Bai et al., 2025a)	Prompting	55.78	56.80	68.73	77.83	64.72	64.64	63.08	66.42
Qwen3-VL-2B (Bai et al., 2025a)	MedProb	83.67	83.80	89.02	84.10	80.79	73.00	84.49	80.30
Gemma-4B (Team et al., 2025)	Prompting	53.94	57.00	68.93	61.69	57.96	52.47	60.28	57.05
Gemma-4B (Team et al., 2025)	MedProb	86.33	86.40	86.43	79.76	80.33	72.24	84.36	79.47
Qwen3-VL-4B (Bai et al., 2025a)	Prompting	59.88	54.60	86.03	77.35	46.24	67.30	64.05	66.42
Qwen3-VL-4B (Bai et al., 2025a)	MedProb	85.69	85.80	90.63	86.02	83.14	74.90	86.49	82.24
Qwen3-VL-8B (Bai et al., 2025a)	Prompting	58.98	58.60	70.36	76.63	46.25	69.58	58.53	68.27
Qwen3-VL-8B (Bai et al., 2025a)	MedProb	86.12	86.20	89.35	84.09	82.34	75.28	85.93	81.86
Llama-3.2-11B-Vision (Grattafiori et al., 2024)	Prompting	48.94	59.80	68.43	69.88	67.60	68.06	61.66	65.91
Llama-3.2-11B-Vision (Grattafiori et al., 2024)	MedProb	86.12	86.20	88.21	82.40	83.89	76.04	86.07	81.55
Gemma-27B (Team et al., 2025)	Prompting	61.18	62.00	73.12	75.42	69.56	69.58	67.95	69.00
Gemma-27B (Team et al., 2025)	MedProb	**86.86**	**87.00**	89.17	82.89	83.14	76.04	86.39	81.98
Qwen3-VL-30B (Bai et al., 2025a)	Prompting	63.23	61.20	88.04	80.96	74.53	77.95	75.27	73.37
Qwen3-VL-30B (Bai et al., 2025a)	MedProb	85.70	85.80	**91.12**	**86.75**	**86.46**	**81.37**	**87.76**	**84.64**

Table 1: Performance comparison on medical VQA benchmarks. **Bold**: best per column; MedProb rows are highlighted in blue; †Trained on PATH-VQA, VQA-RAD, and SLAKE.

splits; inference configurations are in Appendix I.

MedProb uses up to 1,000 labeled training examples to fit the linear probe, while prompting baselines are zero-shot. This is intentional: our central claim is that even this minimal supervised elicitation, a single logistic regression layer with no VLM parameter updates, consistently outperforms zero-shot prompting and extensively fine-tuned medical VLMs, cleanly separating *knowledge representation* from *knowledge elicitation*.

Evaluation Metrics. Following Jeong et al. (2024), we report macro-averaged F1 and accuracy. F1 is computed over two classes for binary questions and macro-averaged across all answer categories for non-binary questions. Prompting baselines use zero-shot evaluation; MedProb trains a linear probe on the training split without updating any VLM parameters.

Implementation Details. We extract the hidden state of the *last input token* at every transformer layer of the frozen VLM. Features are standardized using training-set mean and variance. An independent multinomial logistic regression probe is trained per layer, with regularization coefficient C selected via 5-fold cross-validation over $\{10^{-4}, \ldots, 10^{4}\}$ (max 1,000 iterations per fold). The best-performing layer is selected on a held-out validation split strictly disjoint from the test set. For the prompting baseline, we follow each model's official evaluation protocol; full details are in Appendix C.

5 Results and Analysis

Main Results. Table 1 presents a comprehensive comparison of MedProb against agentic frameworks and medical VLM baselines across all three benchmarks. MedProb consistently and substantially outperforms both categories. Among agentic systems, UCAgents achieves the strongest average accuracy at 69.80%, while MMedAgent trails significantly at 46.10%. Medical VLMs fare better: InfiMed-RL-3B reaches an average F1 of 83.79% and Aloe-Vision-72B-AR achieves 76.74%; yet

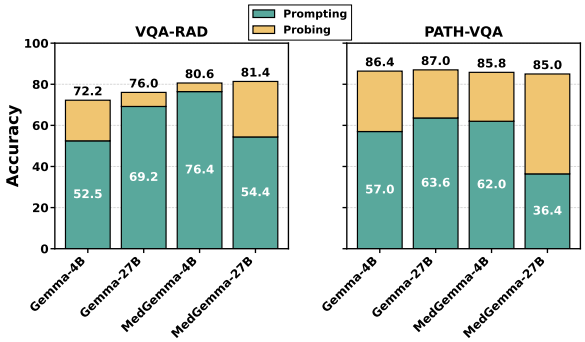

Figure 3: Prompting vs. MedProb accuracy for general (Gemma-4B/27B) and medical (MedGemma-4B/27B) VLMs on VQA-RAD and PATH-VQA.

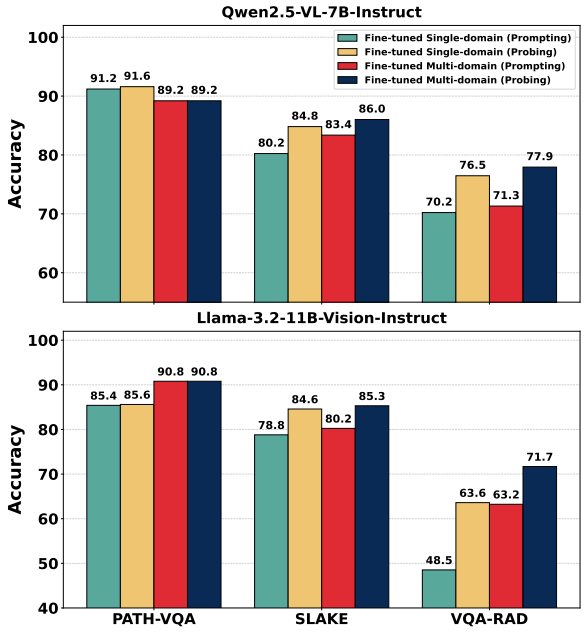

Figure 4: Probing vs. prompting accuracy after single-domain and multi-domain fine-tuning.

even these large-scale, instruction-tuned systems fall short of what linear probing extracts from frozen general-purpose VLMs. Notably, the smallest probing model, InternVL3-1B, already surpasses most agentic and several medical VLM baselines with an average accuracy of 79.32%, despite having far fewer parameters and no domain-specific training. To rule out generation artifacts as the explanation for the prompting gap, we additionally evaluate option-likelihood scoring and conduct paired McNemar significance tests; full results and analysis are in Appendix E. Few-shot (3-shot) prompting results are provided in Appendix D.

The probing results also reveal a clear scaling trend: larger models encode more linearly decodable clinical knowledge. Within the Qwen3-VL family, average accuracy grows from 80.30% at 2B to 82.24% at 4B and 84.64% at 30B. The best overall results are achieved by Qwen2.5-VL-32B, which sets the top scores across SLAKE (F1: 91.60, Acc: 87.46), VQA-RAD (F1: 92.33, Acc: 88.59), and the overall average (F1: 89.94, Acc: 87.35). Similarly, within the Gemma family, scaling from 4B to 27B improves PATH-VQA F1 from 86.33% to 86.86%.

Figure 3 directly contrasts probing and prompting for four representative general-purpose models on VQA-RAD and PATH-VQA. On VQA-RAD, the probing advantage ranges from a clear gain for Gemma-27B (76.0% vs. 69.2%) to a pronounced improvement for Gemma-4B (72.2% vs. 52.5%). The gap is far more dramatic on PATH-VQA, where prompting accuracy collapses to between 57.2% and 63.6% for all four models, while MedProb consistently achieves 82.8%–87.0% for the same models, improvements exceeding 20 percentage points in several cases. This stark contrast on a pathology-focused benchmark strongly indicates that the clinical knowledge needed to interpret pathological images is richly encoded in these models' internal representations, yet systematically suppressed by the generation process.

Effect of Fine-tuning on the Probing Gap. Figure 4 examines whether fine-tuning the underlying VLM narrows the gap between prompting and probing. Fine-tuning the base VLM does not close the probing gap. After in-domain fine-tuning, probing the fine-tuned model still outperforms prompting it on SLAKE (84.7% vs. 79.5%) and VQA-RAD (70.3% vs. 59.5%), with PATH-VQA the sole exception where binary yes/no task structure allows prompted generation to match the probe. Under multi-domain fine-tuning the same pattern holds, and fine-tuning yields only marginal gains for prompting across most settings, confirming that the elicitation mechanism, not the encoded knowledge, remains the bottleneck.

Layer-wise Probing Analysis. Figure 5 shows that probing accuracy rises monotonically with layer depth. Both models start near chance at layer 0 (≈50.4%) and improve steadily, with the most clinically informative signal concentrated in the deeper layers: Qwen2.5-VL-7B peaks at 80.4% at layer 28 and MedVLThinker-7B at 79.5% at layer 22, after which the medical model's accuracy declines. MedVLThinker-7B holds a slight edge in early-to-middle layers (71.4% vs. 69.1% at layer 8), but the general model overtakes it from

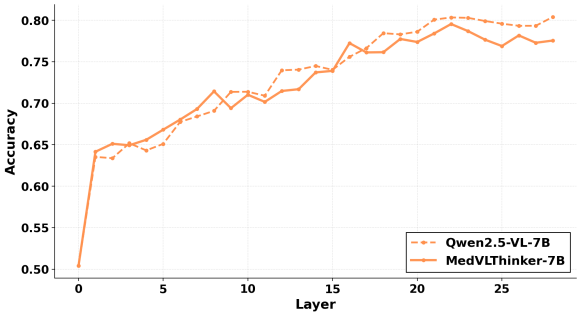

Figure 5: Layer-wise probing accuracy for Qwen2.5-VL-7B and MedVLThinker-7B.

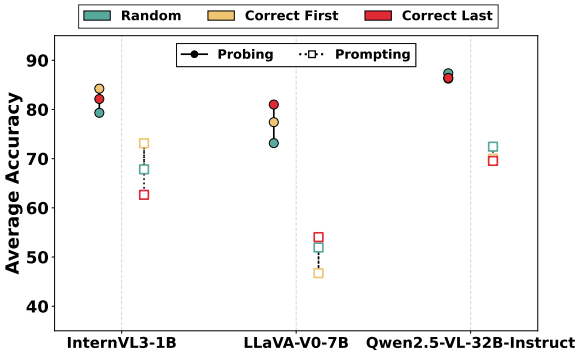

Figure 6: Probing vs. prompting accuracy under random, correct-first, and correct-last answer orderings.

layer 9 onward, indicating that medical adaptation accelerates early-layer encoding but fails to produce richer representations in the deeper layers that matter most.

Robustness to Answer Option Ordering. A critical concern with prompting-based evaluation is positional bias (Pezeshkpour and Hruschka, 2024). Figure 6 reports accuracy for InternVL3-1B, LLaVA-V0-7B, and Qwen2.5-VL-32B under three conditions: random (standard balanced evaluation), correct-answer first, and correct-answer last.

MedProb is far less sensitive to answer position than prompting. The first-vs-last accuracy gap for probing is 2.10 pp (InternVL3-1B), 3.58 pp (LLaVA-V0-7B), and 0.14 pp (Qwen2.5-VL-32B), compared to 10.49 pp, 7.33 pp, and 0.44 pp respectively under prompting. Prompting further exhibits inconsistent, model-dependent biases: LLaVA-V0-7B favors correct-last (recency effect, +7.33 pp), while InternVL3-1B favors correct-first (primacy effect, +10.49 pp). The random condition, which averages over balanced orderings and reflects standard reported results, falls within the prompting range but does not resolve these instabilities. Because MedProb reads directly from internal activations without generating text, it sidesteps positional bias entirely.

General Model	Medical Model	Δ
Gemma-4B	MedGemma-4B	−4.92
Gemma-27B	MedGemma-27B	−0.95
Open-Flamingo-9B	Med-Flamingo-9B	−1.93
InternVL3-1B	BioMed-InternVL3-1B	+2.36
LLaVA-7B	LLaVA-Med-7B	+0.17
LLaMA3.2-11B-Vision	BioMed-LLaMA3.2-11B	−2.30
Qwen2-VL-2B	BioMed-Qwen2-VL-2B	+0.05
Qwen2.5-VL-3B	MedVLThinker-RL-3B	+2.16
Qwen2.5-VL-7B	MedVLThinker-7B	+1.15
Qwen2.5-VL-32B	MedVLThinker-RL-32B	+1.65
Qwen3-VL-2B	MediX-R1-2B	+11.31
Qwen3-VL-4B	MedMo-4B	−1.65
Qwen3-VL-8B	MedMo-8B	−0.19
Qwen3-VL-30B	MediX-R1-30B	+3.72

Table 2: Probing accuracy delta (Δ) between matched general/medical pairs (averaged across benchmarks). Positive: medical better; negative: general better.

General vs. Medical Model Comparison. Table 2 reports the probing accuracy delta (Δ) between each general-purpose model and its medically adapted counterpart.

There is no consistent probing improvement from general to medical models. Out of 14 matched pairs, 6 show the general model outperforming its medical counterpart, and medical instruction-tuning can actively degrade representations: MedGemma-4B trails Gemma-4B by 4.92 pp, BioMed-LLaMA3.2-11B trails its base by 2.30 pp, and where adaptation does help, gains are often negligible (LLaVA-Med: +0.17 pp; BioMed-Qwen2-VL-2B: +0.05 pp). The exception is MediX-R1, which uses RL-based adaptation rather than standard SFT and outperforms its base by 11.31 pp (2B) and 3.72 pp (30B), suggesting RL training may more effectively shape internal clinical representations.

Error Analysis. To understand where MedProb succeeds and where it falls short relative to prompting, we conducted a targeted qualitative analysis over a stratified sample of model outputs. We selected 10 instances from each of the three evaluation benchmarks, yielding 30 samples per model, across all 10 evaluated models, for a total of 300 manually examined responses. For each sample, we inspected the model's full generated output alongside the probe's prediction and the ground-truth label, classifying each case into one of three behavioral patterns. A detailed presentation of representative examples, including verbatim model outputs, is provided in Appendix H.

Prompt model output (idx 440)

"Based on the image, the answer is **A: yes**. The image shows a lobulated mass with a bluish cartilaginous hue infiltrating the soft tissues. This is a characteristic appearance of a cartilaginous tumor."

Prompt prediction: ***A (Yes)*** × *Probe prediction:* ***B (No)*** ✓ *GT:* ***B (No)***

Figure 7: Semantic hallucination failure: prompted model identifies real visual features but draws a false clinical conclusion; probe recovers the correct answer from latent activations.

Where does MedProb improve over prompting? MedProb improves performance primarily on visually grounded binary classification tasks where free-form generation introduces hallucinated medical reasoning. In these cases, internal representations encode the correct answer signal, but language generation leads the model to over-interpret image features and reason itself into the wrong answer. This failure pattern recurs consistently across modalities: in radiology, the prompted model applies textbook logic correctly but misattributes it to the wrong scan; in histopathology, it generates plausible-sounding but incorrect tissue-level narratives. Because the linear probe reads directly from latent activations without generating a reasoning chain, it sidesteps this "hallucinated chain-of-thought" failure mode. MedProb helps most on questions requiring straightforward visual feature matching, organ presence, gross pathological category, imaging modality, where intermediate-layer activations already encode the correct signal but the output layer is misled by overconfident medical language generation. Figure 7 illustrates the most common failure mode corrected by the probe.

As shown in Figure 7, the prompted model correctly identifies real visual features (lobulated shape, cartilaginous hue) but constructs a false pathology narrative from them and answers *yes*; the probe at layer 16 reads directly from latent activations and is unaffected by this hallucinated reasoning chain.

When both methods fail simultaneously, the deficit lies in the model's perceptual encoding rather than the elicitation strategy; when prompting outperforms probing, the task demands compositional clinical reasoning or grounded semantic labels that exceed what a linear probe can decode from a single layer (see Appendix H for detailed examples of both patterns).

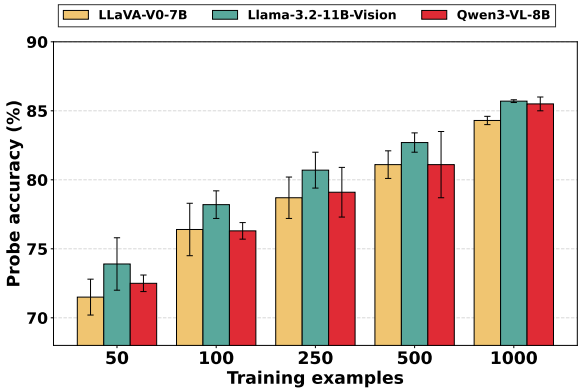

Figure 8: PATH-VQA probe accuracy (%) vs. training-set size (mean ± std over different random seeds).

Sample Efficiency. We measured how probe accuracy scales with labeled training data by retraining on progressively larger subsets (50, 100, 250, 500, 1,000 examples), averaging over different random seeds. Figure 8 shows that all three models improve monotonically, gaining approximately 12 percentage points from 50 to 1,000 training examples. Crucially, even at 50 examples all three models exceed 71% accuracy, establishing that strong clinical decoding is achievable with limited supervision, a practically important property for low-resource medical settings. Gains diminish above 500 examples, suggesting that 500–1,000 labeled samples is a sufficient operating range for MedProb.

6 Conclusion

We introduced MedProb, a lightweight linear probing framework that decodes clinical answers directly from the frozen internal representations of general-purpose VLMs, outperforming medical VLMs and multi-agent pipelines across PATH-VQA, SLAKE, and VQA-RAD with only a logistic regression classifier trained on a small labeled set. Across 14 matched architecture pairs, domain-adaptive pretraining does not reliably enrich internal clinical representations, with general models outperforming their medical counterparts in 6 of 14 cases and RL-based adaptation (MediX-R1) the notable exception. MedProb is also substantially more robust to answer option ordering than prompting, which exhibits inconsistent positional biases across models. Together, these findings reframe the Med-VQA challenge: the bottleneck is not what VLMs know, but how that knowledge is elicited. Future work should explore multi-layer aggregation, nonlinear probes, and open-ended VQA settings to further unlock the clinical knowledge latent in general-purpose representations.

Limitations

This work has several limitations. First, MedProb requires a small labeled training set, while prompted VLMs can be evaluated zero-shot. This means MedProb should not be interpreted as a purely zero-shot method. However, our goal is not to compare zero-shot prompting with supervised learning in general, but to test whether answer-relevant information is already present in frozen VLM representations and can be recovered with minimal supervision. The probe is intentionally simple, updates no VLM parameters, and uses only a lightweight linear classifier.

Second, our experiments focus primarily on closed-ended Med-VQA benchmarks. This setting allows controlled comparison across models and avoids ambiguity in free-text answer matching, but it does not capture the full complexity of open-ended clinical reasoning. We therefore view MedProb as a diagnostic tool for measuring recoverable answer signal, not as a complete clinical question-answering system. Extending the approach to open-ended VQA, report generation, and more complex clinical workflows is an important direction for future work.

Third, linear probing measures what is linearly decodable from a chosen representation, not everything a model may know. A failure of the probe does not necessarily mean that the information is absent from the model; it may be encoded non-linearly, distributed across layers, or dependent on richer aggregation strategies. This limitation makes our results conservative: despite using a simple probe, MedProb still recovers substantially more task-relevant signal than prompting across benchmarks.

Finally, MedProb improves answer prediction but does not itself provide faithful clinical explanations or evidence grounding. This is by design. Our central claim is that generation-based evaluation can underestimate the medical VQA signal present in VLM representations. For deployment-facing systems, probing would need to be combined with calibrated uncertainty estimates, evidence grounding, and clinical validation. These limitations do not change the main finding: much of the answer-relevant signal needed for Med-VQA is already present in frozen VLMs, and the main bottleneck is often how that signal is elicited.

References

Anas Awadalla, Irena Gao, Josh Gardner, Jack Hessel, Yusuf Hanafy, Wanrong Zhu, Kalyani Marathe, Yonatan Bitton, Samir Gadre, Shiori Sagawa, Jenia Jitsev, Simon Kornblith, Pang Wei Koh, Gabriel Ilharco, Mitchell Wortsman, and Ludwig Schmidt. 2023. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, and 45 others. 2025a. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025b. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Jacob Berkowitz, Sophia Kivelson, Apoorva Srinivasan, Undina Gisladottir, Kevin K. Tsang, Jose Miguel Acitores Cortina, Aditi Kuchi, Jake Patock, Ryan Czarny, and Nicholas P. Tatonetti. 2025a. Probing hidden states for calibrated, alignment-resistant predictions in llms. *medRxiv*.

Jacob Berkowitz, Davy Weissenbacher, Apoorva Srinivasan, Nadine A. Friedrich, Jose Miguel Acitores Cortina, Sophia Kivelson, Graciela Gonzalez Hernandez, and Nicholas P. Tatonetti. 2025b. Probing large language model hidden states for adverse drug reaction knowledge. In *Artificial Intelligence in Medicine: 23rd International Conference, AIME 2025, Pavia, Italy, June 23–26, 2025, Proceedings, Part I*, page 55–64, Berlin, Heidelberg. Springer-Verlag.

Junying Chen, Chi Gui, Ruyi Ouyang, Anningzhe Gao, Shunian Chen, Guiming Hardy Chen, Xidong Wang, Zhenyang Cai, Ke Ji, Xiang Wan, and Benyou Wang. 2024a. Towards injecting medical visual knowledge into multimodal LLMs at scale. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7346–7370, Miami, Florida, USA. Association for Computational Linguistics.

Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, and Feng Zhao. 2024b. Are we on the right way for evaluating large vision-language models? In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA. Curran Associates Inc.

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang,

Xizhou Zhu, Lewei Lu, and 1 others. 2024c. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198.

Daixuan Cheng, Shaohan Huang, Ziyu Zhu, Xintong Zhang, Wayne Xin Zhao, Zhongzhi Luan, Bo Dai, and Zhenliang Zhang. 2025. On domain-adaptive post-training for multimodal large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 274–296, Suzhou, China. Association for Computational Linguistics.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Ankan Deria, Komal Kumar, Adinath Madhavrao Dukre, Eran Segal, Salman Khan, and Imran Razzak. 2026. Medmo: Grounding and understanding multimodal large language model for medical images. *Preprint*, arXiv:2602.06965.

Adibvafa Fallahpour, Jun Ma, Alif Munim, Hongwei Lyu, and Bo Wang. 2025. Medrax: Medical reasoning agent for chest x-ray. *Preprint*, arXiv:2502.02673.

Jason Fries, Ethan Steinberg, Scott Fleming, Michael Wornow, Yizhe Xu, Keith Morse, Dev Dash, and Nigam Shah. 2022. How foundation models can advance ai in healthcare.

Nauman Ullah Gilal, Rachida Zegour, Khaled Al-Thelaya, Erdener Özer, Marco Agus, Jens Schneider, and Sabri Boughorbel. 2025. Pathvlm-eval: Evaluation of open vision language models in histopathology. *Journal of Pathology Informatics*, 18:100455.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Jaume Guasch-Martí, Enrique Lopez-Cuena, Martín Suárez-Fernández, Jordi Bayarri-Planas, Anna Arias-Duart, and Dario Garcia-Gasulla. 2026. Aloe-vision: Robust vision-language models for healthcare. In *Proceedings of The 9th International Conference on Medical Imaging with Deep Learning*, volume 315 of *Proceedings of Machine Learning Research*, pages 2404–2426. PMLR.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Xuehai He, Yichen Zhang, Luntian Mou, Eric Xing, and Pengtao Xie. 2020. Pathvqa: 30000+ questions for medical visual question answering. *arXiv preprint arXiv:2003.10286*.

Yutao Hu, Tianbin Li, Quanfeng Lu, Wenqi Shao, Junjun He, Yu Qiao, and Ping Luo. 2024. Omnimed-vqa: A new large-scale comprehensive evaluation benchmark for medical lvlm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22170–22183.

Xiaoke Huang, Juncheng Wu, Hui Liu, Xianfeng Tang, and Yuyin Zhou. 2026. Medvlthinker: Simple baselines for multimodal medical reasoning. *Preprint*, arXiv:2508.02669.

Daniel P Jeong, Saurabh Garg, Zachary Chase Lipton, and Michael Oberst. 2024. Medical adaptation of large language and vision-language models: Are we making progress? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12143–12170, Miami, Florida, USA. Association for Computational Linguistics.

Yubin Kim, Chanwoo Park, Hyewon Jeong, Yik Siu Chan, Xuhai Xu, Daniel McDuff, Hyeonhoon Lee, Marzyeh Ghassemi, Cynthia Breazeal, and Hae Won Park. 2024. Mdagents: an adaptive collaboration of llms for medical decision-making. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, NIPS '24, Red Hook, NY, USA. Curran Associates Inc.

Jason J Lau, Soumya Gayen, Asma Ben Abacha, and Dina Demner-Fushman. 2018. A dataset of clinically generated visual questions and answers about radiology images. *Scientific Data*, 5:180251.

Binxu Li, Tiankai Yan, Yuanting Pan, Jie Luo, Ruiyang Ji, Jiayuan Ding, Zhe Xu, Shilong Liu, Haoyu Dong, Zihao Lin, and Yixin Wang. 2024. MMedAgent: Learning to use medical tools with multi-modal agent. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 8745–8760, Miami, Florida, USA. Association for Computational Linguistics.

Chunyuan Li, Cliff Wong, Sheng Zhang, Naoto Usuyama, Haotian Liu, Jianwei Yang, Tristan Naumann, Hoifung Poon, and Jianfeng Gao. 2023. Llava-med: Training a large language-and-vision assistant for biomedicine in one day. In *Advances in Neural Information Processing Systems*, volume 36, pages 28541–28564. Curran Associates, Inc.

Bo Liu, Li-Ming Zhan, Li Xu, Lin Ma, Yan Yang, and Xiao-Ming Wu. 2021. Slake: A semantically-labeled knowledge-enhanced dataset for medical visual question answering. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 1650–1654.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pages 34892–34916. Curran Associates, Inc.

Hongjun Liu, Yinghao Zhu, Yuhui Wang, Yitao Long, Zeyu Lai, Lequan Yu, and Chen Zhao. 2025a. Medmmv: A controllable multimodal multi-agent framework for reliable and verifiable clinical reasoning. *Preprint*, arXiv:2509.24314.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. 2024. Mmbench: Is your multi-modal model annbsp;all-around player? In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part VI*, page 216–233, Berlin, Heidelberg. Springer-Verlag.

Zeyu Liu, Zhitian Hou, Guanghao Zhu, Zhijie Sang, Congkai Xie, and Hongxia Yang. 2025b. Infimed: Low-resource medical mllms with advancing understanding and reasoning. *Preprint*, arXiv:2505.23867.

Haoyu Lu, Wen Liu, Bo Zhang, Bingxuan Wang, Kai Dong, Bo Liu, Jingxiang Sun, Tongzheng Ren, Zhuoshu Li, Hao Yang, Yaofeng Sun, Chengqi Deng, Hanwei Xu, Zhenda Xie, and Chong Ruan. 2024. Deepseek-vl: Towards real-world vision-language understanding. *Preprint*, arXiv:2403.05525.

Sharan Maiya, Yinhong Liu, Ramit Debnath, and Anna Korhonen. 2025. Improving preference extraction in LLMs by identifying latent knowledge through classifying probes. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9061–9081, Vienna, Austria. Association for Computational Linguistics.

Mason Marks and Claudia E Haupt. 2023. Ai chatbots, health privacy, and challenges to hipaa compliance. *JAMA*, 330(4):309–310.

Meta AI. 2024. Llama 3.2: Model cards and prompt formats. Accessed: 2026-03-25.

Pranav Pushkar Mishra, Mohammad Arvan, and Mohan Zalake. 2026. Teammedagents: Pareto-efficient multi-agent medical reasoning through teamwork theory. *Preprint*, arXiv:2508.08115.

Michael Moor, Oishi Banerjee, Zahra Shakeri Hossein Abad, Harlan M. Krumholz, Jure Leskovec, Eric J. Topol, and Pranav Rajpurkar. 2023a. Foundation models for generalist medical artificial intelligence. *Nature*, 616(7956):259–265.

Michael Moor, Qian Huang, Shirley Wu, Michihiro Yasunaga, Yash Dalmia, Jure Leskovec, Cyril Zakka, Eduardo Pontes Reis, and Pranav Rajpurkar. 2023b. Med-flamingo: a multimodal medical few-shot learner. In *Proceedings of the 3rd Machine Learning for Health Symposium*, volume 225 of *Proceedings of Machine Learning Research*, pages 353–367. PMLR.

Sahal Shaji Mullappilly, Mohammed Irfan Kurpath, Omair Mohamed, Mohamed Zidan, Fahad Khan, Salman Khan, Rao Anwer, and Hisham Cholakkal. 2026. Medix-r1: Open ended medical reinforcement learning. *Preprint*, arXiv:2602.23363.

Junzhi Ning, Wei Li, Cheng Tang, Jiashi Lin, Chenglong Ma, Chaoyang Zhang, Jiyao Liu, Ying Chen, Shujian Gao, Lihao Liu, Yuandong Pu, Huihui Xu, Chenhui Gou, Ziyan Huang, Yi Xin, Qi Qin, Zhongying Deng, Diping Song, Bin Fu, and 8 others. 2025. Unimedvl: Unifying medical multimodal understanding and generation through observation-knowledge-analysis. *Preprint*, arXiv:2510.15710.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 181 others. 2023. GPT-4 Technical Report. *arXiv e-prints*, arXiv:2303.08774.

Jiazhen Pan, Che Liu, Junde Wu, Fenglin Liu, Jiayuan Zhu, Hongwei Bran Li, Chen Chen, Cheng Ouyang, and Daniel Rueckert. 2025. Medvlm-r1: Incentivizing medical reasoning capability ofnbsp;vision-language models (vlms) vianbsp;reinforcement learning. In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2025: 28th International Conference, Daejeon, South Korea, September 23–27, 2025, Proceedings, Part VII*, page 337–347, Berlin, Heidelberg. Springer-Verlag.

Pouya Pezeshkpour and Estevam Hruschka. 2024. Large language models sensitivity to the order of options in multiple-choice questions. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2006–2017, Mexico City, Mexico. Association for Computational Linguistics.

Yakun Zhu Xiaofan Zhang Qi Dou Qianhan Feng, Zhongzhen Huang. 2025. Ucagents: Unidirectional convergence for visual evidence anchored multi-agent medical decision-making. *arXiv preprint arXiv:2512.02485*.

Melanie Sclar, Yejin Choi, Yulia Tsvetkov, and Alane Suhr. 2024. Quantifying language models' sensitivity to spurious features in prompt design or: How i learned to start worrying about prompt formatting. In *The Twelfth International Conference on Learning Representations*.

Andrew Sellergren, Chufan Gao, Fereshteh Mahvar, Timo Kohlberger, Fayaz Jamil, Madeleine Traverse, Alberto Tono, Bashir Sadjad, Lin Yang, Charles Lau, Liron Yatziv, Tiffany Chen, Bram Sterling, Kenneth Philbrick, Richa Tiwari, Yun Liu, Madhuram Jajoo, Chandrashekar Sankarapu, Swapnil Vispute, and 23 others. 2026. Medgemma 1.5 technical report. *Preprint*, arXiv:2604.05081.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin,

Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *Preprint*, arXiv:2409.12191.

Chaoyi Wu, Xiaoman Zhang, Ya Zhang, Hui Hui, Yanfeng Wang, and Weidi Xie. 2025. Towards generalist foundation model for radiology by leveraging web-scale 2d&3d medical data. *Nature Communications*, 16(1):7866.

Yunfei Xie, Ce Zhou, Lang Gao, Juncheng Wu, Xianhang Li, Hong-Yu Zhou, Sheng Liu, Lei Xing, James Zou, Cihang Xie, and Yuyin Zhou. 2024. Medtrinity-25m: A large-scale multimodal dataset with multigranular annotations for medicine. *Preprint*, arXiv:2408.02900.

Lin Yang, Shawn Xu, Andrew Sellergren, Timo Kohlberger, Yuchen Zhou, Ira Ktena, Atilla Kiraly, Faruk Ahmed, Farhad Hormozdiari, Tiam Jaroensri, Eric Wang, Ellery Wulczyn, Fayaz Jamil, Theo Guidroz, Chuck Lau, Siyuan Qiao, Yun Liu, Akshay Goel, Kendall Park, and 28 others. 2024. Advancing multimodal medical capabilities of gemini. *Preprint*, arXiv:2405.03162.

Xingmeng Zhao, Ke Yang, and Anthony Rios. 2025. Comparing clinical and general LLMs on knowledge boundaries and robustness. In *Mechanistic Interpretability Workshop at NeurIPS 2025*.

Yuan Zhong, Ruinan Jin, Qi Dou, and Xiaoxiao Li. 2026. Can generalist vision language models (vlms) rival specialist medical vlms? benchmarking and strategic insights. *Preprint*, arXiv:2506.17337.

Yucheng Zhou, Lingran Song, and Jianbing Shen. 2025. MAM: Modular multi-agent framework for multimodal medical diagnosis via role-specialized collaboration. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 25319–25333, Vienna, Austria. Association for Computational Linguistics.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen Duan, Weijie Su, Jie Shao, Zhangwei Gao, Erfei Cui, Xuehui Wang, Yue Cao, Yangzhou Liu, Xingguang Wei, Hongjie Zhang, Haomin Wang, Weiye Xu, and 32 others. 2025. Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. *Preprint*, arXiv:2504.10479.

Dataset	Split	Size	Yes/No	Multiple Choice
VQA-RAD	Train	820	753	67
	Validation	205	187	18
	Test	272	251	21
	Total	*1297*	*1191*	*106*
PATH-VQA	Train	9804	9804	0
	Validation	3135	3135	0
	Test	3389	3389	0
	Total	*16328*	*16328*	*0*
SLAKE	Train	1943	1681	262
	Validation	422	358	64
	Test	415	355	60
	Total	*2780*	*2394*	*386*
OmniMedVQA	*Total*	*88,996*	*402*	*88,594*

Table 3: Summary of the number of close-ended examples in the train, validation, and test sets of all visual QA datasets.

A Dataset Details

In this section, we provide additional details on the visual QA datasets. In Table 3, we summarize the number of QA examples included in the train, validation, and test sets of each dataset, following the preprocessing steps described in Section 4. For VQA-RAD (Lau et al., 2018), PathVQA (He et al., 2020), and SLAKE (Liu et al., 2021), we report only the closed-ended visual QA examples, as our evaluations are restricted to the closed-ended setting. Following Jeong et al. (2024) and prior work, we convert all closed-ended questions into a consistent multiple-choice format by constructing answer options directly from the question content. No external distractor generation or cross-question negative sampling is performed; all candidate options are either explicitly stated in the question text or follow trivially from the binary answer type.

Concretely, the option construction procedure operates as follows. For **binary yes/no questions**, the two options *yes* and *no* are used as candidates. This covers all closed-ended questions in PATH-VQA (which is exclusively binary) and the majority of questions in VQA-RAD and SLAKE. For **non-binary closed questions** in VQA-RAD and SLAKE, the candidate options are the explicit answer alternatives already named in the question text. For example, a question such as *"Which is bigger in this image, left lung or left kidney?"* is assigned the options *left lung* and *left kidney*; a question such as *"Is this an MRI or a CT scan?"* is assigned *MRI* and *CT*. These options are extracted via a curated lookup table that enumerates all non-binary closed questions occurring in the official VQA-RAD and

SLAKE datasets, with the corresponding answer alternatives read verbatim from the question phrasing, following the same approach as Jeong et al. (2024). For SLAKE, a text-parsing fallback additionally covers residual comparison questions by extracting the two or three alternatives separated by "or" after the first comma in the question string. Records for which options cannot be unambiguously resolved, or for which the ground-truth answer does not appear in the extracted option set, are excluded from evaluation. We refer the reader to our project repository for the full preprocessing pipeline. For OmniMedVQA (Hu et al., 2024), which is released in two portions, open-access and restricted-access, we restrict our use to the open-access subset, as the restricted-access portion is not publicly available. Furthermore, since OmniMedVQA does not provide an official train/validation/test split, we report only the total number of examples for this dataset in Table 3.

To ensure tractable and comparable evaluations across datasets, we cap each dataset at a maximum of 500 test examples and 1,000 training examples for the probing experiments. Subsets are constructed using stratified sampling to preserve the distribution of question types and difficulty levels across splits, following prior work (Liu et al., 2025a; Mishra et al., 2026). Concretely, for the test set we use 500 examples from PATH-VQA, 415 from SLAKE (its full test set), and 263 from VQA-RAD, where the original 272 test examples are reduced after excluding questions whose associated images are unavailable. For the training set used in probing, we use 1,000 examples from PATH-VQA, 1,000 from SLAKE, and the full 820 examples from VQA-RAD.

For PATH-VQA specifically, since all closed-ended questions are binary (yes/no), stratified sampling is performed over the answer label to ensure that the 500-sample test subset maintains a balanced yes/no class distribution matching that of the full 3,389-sample test set. The resulting subset preserves an approximately equal proportion of affirmative and negative examples, ensuring that accuracy and macro-F1 scores computed on the subsample are representative of those on the full test set. For OmniMedVQA, the open-access subset spans eight imaging modalities: CT (Computed Tomography), Dermoscopy, Fundus Photography, MRI (Magnetic Resonance Imaging), Microscopy, OCT (Optical Coherence Tomography), X-Ray, and Ultrasound. To ensure balanced representation across all modalities, we construct our subsets via stratified sampling over modalities, selecting 100 examples per modality for training (800 examples in total) and 50 examples per modality for evaluation (400 examples in total).

Model Name	HF Link
HuatuoGPT-Vision-34B	🤗 Link
Aloe-Vision-72B-AR	🤗 Link
InfiMed-RL-3B	🤗 Link
UniMedVL-14B	🤗 Link
Gemma-3-4B-IT	🤗 Link
Gemma-3-27B-IT	🤗 Link
MedGemma-4B-IT	🤗 Link
MedGemma-27B-IT	🤗 Link
LLaVA-7B-Delta-v0	🤗 Link
LLaVA-Med-7B-Delta	🤗 Link
OpenFlamingo-9B	🤗 Link
Med-Flamingo	🤗 Link
MedVLThinker-3B	🤗 Link
MedVLThinker-7B	🤗 Link
MedVLThinker-32B	🤗 Link
MedMO-4B-Next	🤗 Link
MedMO-8B-Next	🤗 Link
MediX-R1-2B	🤗 Link
MediX-R1-30B	🤗 Link
Qwen2.5-VL-3B-Instruct	🤗 Link
Qwen2.5-VL-7B-Instruct	🤗 Link
Qwen2.5-VL-32B-Instruct	🤗 Link
Qwen2-VL-2B-Instruct	🤗 Link
Qwen3-VL-2B-Instruct	🤗 Link
Qwen3-VL-4B-Instruct	🤗 Link
Qwen3-VL-8B-Instruct	🤗 Link
Qwen3-VL-30B-A3B-Instruct	🤗 Link
BioMed-Qwen2-VL-2B	🤗 Link
BioMed-Llama-3.2-11B	🤗 Link
BioMed-InternVL3-1B	🤗 Link
Llama-3.2-11B-Vision-Instruct	🤗 Link

Table 4: Vision-language models and their Hugging Face repositories.

Dataset Links

- **VQA-RAD**: https://osf.io/89kps/overview
- **PathVQA**: https://huggingface.co/datasets/flaviagiammarino/path-vqa
- **SLAKE**: https://www.med-vqa.com/slake/
- **OmniMedVQA**: https://openxlab.org.cn/datasets/GMAI/OmniMedVQA

B Training process of Med-VLMs

Med-VLMs are developed by adapting general-purpose vision-language models through continued pretraining and instruction tuning on curated medical datasets, spanning radiology, pathology,

General Model	Medical Model	Medical Training Datasets
Gemma (4B, 27B) (Team et al., 2025)	MedGemma (4B, 27B) (Sellergren et al., 2026)	MIMIC-CXR, CT-US1, MRI-US1, PMC-OA, EyePACS, PAD-UFES-20, Digital Knee X-ray, internal histopathology (∼32.55M patches), internal dermatology (51K images), SLAKE, VQA-RAD.
LLaVA-V0-7B (Liu et al., 2023)	LLaVA-Med-7B (Li et al., 2023)	PMC-15M (600K caption pairs; 60K GPT-4 instruction pairs across CXR, CT, MRI, histopathology, and gross pathology).
OpenFlamingo-9B (Awadalla et al., 2023)	Med-Flamingo-9B (Moor et al., 2023b)	Medical Textbook dataset (MTB; ∼0.8M images), PMC-OA (1.3M pairs).
InternVL3-1B (Zhu et al., 2025)	BioMed-InternVL3-1B (Cheng et al., 2025)	PubMedVision/PMCRefined (500K caption pairs), 144K synthetic visual instruction tasks.
Llama-3.2-11B Vision (Grattafiori et al., 2024)	BioMed-Llama-3.2-11B (Cheng et al., 2025)	PubMedVision/PMCRefined (500K caption pairs), 144K synthetic visual instruction tasks.
Qwen2-VL-2B (Wang et al., 2024)	BioMed-Qwen2-VL-2B (Cheng et al., 2025)	PubMedVision/PMCRefined (500K caption pairs), 144K synthetic visual instruction tasks.
Qwen2.5-VL (3B, 7B, 32B) (Bai et al., 2025b)	MedVLThinker-RL (3B, 7B, 32B) (Huang et al., 2026)	PMC-VQA (115K filtered samples), m23K text-only medical QA (16.5K).
Qwen3-VL (4B, 8B) (Bai et al., 2025a)	MedMO (4B, 8B)-Next (Deria et al., 2026)	MedTrinity (18.5M), MIMIC-CXR, IU-Xray, CheXpert, PMC-VQA, PubMedVision, ROCO-V2, NIH Chest X-ray, DeepLesion, PMC-OA, SLAKE, VQA-RAD, PathVQA.
Qwen3-VL (2B, 30B) (Bai et al., 2025a)	MediX-R1 (2B, 30B) (Mullappilly et al., 2026)	PMC-VQA (25K), PathVQA (19.6K), SLAKE (4.9K), VQA-RAD (1.8K).

Table 5: General-to-medical VLMs correspondence and training datasets.

ophthalmology, and dermatology. As summarized in Table 5, training corpora vary widely in scale and modality, from large-scale image–caption pairs harvested from PubMed Central to GPT-4-generated instruction-following examples and reinforcement learning signals aimed at improving clinical reasoning.

C Additional Details on Prompting

In this section, we summarize the prompting details used for the general-domain and medical vision–language models evaluated in our work (Appendix C.1), and the default prompt formats used for each model family (Appendix C.2).

C.1 Reproducibility of Prompting Details

In Table 6, we provide a summary of the prompting details available for each model family. We share these details to document our best efforts at reproducing the prompting setups used in our evaluations; in cases where official documentation is absent, we describe the choices we made in designing these setups. In particular, we focus on three components: (i) a **system prompt**; (ii) a **prompt format** for closed-ended multiple-choice QA tasks; and (iii) the **sampling details** governing text generation (e.g., `temperature`, `top_p`).

Gemma 3 (4B and 27B). For both Gemma 3 variants (4B and 27B), we adopt greedy decoding to ensure fully deterministic and reproducible outputs, consistent with standard practice in medical VQA benchmarking and with the generation settings specified in the official Gemma 3 model card and model paper (Team et al., 2025). Specifically, greedy decoding means the model always selects the most probable next token at each step rather than sampling stochastically, which eliminates any randomness in the output. We also cap the number of generated tokens at 200, which is sufficient to produce any multiple-choice answer while avoiding unnecessary over-generation. Since generation is deterministic, parameters that control stochastic sampling such as nucleus sampling and top-k filtering are left at their defaults and have no effect. As discussed in Appendix C.2, the Gemma 3 instruction-tuned models are designed to operate with only two dialogue roles, namely the user and the model, and do not support a dedicated system role.[1] Accordingly, no system prompt is used, and the system prompt entry in Table 6 is marked as not applicable. The prompt format is marked as not available because it was derived from the MedGemma fine-tuning notebook rather than from a dedicated closed-ended evaluation protocol (see Appendix C.2).

MedGemma (4B and 27B). For both MedGemma variants (4B and 27B), we apply the same generation configuration as for Gemma 3, in line with the official MedGemma model card and model paper (Sellergren et al., 2026). Specifically, as with Gemma 3, we use greedy decoding, meaning the model always selects the most probable next token at each step rather than sampling stochastically. This

[1] https://ai.google.dev/gemma/docs/core/prompt-structure

eliminates any randomness in the output, ensuring that results are fully deterministic and reproducible across runs. We also cap the number of generated tokens at 200, which is sufficient to produce any multiple-choice answer while avoiding unnecessary over-generation. Since generation is deterministic, parameters that control stochastic sampling such as nucleus sampling and top-k filtering are left at their defaults and have no effect. Crucially, MedGemma is built on the Gemma 3 architecture rather than PaliGemma, and therefore shares the same two-role chat template with no native support for a system turn.[2] As with Gemma 3, we omit any system prompt, keeping the input minimal and consistent with the architecture's design. This deterministic setup is essential for fair comparison across model families in medical imaging evaluation.

LLaVA-V0 (7B). Following Jeong et al. (2024), we adopt the `simple_conv` conversational template[3] from the official LLaVA-Med repository (Liu et al., 2023) as the default prompt format. We note the following caveats. First, the exact system prompt and conversational format used in the original evaluation are not documented in the paper or the repository; we therefore default to the `simple_conv` template, which reflects the VICUNA-V0 backbone (Chiang et al., 2023) on which LLaVA-V0 is built. Second, the answer-choice formatting convention for closed-ended QA is inferred solely from the raw VQA-RAD results file[4] rather than from a dedicated evaluation protocol. Consistent with both our greedy decoding approach for the other model families and the generation settings of Jeong et al. (2024), we set the temperature to zero and restrict generation to a single beam, ensuring fully deterministic and reproducible outputs; accordingly, the sampling entry in Table 6 is marked as partially available.

LLaVA-Med (7B). Following Jeong et al. (2024), we adopt the `simple_conv_med` template[5] from the official repository (Li et al., 2023) as the default prompt format. The caveats regarding undocumented evaluation choices and inferred answer-choice formatting described for LLaVA-V0 above apply here as well; given this additional uncertainty, we also search over prompt formats for model-specific prompt selection. We select the `simple_conv_med` template specifically because it includes a system prompt tailored to LLaVA-Med (beginning "`You are LLaVA-Med, a large language and vision assistant trained by a group of researchers at Microsoft. . .`") and follows the same VICUNA-V0 conversational structure (Chiang et al., 2023). We apply the identical generation configuration as for LLaVA-V0, setting the temperature to zero and restricting generation to a single beam. Because LLaVA-Med was not pre-trained on multi-image inputs nor evaluated in a few-shot setting, details on few-shot example construction are not applicable.

OpenFlamingo (9B). For OpenFlamingo (Awadalla et al., 2023), no official documentation is available for the system prompt, prompt format, or sampling configuration in the context of closed-ended medical VQA evaluation; all three entries are therefore marked as not applicable in Table 6. We instead follow Jeong et al. (2024) exactly, adopting their prompt format and generation settings without modification. Consistent with their official configuration file,[6] we set the temperature to zero and restrict generation to a single beam, ensuring fully deterministic and reproducible outputs.

Med-Flamingo (9B). For Med-Flamingo (Moor et al., 2023b), we adopt by default the system prompt and prompt format provided in the demo code of the official GitHub repository;[7] accordingly, the system prompt entry in Table 6 is marked as partially available. However, since the demo does not specify how answer choices should be formatted in a closed-ended QA context, the prompt format entry is likewise marked as partially available, and we additionally search over prompt formats when performing model-specific prompt selection. As with OpenFlamingo above, we apply the identical generation configuration from Jeong

[2] https://developers.google.com/health-ai-developer-foundations/medgemma/model-card

[3] https://github.com/microsoft/LLaVA-Med/blob/b9a98a736d2ef05bcf5ff345be6403fb3a664eaf/llava/conversation.py#L243

[4] https://github.com/microsoft/LLaVA-Med/blob/b9a98a736d2ef05bcf5ff345be6403fb3a664eaf/llava/eval/eval_metrics/answer-file-llava-zeorshot.jsonl

[5] https://github.com/microsoft/LLaVA-Med/blob/b9a98a736d2ef05bcf5ff345be6403fb3a664eaf/llava/conversation.py#L257

[6] https://github.com/mlfoundations/open_flamingo/blob/main/open_flamingo/src/flamingo.py#L124

[7] https://github.com/snap-stanford/med-flamingo/blob/7bcbb6c3932c814a6f7ae4b838ae4fada39b42a4/scripts/demo.py#L54

Category	Model	System Prompt	Prompt Format	Sampling
General Models	Gemma (4B, 27B) (Team et al., 2025)	✗	✗	●
	LLaVA-V0-7B (Liu et al., 2023)	●	●	●
	OpenFlamingo-9B (Awadalla et al., 2023)	✗	✗	✗
	InternVL3-1B (Zhu et al., 2025)	✗	✓	●
	Llama-3.2-11B-Vision (Grattafiori et al., 2024)	✗	●	✗
	Qwen3-VL (2B, 4B, 8B, 30B) (Bai et al., 2025a)	✗	✓	✓
	Qwen2.5-VL (3B, 7B, 32B) (Bai et al., 2025b)	✗	✗	✗
	Qwen2-VL-2B (Wang et al., 2024)	✗	✗	✗
Medical Models	MedGemma (4B, 27B) (Sellergren et al., 2026)	✗	●	●
	LLaVA-Med-7B (Li et al., 2023)	●	●	●
	Med-Flamingo-9B (Moor et al., 2023b)	●	●	✗
	BioMed-Llama-3.2-11B (Cheng et al., 2025)	✗	✓	●
	BioMed-Qwen2-VL-2B (Cheng et al., 2025)	✗	✓	●
	BioMed-InternVL3-1B (Cheng et al., 2025)	✗	✓	●
	MedVLThinker-RL (3B, 7B, 32B) (Huang et al., 2026)	✓	✓	✓
	MedMO (4B, 8B)-Next (Deria et al., 2026)	✗	✓	✗
	MediX-R1 (2B, 30B) (Mullappilly et al., 2026)	✓	✓	✓

Table 6: Summary of prompting details available for each evaluated model. ✓ = fully provided; ● = partially provided; ✗ = not provided or not applicable.

et al. (2024) and their official configuration file,[8] setting the temperature to zero and restricting generation to a single beam; the sampling entry is therefore marked as not applicable.

BioMed-Llama-3.2-11B-Vision (11B). BioMed-Llama-3.2-11B-Vision is a medical vision–language model from Cheng et al. (2025), fine-tuned on the general-domain Llama-3.2-11B-Vision-Instruct backbone (Meta AI, 2024). No system prompt is documented in either the official paper or the corresponding HuggingFace model repositories; we therefore do not apply any system prompt, and that entry is marked as not applicable in Table 6. The prompt format is adopted directly from the multiple-choice VQA protocol documented in the official paper appendix,[9] and is therefore marked as fully available. The sampling configuration is partially available: we set the temperature to zero following the official inference script,[10] while other generation hyperparameters are left at their defaults; the sampling entry is accordingly marked as partially available.

BioMed-Qwen2-VL-2B (2B). BioMed-Qwen2-VL-2B is another model released by Cheng et al. (2025), obtained by fine-tuning the Qwen2-VL-2B-Instruct backbone (Wang et al., 2024) on the same domain-adapted medical data. As with BioMed-Llama-3.2-11B-Vision above, no system prompt is used, the prompt format is taken directly from the official paper appendix, and the temperature is set to zero following the same inference script; all three entries in Table 6 are therefore assigned the same marks as for BioMed-Llama-3.2-11B-Vision.

BioMed-InternVL3-1B (1B). BioMed-InternVL3-1B is another model from Cheng et al. (2025), fine-tuned on the InternVL3-1B backbone (Zhu et al., 2025). The prompting and generation setup is identical to that described for BioMed-Llama-3.2-11B-Vision and BioMed-Qwen2-VL-2B above: no system prompt is applied, the prompt format follows the multiple-choice VQA protocol from the official paper appendix, and the temperature is fixed at zero per the same inference script, resulting in the same set of marks in Table 6.

InternVL3-1B (1B). For InternVL3-1B (Zhu et al., 2025), no system prompt is documented in either the official paper or the HuggingFace model repository; we therefore do not apply any system prompt, and that entry is marked as not applicable in Table 6. For the prompt format, the official paper reports that evaluations are conducted using VLMEvalKit;[11] for consistency, we adopt the VLMEvalKit multiple-choice question prompt format directly,[12] and the prompt format entry is

[8] https://github.com/taekb/eval-medical-dapt/blob/main/configs/vlm/eval/model/med-flamingo-9b.yaml

[9] https://aclanthology.org/2025.findings-emnlp.17.pdf#page=18

[10] https://github.com/bigai-ai/QA-Synthesizer/blob/main/vllm_inference/inference.py#L33

[11] https://arxiv.org/pdf/2504.10479#page=7

[12] https://github.com/open-compass/VLMEvalKit/blob/43af13e052de6805a8b08cd04aed5e0d74f82ff5/vl

accordingly marked as fully available. The sampling configuration is partially available: we set the temperature to zero following the official InternVL3 evaluation script,[13] while other generation hyperparameters are left at their defaults.

Llama-3.2-11B-Vision (11B). For Llama-3.2-11B-Vision (Meta AI, 2024; Grattafiori et al., 2024), no system prompt is documented in either the official paper or the HuggingFace model repository; we therefore do not apply any system prompt, and that entry is marked as not applicable in Table 6. The prompt format is marked as partially available: the official vision prompt format documentation[14] specifies that the `<|image|>` tag must appear within the user message body immediately after the role header, with the text query following the image tag, as images only attend to subsequent text tokens; however, no dedicated closed-ended evaluation protocol is provided, and the answer-choice formatting is therefore inferred from this structural constraint rather than from an official benchmark setup. No sampling configuration is documented; we accordingly leave all generation hyperparameters at their defaults, and the sampling entry is marked as not applicable in Table 6.

Qwen3-VL (2B, 4B, 8B, and 30B). For the Qwen3-VL family (Bai et al., 2025a), no system prompt is documented in either the official paper or the HuggingFace model repositories; we therefore do not apply any system prompt, and that entry is marked as not applicable in Table 6. The prompt format is marked as fully available: the official Qwen3-VL paper documents the multiple-choice prompt format used for general-domain VQA benchmarks such as MMBench (Liu et al., 2024) and MMStar (Chen et al., 2024b),[15] and we adopt this format directly. The sampling configuration is likewise fully available: following the official Qwen3-VL paper[16] and the corresponding evaluation script,[17] we set the temperature to 0.7, which controls the randomness of the output distribution by scaling the logits before sampling, and the nucleus sampling threshold to 0.8, which restricts sampling to the smallest set of tokens whose cumulative probability mass meets or exceeds that value; all other generation hyperparameters are left at their defaults.

Qwen2.5-VL (3B, 7B, and 32B). For the Qwen2.5-VL family (Bai et al., 2025b), no system prompt is documented in either the official paper or the HuggingFace model repositories; as with Qwen3-VL above, we do not apply any system prompt. No official closed-ended evaluation protocol or dedicated multiple-choice prompt format is provided; accordingly, the prompt format entry is marked as not applicable in Table 6. Since Qwen2.5-VL shares the same underlying chat format as Qwen3-VL, we apply the same multiple-choice prompt format documented in the Qwen3-VL paper for consistency across the Qwen model family. Similarly, as no official sampling configuration is provided, we apply the same generation settings used for Qwen3-VL; the sampling entry is accordingly marked as not applicable. This approach is consistent with recent work that evaluates Qwen2-VL and related Qwen families using a unified evaluation framework (Gilal et al., 2025).

Qwen2-VL (2B). For Qwen2-VL (Wang et al., 2024), no system prompt is documented in either the official paper or the HuggingFace model repository, and no dedicated closed-ended evaluation protocol is provided; both entries are therefore marked as not applicable in Table 6. As with Qwen2.5-VL above, Qwen2-VL shares the same underlying chat format across the Qwen family, and we therefore apply the same multiple-choice prompt format and generation settings following the Qwen3-VL paper for consistency.

MedVLThinker-RL (3B, 7B, and 32B). For the MedVLThinker-RL family (Huang et al., 2026), both the system prompt and the multiple-choice prompt format are fully documented and available. The system prompt is taken directly from the official GitHub evaluation script,[18] and the prompt format follows the template specified in both the official paper[19] and the same evaluation script;[20] both entries are accordingly marked as fully available in Table 6. For sampling, although the repository

meval/dataset/image_mcq.py#L164

[13] https://github.com/OpenGVLab/InternVL/blob/main/internvl_chat/eval/vqa/evaluate_vqa.py#L498

[14] https://github.com/meta-llama/llama-models/blob/main/models/llama3_2/vision_prompt_format.md

[15] https://arxiv.org/pdf/2511.21631#page=35

[16] https://arxiv.org/pdf/2511.21631#page=20

[17] https://github.com/QwenLM/Qwen3-VL/blob/main/evaluation/mmmu/run_mmmu.py#L393

[18] https://github.com/UCSC-VLAA/MedVLThinker/blob/main/eval/run_offline_inference.py#L327

[19] https://arxiv.org/pdf/2508.02669#page=12

[20] https://github.com/UCSC-VLAA/MedVLThinker/blob/main/eval/run_offline_inference.py#L330

defines default generation hyperparameter values, the official demo instructs users to set the temperature to 0.6, which controls the randomness of the output distribution by scaling the logits before sampling, and the nucleus sampling threshold to 0.95, which restricts sampling to the smallest set of tokens whose cumulative probability mass meets or exceeds that value;[21] for consistency with the intended usage, we adopt these values accordingly, and the sampling entry is marked as fully available.

MedMO-4B-Next and MedMO-8B-Next. For both MedMO variants (4B-Next and 8B-Next) (Deria et al., 2026), we adopt the prompt template specified directly in the official GitHub evaluation script.[22] No system prompt is documented in either the official paper or the corresponding repository; we therefore do not apply any system prompt, and that entry is marked as not applicable in Table 6. The prompt format is taken directly from the official evaluation script and is accordingly marked as fully available. No official sampling configuration or generation hyperparameters are documented for closed-ended evaluation; we therefore leave all generation hyperparameters at their defaults, and the sampling entry is marked as not applicable in Table 6.

MediX-R1 (2B and 30B). For both MediX-R1 variants (2B and 30B) (Mullappilly et al., 2026), we adopt the prompt template, system prompt, and generation configuration specified directly in the official GitHub evaluation script.[23] The system prompt is fully documented in both the official paper appendix[24] and the corresponding training prompt template;[25] both entries are accordingly marked as fully available in Table 6. MediX-R1 is the second reasoning-oriented model family in our evaluation, alongside MedVLThinker-RL; it uses a dedicated system prompt that instructs the model to first identify the imaging modality, then reason through the problem within structured tags, before producing a final answer. Following the official evaluation script,[26] the system prompt and the question body are combined within a single user turn in chat-style formatting. For sampling, we adopt the generation hyperparameters specified in the official evaluation script:[27] specifically, the temperature is set to zero, meaning the model always selects the most probable next token at each step rather than sampling stochastically, which ensures fully deterministic and reproducible outputs; the nucleus sampling threshold is set to one, which places no restriction on the cumulative token probability mass and is therefore effectively inactive under greedy decoding. All other generation hyperparameters are left at their defaults, and the sampling entry is marked as fully available in Table 6.

C.2 Default Prompt Formats

In this section, we describe the default prompt formats used for each model family, using a representative multiple-choice question as a running example. We use the following color conventions throughout: the question text is shown in **violet**, the answer options in **orange**, and the image token in **red**. Importantly, the placement of the image token relative to the text block is not uniform across model families. For the Gemma 3 family (including MedGemma), the text block is placed *before* the image token within the user turn, following the ordering convention specified in the official model cards. We follow each model's recommended convention strictly, as incorrect ordering can affect how the model attends to the visual input.

Gemma 3 (4B and 27B). Both Gemma 3 variants are prompted using the standard two-role chat template prescribed by the model architecture, which supports only a `user` turn and a `model` turn. Since the Gemma 3 instruction-tuned models do not natively support a system role, any system-level text would be silently folded into the first user turn without a dedicated marker, we omit a system prompt entirely and keep the input as minimal as possible (Team et al., 2025). Each prompt consists of a single user turn containing the question text followed by the answer options, formatted as a multiple-choice question consistent with the structure demonstrated in the official MedGemma fine-

[21] https://github.com/UCSC-VLAA/MedVLThinker/tree/main?tab=readme-ov-file#demo
[22] https://github.com/genmilab/MedMO/blob/main/examples/scripts/medevalkit_loader.py#L35
[23] https://github.com/mbzuai-oryx/MediX-R1/blob/main/eval/utils.py#L83
[24] https://arxiv.org/pdf/2602.23363#page=17
[25] https://github.com/mbzuai-oryx/MediX-R1/blob/main/training/examples/format_prompt/medical_format.jinja
[26] https://github.com/mbzuai-oryx/MediX-R1/blob/main/eval/utils.py#L144
[27] https://github.com/mbzuai-oryx/MediX-R1/blob/main/eval/utils.py#L178

tuning notebook;[28] we adopt this same format for prompting. A single image is appended after the text within the same user turn, following the ordering convention specified in the model card; note that this model family does not support multi-image inputs (Team et al., 2025). The resulting prompt format is illustrated in Box 1.

Box 1: Gemma 3 (4B & 27B)

```
<start_of_turn>user
{question}
A: {option_A}
B: {option_B}
C: {option_C}
D: {option_D}
<image>
<end_of_turn>
<start_of_turn>model
```

MedGemma (4B and 27B). MedGemma shares the same underlying Gemma 3 architecture and tokenizer, and therefore uses an identical chat template structure (Sellergren et al., 2026). As with Gemma 3, the instruction-tuned models support only a user turn and a model turn, and do not natively support a system role; we therefore omit a system prompt entirely and keep the input minimal. Each prompt consists of a single user turn containing the question text followed by the answer options, formatted as a multiple-choice question consistent with the structure demonstrated in the official MedGemma fine-tuning notebook. A single image is appended after the text within the same user turn, following the ordering convention specified in the model card. Although MedGemma has been domain-adapted on medical imaging data (Sellergren et al., 2026), its prompt format is marked as partially available because no dedicated closed-ended evaluation protocol has been publicly released; we therefore rely on the format demonstrated in the fine-tuning notebook. The resulting prompt format is illustrated in Box 2.

[28] https://colab.research.google.com/github/google-health/medgemma/blob/main/notebooks/fine_tune_with_hugging_face.ipynb

Box 2: MedGemma (4B & 27B)

```
<start_of_turn>user
{question}
A: {option_A}
B: {option_B}
C: {option_C}
D: {option_D}
<image>
<end_of_turn>
<start_of_turn>model
```

LLaVA-V0 (7B). LLaVA-V0 uses the `simple_conv` template, which renders a flat dialogue string with no dedicated special token for the system role; the general-domain system preamble is prepended directly to the conversation. The conversation body follows the VICUNA-V0 format, with each turn prefixed by `### Human:` or `### Assistant:`, and generation is triggered by a trailing `### Assistant:` with no closing text. Answer options are appended inline within the human turn. The image is not represented by a placeholder `<image>` token at inference time; instead, the runtime loader injects the model's internal image-patch token block directly into the prompt, expanding it as `<im_start><im_patch>...<im_patch><im_end>` using the checkpoint's `mm_use_im_start_end` configuration[29] and an image token length of 256[30]. The resulting prompt format is illustrated in Box 3.

Box 3: LLaVA-V0 (7B)

```
A chat between a curious human and an artificial intelligence assistant. The assistant gives helpful, detailed, and polite answers to the human's questions.
### Human: {question} Please choose from the following options: [{option_1}, {option_2}, {option_3}, {option_4}]. <image>
### Assistant:
```

LLaVA-Med (7B). LLaVA-Med shares the same flat dialogue rendering and VICUNA-V0 conver-

[29] https://github.com/taekb/eval-medical-dapt/blob/main/src/dataset.py#L1163

[30] https://github.com/taekb/eval-medical-dapt/blob/main/src/vlm/llava_models/infer_llava.py#L370

sational structure as LLaVA-V0, differing only in its system prompt. The simple_conv_med template replaces the general-domain preamble with a medically-adapted one specific to LLaVA-Med. Image token injection, answer-option placement, and the trailing ### Assistant: generation trigger follow the same conventions described for LLaVA-V0 above. The resulting prompt format is illustrated in Box 4.

Box 4: LLaVA-Med (7B)

```
You are LLaVA-Med, a large language and vision assistant trained by a group of researchers at Microsoft, based on the general domain LLaVA architecture. You are able to understand the visual content that the user provides, and assist the user with a variety of medical and clinical tasks using natural language. Follow the instructions carefully and explain your answers in detail.
### Human: {question} Please choose from the following options: [{option_1}, {option_2}, {option_3}, {option_4}]. <image>
### Assistant:
```

OpenFlamingo (9B). OpenFlamingo uses no system prompt. Each prompt opens with a brief task instruction prepended directly to the conversation body, followed by the image token and the question text within the same turn. Answer options are formatted with lettered parenthetical labels on separate lines, and generation is triggered by a trailing Answer: stub. Notably, the image token is placed *before* the question text within the user turn, which is the opposite ordering convention from the Gemma 3 family described above. The resulting prompt format is illustrated in Box 5.

Box 5: OpenFlamingo (9B)

```
The following is a multiple-choice visual question requiring medical knowledge. Answer the question by choosing one of the provided answer options.
<image> {question}
(A) {option_A}
(B) {option_B}
(C) {option_C}
(D) {option_D}
Answer:
```

Med-Flamingo (9B). Med-Flamingo shares the same overall prompt structure as OpenFlamingo above, differing primarily in its system preamble. Rather than a generic task instruction, the preamble is replaced by the medically-adapted prompt from the official demo, which instructs the model to act as a helpful medical assistant and to follow provided examples before answering. The image token, question text, lettered answer options, and trailing Answer: stub follow the same ordering and formatting conventions described for OpenFlamingo. The resulting prompt format is illustrated in Box 6.

Box 6: Med-Flamingo (9B)

```
You are a helpful medical assistant. You are being provided with images, a question about the image and an answer. Follow the examples and answer the last question.
<image> {question}
(A) {option_A}
(B) {option_B}
(C) {option_C}
(D) {option_D}
Answer:
```

BioMed-Llama-3.2-11B-Vision (11B). BioMed-Llama-3.2-11B-Vision uses no system prompt. Each prompt begins with the image token on its own line, followed by the question text prefixed by Question:, and then the answer options formatted as a Python-style list on a single line prefixed by The choices are:. The resulting prompt format is illustrated in Box 7.

Box 7: BioMed-Llama-3.2-11B-Vision (11B)

```
<image>
Question: {question}
The choices are: [{option_A}, {option_B}, {option_C}, {option_D}]
```

BioMed-Qwen2-VL-2B (2B). BioMed-Qwen2-VL-2B shares an identical prompt structure to BioMed-Llama-3.2-11B-Vision above, with the image token placed before the question text, the same `Question:` and `The choices are:` prefixes, and options rendered as a Python-style list. The only difference lies in the underlying backbone architecture; the prompt format itself is unchanged across the three BioMed models released by Cheng et al. (2025). The resulting prompt format is illustrated in Box 8.

Box 8: BioMed-Qwen2-VL-2B (2B)

```
<image>
Question: {question}
The choices are: [{option_A}, {option_B}, {option_C}, {option_D}]
```

BioMed-InternVL3-1B (1B). BioMed-InternVL3-1B follows the same prompt format as BioMed-Llama-3.2-11B-Vision and BioMed-Qwen2-VL-2B, with no system prompt, the image token prepended before the question text, and answer options presented as a Python-style list under the `The choices are:` prefix. The resulting prompt format is illustrated in Box 9.

Box 9: BioMed-InternVL3-1B (1B)

```
<image>
Question: {question}
The choices are: [{option_A}, {option_B}, {option_C}, {option_D}]
```

InternVL3-1B (1B). InternVL3-1B uses no system prompt. Each prompt begins with the image token on its own line, followed by the question text prefixed by `Question:`, and then the answer options listed on separate lines under an `Options:` header, each prefixed by its corresponding letter label. A closing instruction line, `Please select the correct answer from the options above.`, is appended after the options. The image token is placed *before* the question text, consistent with the ordering convention used by the BioMed-InternVL3-1B described above. The resulting prompt format is illustrated in Box 10.

Box 10: InternVL3-1B (1B)

```
<image>
Question: {question}
Options:
A. {option_A}
B. {option_B}
C. {option_C}
D. {option_D}
Please select the correct answer from the options above.
```

Llama-3.2-11B-Vision (11B). Llama-3.2-11B-Vision uses no system prompt and follows the chat template prescribed by the official vision prompt format documentation. Each prompt is wrapped in the model's native special tokens, opening with `<|begin_of_text|>` and a `<|start_header_id|>user<|end_header_id|>` role header. The `<|image|>` tag is placed immediately after the role header and before the question text, as required by the architecture: images only attend to subsequent text tokens, so the question must follow rather than precede the image tag. As no official closed-ended evaluation protocol is provided, the formatting of answer options and the closing instruction `Answer with the letter only.` were designed by us; answer options are listed on separate lines with parenthetical letter labels, and the closing instruction is appended after the final option. Generation is triggered by a trailing `assistant<|end_header_id|>` header. The resulting prompt format is illustrated in Box 11.

Box 11: Llama-3.2-11B-Vision (11B)

```
<|begin_of_text|><|start_header_id|>user<|end_header_id|>
<|image|>{question}
Options:
A) {option_A}
B) {option_B}
C) {option_C}
D) {option_D}
Answer with the letter only.<|eot_id|>
<|start_header_id|>assistant<|end_header_id|>
```

Qwen3-VL (2B, 4B, 8B, and 30B). Qwen3-VL uses no system prompt. Each prompt begins with

the image token on its own line, followed by the question text prefixed by `Question:`, and the answer options listed on separate lines under an `Options:` header, each prefixed by its corresponding letter label followed by a period. A closing instruction line, `Please select the correct answer from the options above.`, is appended after the final option. This format follows the multiple-choice convention documented in the official Qwen3-VL paper for general-domain VQA benchmarks, and is identical in structure to the VLMEvalKit MCQ format adopted for InternVL3-1B described above. The image token is placed before the question text, following the official Qwen3-VL evaluation script.[31] The resulting prompt format is illustrated in Box 12.

Box 12: Qwen3-VL (2B, 4B, 8B, & 30B)

```
<image>
Question: {question}
Options:
A. {option_A}
B. {option_B}
C. {option_C}
D. {option_D}
Please select the correct answer from
the options above.
```

Qwen2.5-VL (3B, 7B, and 32B). Qwen2.5-VL uses no system prompt and shares the same prompt structure as Qwen3-VL above, owing to the common underlying chat format across the Qwen family. The image token, the `Question:` prefix, the lettered `Options:` block, and the closing instruction line follow the identical conventions described for Qwen3-VL. The resulting prompt format is illustrated in Box 13.

Box 13: Qwen2.5-VL (3B, 7B, & 32B)

```
<image>
Question: {question}
Options:
A. {option_A}
B. {option_B}
C. {option_C}
D. {option_D}
Please select the correct answer from
the options above.
```

[31] https://github.com/QwenLM/Qwen3-VL/blob/main/evaluation/mmmu/run_mmmu.py#L27

Qwen2-VL (2B). Qwen2-VL likewise uses no system prompt and follows the same prompt structure as Qwen3-VL and Qwen2.5-VL, for the same reasons of shared chat format across the Qwen family. All formatting conventions — image token placement, `Question:` prefix, lettered `Options:` block, and closing instruction — are identical to those described above. The resulting prompt format is illustrated in Box 14.

Box 14: Qwen2-VL (2B)

```
<image>
Question: {question}
Options:
A. {option_A}
B. {option_B}
C. {option_C}
D. {option_D}
Please select the correct answer from
the options above.
```

MedVLThinker-RL (3B, 7B, and 32B). MedVLThinker-RL is the first model family in our evaluation to use a reasoning-oriented prompt structure. A dedicated system prompt instructs the model to reason through the problem before answering, and the prompt body follows this convention by requiring the model to place its reasoning within `<think>` `</think>` tags and its final answer within `<answer>` `</answer>` tags. The question text is prefixed by `Question:` and the answer options are listed on separate lines under an `Options:` header, each prefixed by its corresponding letter label followed by a period. MedVLThinker places the `<image>` token at the beginning of the user turn, before the question text, to indicate where the visual input is injected. The resulting prompt format is illustrated in Box 15.

Box 15: MedVLThinker-RL (3B, 7B, & 32B)

[System]: You will solve a problem/request. You should provide your thoughts within <think> </think> tags before providing the answer. Write your final answer within <answer> </answer> tags.

[User]:
<image>
Question: {question}

Options:

A. {option_A}

B. {option_B}

C. {option_C}

D. {option_D}

MedMO-4B-Next and MedMO-8B-Next. Both MedMO variants use no system prompt. Each prompt begins with the image token on its own line, followed by the question text, and then the answer options listed on separate lines under an Options: header, each prefixed by its corresponding letter label enclosed in parentheses. No closing instruction line is appended after the final option. The image token is placed *before* the question text, consistent with the ordering convention adopted in the official evaluation script.[32] The resulting prompt format is illustrated in Box 16.

Box 16: MedMO-4B-Next & MedMO-8B-Next

<image>
{question}
Options:
(A) {option_A}
(B) {option_B}
(C) {option_C}
(D) {option_D}

MediX-R1 (2B and 30B). MediX-R1 uses a reasoning-oriented prompt structure and is the only model family in our evaluation to require explicit image modality identification as part of its output format. The system prompt instructs the model to act as a medical AI assistant with advanced reasoning capabilities, and prescribes a strict three-step output structure: first, the model must output a single imaging modality tag from a predefined set (e.g., <X_RAY>, <MRI_SCAN>, <CT_SCAN>); second, it must place its reasoning within <thinking></thinking> tags; and third, it must provide its final answer within <answer></answer> tags, with no additional text permitted outside these tags. Following the official evaluation script,[33] the system prompt is prepended directly within the user turn rather than occupying a dedicated system role, so that the full input is rendered as a single combined user message. Within the question body, the <image> token is placed after the Question: label and before the question text, which is the opposite ordering convention from most other model families in our evaluation that place the image token before the question prefix. Answer options are listed on separate lines under a Choices: header, each prefixed by its corresponding letter label followed by a period. The resulting prompt format is illustrated in Box 17.

[32] https://github.com/genmilab/MedMO/blob/main/examples/scripts/medevalkit_sft_loader.py#L198

[33] https://github.com/mbzuai-oryx/MediX-R1/blob/main/eval/utils.py#L144

Box 17: MediX-R1 (2B & 30B)

```
You are a Medical AI Assistant with advanced reasoning capabilities

Your task:
1. First output the image modality tag from this set:
<X_RAY>, <MICROSCOPY>, <CLINICAL_PHOTOGRAPHY>, <CT_SCAN>, <GRAPHICS>,
<ANGIOGRAPHY>, <PET_SCAN>, <ULTRASOUND>, <MRI_SCAN>, <FUNDUS_PHOTOGRAPHY>,
<OCT_SCAN>, <ENDOSCOPY>, <MAMMOGRAPHY>, <FLUOROSCOPY>, <OTHER>, <SPECT>
(Only output the tag, nothing else.)
2. Then output the thinking and medical reasoning process in <thinking>...</thinking> tags.
3. Finally, provide the correct answer inside <answer>...</answer> tags.
4. Do not include any extra information or text outside of these tags.

Question:
<image>
{question}

Choices:
A. {option_A}
B. {option_B}
C. {option_C}
D. {option_D}
```

D Few-Shot Probing Results

All probing results reported in the main paper correspond to the zero-shot setting, in which the prompt contains no in-context examples from the target benchmark. Table 7 compares zero-shot and 3-shot probing accuracy across a representative set of models and benchmarks. In the 3-shot setting, the prompt is augmented with three labeled examples from the training split of the *same* benchmark, following standard in-context learning practice.

The zero-shot setting is competitive with or superior to 3-shot for most models. Zero-shot probing outperforms 3-shot on average for 7 of 9 models. The primary exceptions are Qwen3-VL-30B and Qwen2.5-VL-32B, where 3-shot prompting provides additional signal that improves VQA-RAD and PATH-VQA performance at the largest scale.

Table 7: Probing accuracy (%) under 0-shot and 3-shot settings. Bold indicates the better result per model–benchmark pair. The shaded column highlights the average.

Model	Shot	PATH	SLAKE	RAD	Avg.
InternVL3-1B	3	58.20	66.99	**68.82**	64.67
	0	**62.80**	**75.66**	65.02	**67.83**
BioMed-Qwen2-VL-2B	3	37.20	41.93	12.17	30.43
	0	**56.20**	**63.62**	**65.02**	**61.61**
Qwen3-VL-2B	3	**57.20**	74.22	**65.02**	65.48
	0	56.80	**77.83**	64.64	**66.42**
Qwen2.5-VL-3B	3	**60.00**	62.89	68.06	63.65
	0	57.60	**74.22**	**75.29**	**69.03**
LLaVA-V0-7B	3	54.20	42.41	44.87	47.16
	0	**56.00**	**50.36**	**49.43**	**51.93**
Qwen3-VL-8B	3	**59.60**	74.70	**73.38**	**69.23**
	0	58.60	**76.63**	69.58	68.27
Open-Flamingo-9B	3	49.80	43.86	47.53	47.06
	0	**53.00**	**48.92**	**48.67**	**50.20**
Qwen3-VL-30B	3	**67.20**	79.76	76.81	**74.59**
	0	61.20	**80.96**	**77.95**	73.37
Qwen2.5-VL-32B	3	**66.60**	**80.72**	**87.45**	**78.26**
	0	57.20	77.59	82.51	72.43

For models such as BioMed-Qwen2-VL-2B, 3-shot probing substantially *degrades* performance (30.43% vs. 61.61% average), suggesting that in-context examples from different domains can disrupt the hidden-state structure on which the probe relies. These results validate our choice of zero-shot as the primary evaluation setting, as it is both simpler and generally more effective for linear probing.

E Option-Likelihood Prompting and Statistical Significance

A core methodological concern is whether the prompting baseline underperforms because of free-text generation artifacts, refusals, verbose paraphrases, or formatting failures, rather than genuine knowledge limits. We address this by evaluating **option-likelihood scoring**: each candidate answer is scored by its conditional log-probability under the frozen VLM, and the highest-scoring option is selected. This protocol sidesteps generation artifacts entirely and provides an upper bound on what prompting can extract without decoding. Table 8 compares free-text prompting, option-likelihood scoring, and MedProb for three representative models.

Option-likelihood improves over free-text prompting by an average of 4.6 points across nine model–dataset pairs (59.8% → 64.4%), confirming

that generation does suppress some correct signal. However, probing outperforms option-likelihood by a further 14.2 points (64.4% → 78.6%), a gap far too large to attribute to generation artifacts. The probing advantage therefore reflects richer access to the model's internal representations, not merely an elicitation-format effect.

Model	Method	PVQ	SLK	RAD	Avg.
LLaVA-V0-7B	Free-text	49.6	45.5	42.6	45.9
	Opt.-likeli.	45.8	53.7	56.2	51.9
	Probing	84.6	68.0	66.9	**73.2**
Llama-3.2-11B	Free-text	59.8	69.9	66.5	65.4
	Opt.-likeli.	62.2	72.8	72.1	69.0
	Probing	86.2	82.4	75.4	**81.3**
Qwen3-VL-8B	Free-text	59.0	76.6	68.8	68.1
	Opt.-likeli.	67.2	77.3	72.4	72.3
	Probing	86.2	84.1	73.9	**81.4**

Table 8: Accuracy (%) for free-text prompting, option-likelihood scoring, and MedProb on three representative VLMs. PVQ = PATH-VQA; SLK = SLAKE; RAD = VQA-RAD. Llama = Llama-3.2-11B-Vision.

To confirm that the probing advantage is statistically reliable, we ran paired significance tests on the same model–dataset pairs, comparing probing against free-text prompting on exactly aligned predictions. Table 9 reports the McNemar exact p-value and a 95% bootstrap confidence interval for the probing-minus-prompting accuracy delta. Eight of nine comparisons are significant at $p < 0.05$. The sole exception is Qwen3-VL-8B on VQA-RAD ($\Delta = +5.1$, CI $[-0.4, 11.0]$, $p = 0.098$), where the smaller test set (263 examples) limits statistical power. On PATH-VQA, all three models clear $p < 10^{-21}$, consistent with the observation that the probing advantage is most pronounced on the pathology benchmark.

Model	Dataset	Δ	95% CI	p
LLaVA-V0-7B	PATH-VQA	+35.0	[29.6, 40.4]	6.5×10^{-32}
	SLAKE	+22.4	[15.7, 29.2]	4.1×10^{-10}
	VQA-RAD	+24.3	[16.2, 32.7]	5.6×10^{-8}
Llama-3.2-11B	PATH-VQA	+26.4	[21.2, 31.6]	2.4×10^{-21}
	SLAKE	+12.5	[7.7, 17.3]	1.2×10^{-6}
	VQA-RAD	+8.8	[3.3, 14.3]	2.7×10^{-3}
Qwen3-VL-8B	PATH-VQA	+27.2	[22.6, 31.8]	1.0×10^{-27}
	SLAKE	+7.5	[3.1, 11.8]	1.3×10^{-3}
	VQA-RAD	+5.1	[−0.4, 11.0]	0.098

Table 9: Paired significance analysis for probing vs. free-text prompting. Δ = probing−prompting accuracy (percentage points). Model abbreviations as in Table 8.

Model	Train Dataset	PATH-VQA	SLAKE	VQA-RAD
InternVL3-1B	OmniMedVQA	55.40	72.29	61.60
	PATH-VQA	**82.20**	57.35	57.41
	SLAKE	63.00	**83.13**	58.94
	VQA-RAD	61.40	75.18	**72.62**
LLaVA-V0-7B	OmniMedVQA	53.60	51.08	53.61
	PATH-VQA	**84.60**	53.49	55.13
	SLAKE	58.60	**67.95**	50.57
	VQA-RAD	62.00	54.70	**66.92**
Qwen3-VL-2B	OmniMedVQA	54.20	67.71	55.89
	PATH-VQA	**83.80**	56.63	52.85
	SLAKE	58.60	**84.10**	60.84
	VQA-RAD	58.60	71.33	**73.00**

Table 10: Probing accuracy (%) across models and training datasets. Bold indicates in-domain (train and test from same benchmark). PATH-VQA-trained probes show severe domain-locking, collapsing on radiology benchmarks. VQA-RAD-trained probes transfer well to SLAKE due to shared radiology domain.

F Cross-Dataset Generalization

Table 10 reports probing accuracy for three models across all combinations of training and test datasets, including a generalist condition using OmniMedVQA training data.

In-domain training consistently yields the strongest probing accuracy, confirming that domain-matched representations are most linearly separable. However, a striking asymmetry emerges: models trained on PATH-VQA suffer dramatic performance collapses on SLAKE and VQA-RAD, with InternVL3-1B dropping to 57.35% on SLAKE and 57.41% on VQA-RAD, figures barely above random and well below even the generalist OmniMedVQA baseline of 72.29% and 61.60%. This *domain-locked* behavior is consistent across all three models and indicates that pathology-specific visual patterns in PATH-VQA induce highly specialized representations that do not transfer to radiology benchmarks.

In contrast, models trained on VQA-RAD exhibit substantial cross-domain transfer to SLAKE: 75.18% for InternVL3-1B and 71.33% for Qwen3-VL-2B. This asymmetry likely reflects the shared radiology domain between VQA-RAD and SLAKE, where overlapping visual concepts support generalization. The generalist OmniMedVQA condition produces consistently moderate results but never approaches in-domain performance, indicating that broad medical coverage alone is insufficient to match domain-specific representation learning.

G Additional Layer-wise Probing Results

This appendix collects the layer-wise probing accuracy plots for the remaining general-medical model pairs. Across families, clinical information is typically decodable from very early layers, but the effect of medical specialization is not uniform: some medical models improve the earliest and middle representations, whereas several general-purpose backbones recover or surpass them in the deeper layers. The pairwise discussions below summarize the dominant trend in each plot and point to the corresponding figure.

G.1 Gemma-4B vs. MedGemma-4B

Figure 9 shows that MedGemma-4B is already slightly ahead in the early layers and then opens a clearer margin through the middle and late parts of the network. The separation becomes most visible in the final third of the model, suggesting that medical specialization improves how clinically relevant information is preserved and amplified in deeper representations.

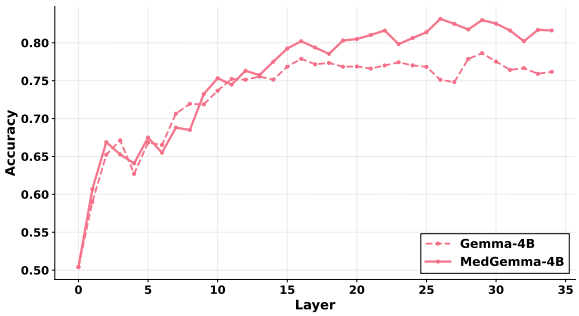

Figure 9: Layer-wise probing accuracy for Gemma-4B and MedGemma-4B. The medical model is competitive from the first layers and establishes a clearer advantage in the deeper part of the network.

G.2 Gemma-27B vs. MedGemma-27B

As shown in Figure 10, the larger Gemma pair is much more balanced. Gemma-27B leads more often in the early layers, but the two curves converge in the middle of the model and MedGemma-27B becomes slightly stronger in part of the late-layer region, indicating that scaling reduces the gap between the general and medically adapted variants.

G.3 Open-Flamingo-9B vs. Med-Flamingo-9B

Figure 11 presents one of the cleanest specialization gains in the appendix. Med-Flamingo-9B stays above Open-Flamingo-9B for almost the entire depth of the model, and the gap remains fairly stable from shallow to deep layers, implying that the medical adaptation improves clinical separability throughout the representation stack rather than only at the end.

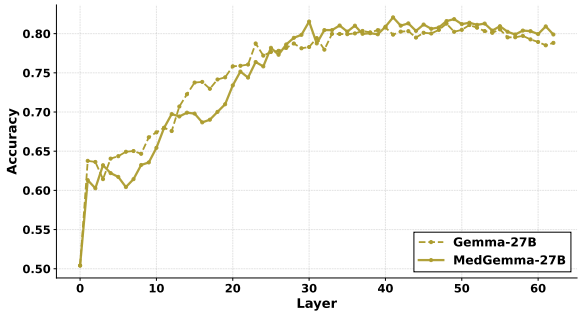

Figure 10: Layer-wise probing accuracy for Gemma-27B and MedGemma-27B. The two models remain close overall, with an early advantage for the general model and a more competitive late-layer profile for the medical variant.

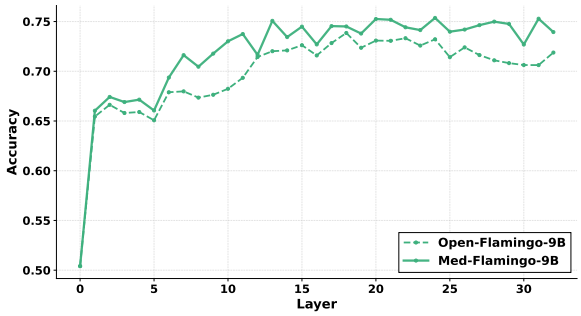

Figure 11: Layer-wise probing accuracy for Open-Flamingo-9B and Med-Flamingo-9B. The medical model maintains a consistent advantage across nearly all layers.

G.4 InternVL3-1B vs. BioMed-InternVL3-1B

The pattern in Figure 12 is stage-dependent. BioMed-InternVL3-1B has a small advantage across much of the early and middle layers, but the general InternVL3-1B catches up in the final third and finishes with the stronger deep-layer accuracy. This suggests that medical tuning helps earlier clinical extraction, while the base model retains a stronger late-layer refinement stage.

G.5 LLaVA-7B vs. LLaVA-Med-7B

Figure 13 shows a near-overlap between LLaVA-7B and LLaVA-Med-7B. The curves cross several times, the mid-layer region slightly favors the general model, and both models peak at almost the same depth and accuracy. For this pair, medical

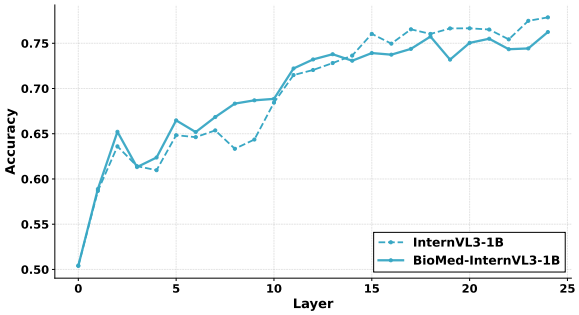

Figure 12: Layer-wise probing accuracy for InternVL3-1B and BioMed-InternVL3-1B. The medical model is stronger early, while the general model overtakes it in the deeper layers.

adaptation appears to change the distribution of information only marginally.

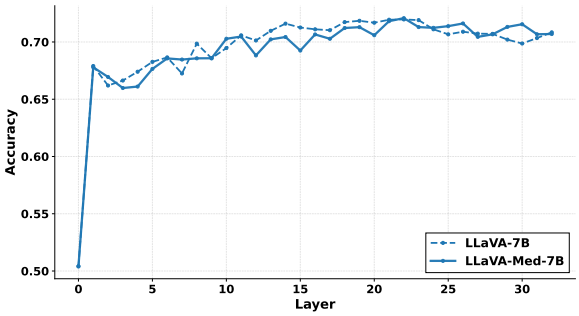

Figure 13: Layer-wise probing accuracy for LLaVA-7B and LLaVA-Med-7B. The two curves are tightly matched, with only small and localized differences across depth.

G.6 LLaMA3.2-11B-Vision vs. BioMed-LLaMA3.2-11B

In Figure 14, BioMed-LLaMA3.2-11B is generally ahead after the first few layers and keeps that margin through most of the network. The difference is not dramatic, but it is persistent, which is consistent with a medical adaptation that improves clinical linear separability without radically changing the overall shape of the representation trajectory.

G.7 Qwen2-VL-2B vs. BioMed-Qwen2-VL-2B

Figure 15 indicates that Qwen2-VL-2B is slightly stronger through the early and middle layers, while the medical variant stays close and reduces the gap toward the end. The general model nevertheless achieves the higher late-layer peak, suggesting that domain adaptation does not consistently dominate once the backbone reaches its deepest representations.

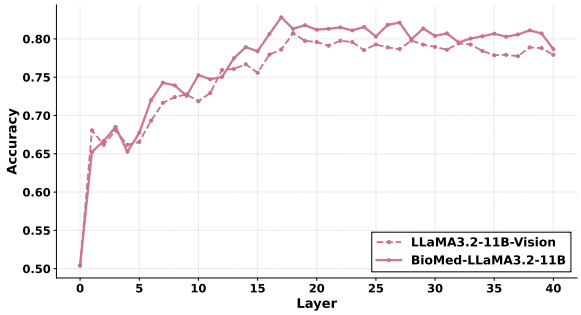

Figure 14: Layer-wise probing accuracy for LLaMA3.2-11B-Vision and BioMed-LLaMA3.2-11B. The medical model holds a modest but persistent advantage across most of the depth.

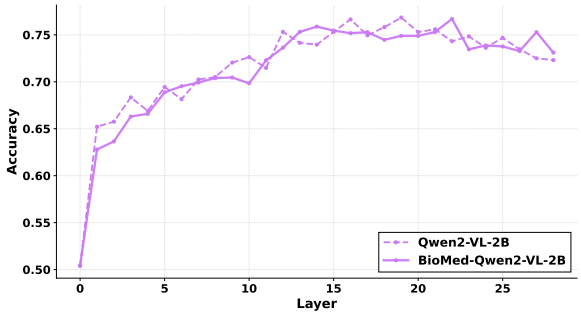

Figure 15: Layer-wise probing accuracy for Qwen2-VL-2B and BioMed-Qwen2-VL-2B. The general model has a small early-to-middle advantage and reaches a slightly higher late-layer maximum.

G.8 Qwen2.5-VL-3B vs. MedVLThinker-RL-3B

As shown in Figure 16, Qwen2.5-VL-3B and MedVLThinker-RL-3B remain close for much of the network, but the general model separates more clearly in the final third and attains the stronger peak accuracy. This pattern suggests that the base Qwen2.5-VL-3B keeps a deeper-layer advantage even when the medical model is competitive in the earlier stages.

G.9 Qwen2.5-VL-32B vs. MedVLThinker-RL-32B

Figure 17 shows a broad advantage for Qwen2.5-VL-32B. After a few minor early exchanges, the general model is ahead for most middle and late layers and reaches the higher overall maximum. The medical variant remains competitive, but at this scale the general backbone appears to retain stronger clinically decodable information in deep representations.

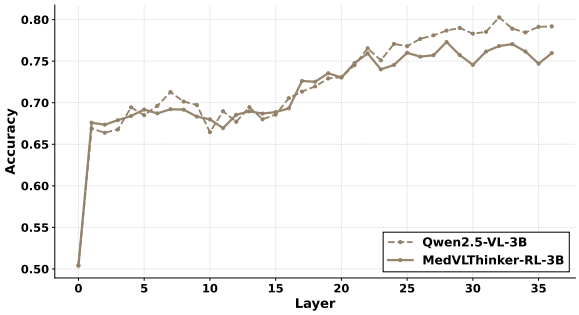

Figure 16: Layer-wise probing accuracy for Qwen2.5-VL-3B and MedVLThinker-RL-3B. The curves stay close early on, but the general model pulls ahead in the deepest layers.

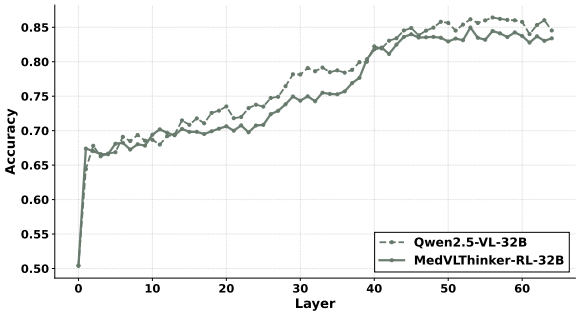

Figure 17: Layer-wise probing accuracy for Qwen2.5-VL-32B and MedVLThinker-RL-32B. The general model leads through most middle and late layers and achieves the higher peak accuracy.

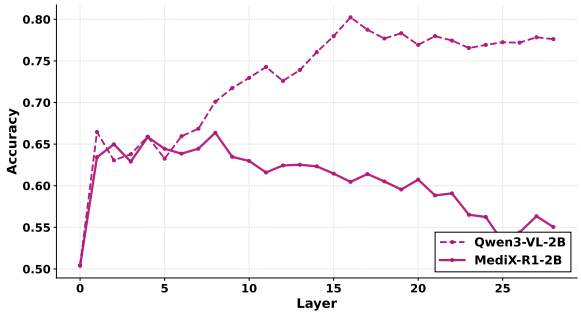

Figure 18: Layer-wise probing accuracy for Qwen3-VL-2B and MediX-R1-2B. The general model quickly establishes a large and growing advantage across the deeper layers.

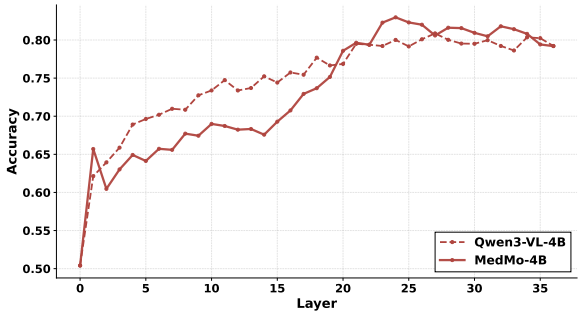

Figure 19: Layer-wise probing accuracy for Qwen3-VL-4B and MedMo-4B. The general model leads across much of the network, but the medical model rebounds late and reaches a slightly higher maximum.

G.10 Qwen3-VL-2B vs. MediX-R1-2B

Figure 18 exhibits the largest gap among all appendix pairs. After a brief and noisy start, Qwen3-VL-2B moves decisively ahead and the margin widens sharply through the middle and final layers, while MediX-R1-2B peaks much earlier and at a substantially lower level. This suggests that the medical variant loses a considerable amount of deep-layer clinical separability relative to the base model.

G.11 Qwen3-VL-4B vs. MedMo-4B

The curves in Figure 19 show a split trend. Qwen3-VL-4B leads for much of the first two thirds of the model, but MedMo-4B recovers in the later layers and eventually reaches the higher absolute peak. The result points to a medical model that is weaker in the intermediate regime yet competitive again at the deepest layers.

G.12 Qwen3-VL-8B vs. MedMo-8B

Figure 20 follows a related but more balanced pattern. Qwen3-VL-8B is stronger across most early and middle layers, whereas MedMo-8B narrows the gap later and nearly matches the general model at the best-performing depth. The medical adaptation therefore improves late-layer competitiveness, but it does not overturn the general model's broader lead across the stack.

G.13 Qwen3-VL-30B vs. MediX-R1-30B

Finally, Figure 21 shows a strong and stable advantage for Qwen3-VL-30B. Aside from a few isolated layer crossings, the general model remains ahead almost everywhere, and the gap grows again in the final layers where the best deep representations emerge. This indicates that, for the 30B pair, the base Qwen3-VL model preserves the more linearly accessible clinical information across the full depth of the network.

H Extended Error Analysis

This appendix provides representative case studies that support the qualitative error analysis presented in Section 5. For each of the three behavioral categories, we first present a summary table of the

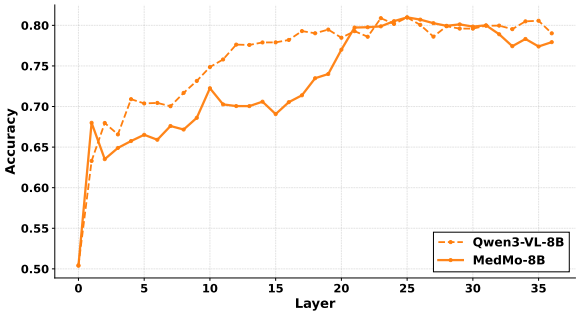

Figure 20: Layer-wise probing accuracy for Qwen3-VL-8B and MedMo-8B. The general model leads through most early and middle layers, while the medical model catches up in the deeper region.

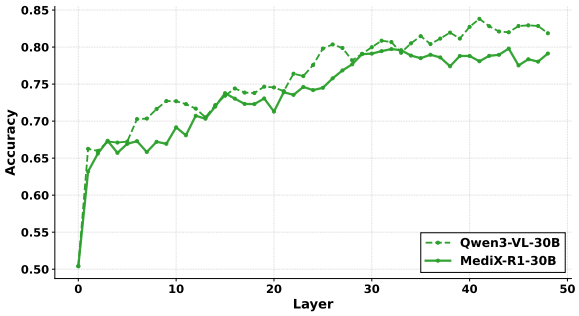

Figure 21: Layer-wise probing accuracy for Qwen3-VL-30B and MediX-R1-30B. The general model stays ahead across almost the entire network and widens the gap again in the deepest layers.

selected examples, then give a per-example breakdown that includes a verbatim excerpt from the model's generated response, the prediction made by the linear probe, the ground-truth label, and an interpretation of why each method succeeded or failed. Examples are drawn from the 300-response analysis described in Section 5 (10 samples × 3 datasets × 10 models). We denote the ground truth label as **GT**, the prompt-based prediction as **Prompt**, and the linear probe prediction as **Probe**. Correct predictions are marked ✓ and incorrect ones ×.

H.1 Where MedProb Improves Over Prompting

These are cases in which the probe recovers the correct answer despite the prompted model reasoning itself into a wrong conclusion. The common thread is that prompting introduces hallucinated or misapplied medical language that overrides a correct visual signal already present in the model's latent representation.

Example E-1

Ex.	Modality / Task	GT	Prompt	Probe	Failure Pattern
E-1	Histopathology (soft-tissue mass classification)	B (No)	A ×	B ✓	Hallucinated pathology narrative
E-2	Brain MRI (imaging modality identification)	B (No)	A ×	B ✓	Correct rule, misapplied context

Table 11: Examples where MedProb succeeds and prompting fails (**Probe** = ✓, **Prompt** = ×).

Prompt model output — idx 440

"Based on the image, the answer is **A: yes**. The image shows a lobulated mass with a bluish cartilaginous hue infiltrating the soft tissues. This is a characteristic appearance of a cartilaginous tumor."

Prompt prediction: ***A (Yes)*** × *Probe prediction:* ***B (No)*** ✓ *GT:* ***B (No)***

Figure 22: Semantic hallucination failure (idx 440): the model accurately describes the visual features but constructs a false pathology narrative and answers incorrectly; the probe at layer 16 is unaffected by this chain-of-thought error.

As shown in Figure 22, the prompted model correctly identifies two real visual features, the lobulated shape and the bluish cartilaginous hue, but then constructs a pathology narrative from them ("characteristic appearance of a cartilaginous tumor") and uses that narrative to justify answering *yes*, the wrong answer. This is a prototypical **semantic hallucination** failure: the model describes what it sees accurately but draws a false clinical conclusion. The linear probe, reading directly from layer-16 activations without generating any text, is not susceptible to this chain-of-thought error and correctly outputs the *no* label. This pattern, where prompting over-interprets a real visual cue into a wrong binary answer, is the single most common failure mode corrected by the probe.

Example E-2

Figure 23 demonstrates a subtler failure: the model's radiological reasoning is internally correct. It accurately defines T2 MRI contrast (CSF bright, parenchyma dark) and even observes those features in the image. Yet the final answer is wrong. The model applied a textbook rule correctly but misattributed it to the wrong conclusion given the actual question being asked. This is a **rule-application failure**, not a knowledge failure. The probe at layer 22 is unaffected by the verbalized reasoning chain and holds the correct answer in its activation pat-

Prompt model output — idx 383

"Let me think. The image provided is an MRI scan, specifically a T2-weighted image. This determination can be made based on the contrast and brightness levels of the different tissues within the brain. In a T2-weighted MRI, fluids such as cerebrospinal fluid (CSF) appear bright, and the brain tissue appears darker. This contrast is clearly visible in the image, with the ventricles and other fluid-filled spaces appearing bright white, and the brain parenchyma appearing in varying shades of gray. Therefore, the correct answer to the question is: **yes**."

Prompt prediction: ***A (Yes)*** × *Probe prediction:* ***B (No)*** ✓ *GT:* ***B (No)***

Figure 23: Rule-application failure (idx 383): the model's radiological reasoning is internally correct but misattributed to the wrong conclusion; the probe at layer 22 holds the correct answer in its activation pattern regardless of the verbalized reasoning chain.

tern. The key insight is that a latent representation can encode the correct binary signal even when the language decoder reasons convincingly but incorrectly from the same visual input.

H.2 Where MedProb Fails to Improve

MedProb fails to resolve errors on questions where the underlying visual distinction is too fine-grained, structurally ambiguous, or beyond the discriminative capacity of the model's representations at any single layer. In these cases, both prompting and probing fail simultaneously, indicating that the deficit lies in the depth of perceptual encoding rather than in the elicitation mechanism. These tasks tend to involve subtle radiological patterns (e.g., distinguishing ring-enhancing from non-ring-enhancing lesions), complex histological tissue architecture, or multi-feature pathological descriptions requiring simultaneous integration of several visual cues. When representations do not encode a reliable decision boundary, a linear probe has nothing to extract, and these questions expose the true perceptual ceiling of the underlying vision-language model. No elicitation mechanism can recover signal that is absent from the representation.

Example E-3

Figure 24 shows responses from **two different evaluated models** on the same question (idx 223). Each model generates its own prompt response and has its own probe applied at a different hidden layer. Their independent failure on the same question strengthens the conclusion: this is a task-level difficulty, not an artifact of any single model.

Ex.	Modality / Task	GT	Prompt	Probe	Failure Pattern
E-3	Brain MRI (ring-enhancement pattern detection)	A (Yes)	B ×	B ×	Subtle spatial pattern beyond model's resolution
E-4	Gross pathology (bowel wall skip-lesion detection)	B (No)	A ×	A ×	Fine-grained structural discrimination failure

Table 12: Examples where both MedProb and prompting fail (**Probe** = ×, **Prompt** = ×).

Model 1 — prompt response (idx 223)

"Ring-enhancing lesions are typically seen in conditions like abscesses or some tumors, where there is a ring of enhancement around a central area. Looking at the image, the lesions here, do they show ring enhancement? The lesions don't show a ring pattern. So the answer would be B. no."

Prompt: ***B (No)*** × *Probe:* ***B (No)*** × *GT:* ***A (Yes)***

Model 2 — prompt response (idx 223)

"The lesions have a more uniform appearance with a hypointense core and a surrounding area of hyperintensity, which is more consistent with the appearance of multiple sclerosis plaques on T2-weighted MRI images. These plaques are often referred to as 'Dawson's fingers' and are a common finding in patients with multiple sclerosis. Therefore, based on the MRI sequence presented, the lesions are not ring-enhancing."

Prompt: ***B (No)*** × *Probe:* ***B (No)*** × *GT:* ***A (Yes)***

Figure 24: Perceptual ceiling failure (idx 223): two independently trained models both fail on ring-enhancement detection across different reasoning strategies, with both prompt and probe predicting incorrectly, indicating a task-level perceptual limitation rather than a model-specific artifact.

Both models demonstrate genuine radiological expertise: Model 1 correctly defines ring enhancement, and Model 2 appropriately considers multiple sclerosis as a differential diagnosis, even invoking the clinical term "Dawson's fingers." Yet both arrive at the wrong answer, and both probes also fail. The fact that two independently trained models, using different reasoning strategies, produce the same wrong conclusion—with both their latent representations and their language outputs failing—indicates this is a **perceptual encoding failure** rather than a reasoning failure. The fine-

grained spatial distinction between ring-shaped and nodular enhancement is not encoded reliably in any probed layer of either model.

Example E-4

Prompt model output — idx 388

"The image shows a segment of the bowel with a thickened wall and a narrowed lumen, which is highlighted in pink. The surrounding areas of the bowel appear to be normal or 'skipped,' meaning they are not involved in the thickening or narrowing. This pattern is characteristic of certain conditions where segments of the bowel are affected, while intervening areas remain normal. Given the question asks if the corresponding area shows a segment of thickened wall with a narrow lumen [...] and the intervening areas are uninvolved or skipped, the image supports this description."

Prompt prediction: ***A (Yes)*** × *Probe prediction:* ***A (Yes)*** × *GT:* ***B (No)***

Figure 25: Confident bilateral failure (idx 388): the model identifies all key pathological features and concludes they are present, but the ground truth is negative; both prompt and probe agree on the wrong answer, indicating representational incapacity for this class of fine-grained bowel pathology.

As shown in Figure 25, the prompted model explicitly identifies the key pathological features (thickened wall, narrowed lumen, skip areas) and concludes they are present, but the ground truth says they are not. The probe also predicts incorrectly. Both methods agree on the wrong answer, suggesting that either the visual features being described are genuinely ambiguous in this image, or the model is generating a plausible narrative without reliable visual grounding. This is a **confident bilateral failure**: neither elicitation mechanism provides a corrective signal, because the underlying visual discrimination required is beyond the model's current representational capacity for this class of fine-grained bowel pathology.

H.3 Where MedProb Underperforms Prompting

MedProb underperforms prompting on questions that benefit from explicit semantic reasoning, medical world knowledge, or direct visual labeling naturally expressible in language. Prompting succeeds on these cases by leveraging the model's broad medical knowledge base, composing pathological features into a clinical judgment, naming anatomy from contextual cues, or reasoning step-by-step from visible findings to a diagnosis. A linear probe is a shallow decoder capable only of recovering information that is *linearly separable* in a single layer's activation space. Compositional reasoning, clinical inference chains, and world-knowledge-driven interpretation are fundamentally nonlinear and distributed, precisely the features a linear probe is least equipped to capture. As a result, the probe performs worse on questions where the correct answer requires integrating multiple visible features into a high-level clinical concept.

Table 13: Examples where prompting succeeds but MedProb fails (**Probe** = ×, **Prompt** = ✓).

Ex.	Modality / Task	GT	Prompt	Probe	Failure Pattern
E-5	Lung histology (coarse tissue category recognition)	A (Yes)	A ✓	B ×	Direct semantic label, probe misses coarse class
E-6	Gross pathology (cardiac CHF classification)	A (Yes)	A ✓	B ×	Compositional clinical reasoning required

Example E-5

Prompt model output — idx 190

"The image clearly shows lung tissue. So the answer is A."

Prompt prediction: ***A (Yes)*** ✓ *Probe prediction:* ***B (No)*** × *GT:* ***A (Yes)***

Figure 26: Coarse semantic label failure (idx 190): prompting trivially names the tissue in one sentence and is correct; the probe fails on the same question, suggesting coarse semantic categories may not be linearly separable in intermediate representation layers.

Figure 26 is the starkest example in the dataset. The prompt requires a single grounded visual label, produces it in one sentence, and is correct. Yet the probe fails on this same question. The brevity of the correct output is the key observation: when the answer reduces to "name the tissue you see," language generation has a direct pathway to the correct answer that the linear probe, a single affine transformation over one layer's activations, does not reliably replicate. This suggests that **coarse semantic category labels** (lung, liver, kidney) may not be linearly separable in the intermediate representation layers probed, even when the language model can verbalize them trivially. The probe is decoding at the wrong abstraction level for this class of question.

Example E-6

Prompt model output — idx 209

"Let's think. The image depicts a heart with significant pathological changes, including thickening of the ventricular walls and the presence of fibrosis. These features are consistent with the pathology of congestive heart failure (CHF), where the heart muscle becomes enlarged and less efficient in pumping blood. The dilated chambers and the presence of fibrotic tissue are indicative of the chronic changes that occur in the heart due to the inability to pump blood effectively, leading to a build-up of fluid in the body. Therefore, based on the visible characteristics in the image, it can be concluded that the heart shown is indeed affected by congestive heart failure. The answer is: [yes]"

Prompt prediction: ***A (Yes)*** ✓ *Probe prediction:* ***B (No)*** × *GT:* ***A (Yes)***

Figure 27: Compositional reasoning failure (idx 209): the prompted model correctly chains wall thickening → fibrosis → dilated chambers → CHF; the probe cannot replicate this nonlinear multi-step clinical judgment from a single hidden layer.

Figure 27 illustrates the maximum expressiveness gap between prompting and probing. The prompted model performs a multi-step compositional inference chain: ventricular wall thickening → fibrosis → dilated chambers → CHF diagnosis. Each step draws on medical world knowledge to connect a visual feature to a pathophysiological mechanism, and the chain is correct. A linear probe operating on a single hidden layer cannot integrate these features into a CHF judgment in the same way. Doing so would require the concept "CHF as a clinical gestalt" to be linearly separable in that layer's activation space, which it evidently is not for this model. This case exemplifies the fundamental limitation of probing for questions that require **compositional clinical reasoning**: the information needed to answer correctly is distributed nonlinearly across the model's computation, making it inaccessible to a shallow linear decoder.

H.4 Summary

Table 14 provides a consolidated view of all six examples across the three behavioral categories. Taken together, these cases suggest that the choice between probing and prompting is not simply a question of which approach is generally superior, but depends critically on *what type of inference is required*: probing outperforms prompting when language generation introduces spurious reasoning, while prompting outperforms probing when the correct answer requires compositional knowledge or grounded semantic labeling. Questions where both fail point to genuine perceptual limitations of the underlying vision-language model that neither elicitation strategy can overcome.

Cat.	Ex.	Idx	GT	Prompt	Probe	Key Failure Mode
P>M	E-1	440	B	× A	✓ B	Hallucinated pathology narrative
	E-2	383	B	× A	✓ B	Textbook rule misapplied
Both×	E-3	223	A	× B	× B	Ring-enhancement too subtle
	E-4	388	B	× A	× A	Bowel skip-lesion ambiguity
M>P	E-5	190	A	✓ A	× B	Coarse semantic label inaccessible to probe
	E-6	209	A	✓ A	× B	CHF requires compositional reasoning

Table 14: Consolidated summary of all six error analysis examples. Category abbreviations: **P>M** = Probe better than prompting; **Both×** = Both fail; **M>P** = Prompting better than probe.

H Probing vs. Prompting Across Medical Imaging Modalities

Table 15 reports accuracy for each model under prompting and probing across eight medical imaging modalities on the OmniMedVQA evaluation set. Probing consistently and substantially outperforms prompting across virtually all modalities and model families, with the advantage being most dramatic for larger general-purpose models. Qwen2.5-VL-32B is the starkest example: its prompting average collapses to **10.0%**, with near-zero scores on MRI (**2.0%**) and Ultrasound (**0.0%**), whereas probing recovers a strong average of **79.0%** on the same frozen model. A nearly identical pattern holds for Qwen3-VL-30B, whose prompting average of **6.0%** contrasts with a probing average of **88.0%**, a gap of 82 percentage points, confirming that the clinical knowledge encoded in large-scale general-purpose models is severely suppressed by free-text generation. Similarly, LLaVA-7B achieves a prompting average of only **10.0%** yet reaches **51.0%** under probing, and Open-Flamingo-9B improves from **28.0%** to **66.0%**. The probing advantage is consistent across modalities: even in domains typically considered visually demanding, such as Dermoscopy and OCT, probing reliably extracts clinically relevant signal that prompting fails to surface. The principal exception in the table is MedMo-8B, whose prompting average of **99.0%** is the highest across all conditions, a result that likely reflects the specific prompt format and dataset overlap with its medical training corpus rather than a general superiority of generation-based decod-

Model	Approach	Modality								
		CT	Dermoscopy	Fundus Photography	Microscopy Images	MRI	OCT	Ultrasound	X-Ray	Average
Open-Flamingo-9B	Prompting	18.0	26.0	36.0	30.0	26.0	26.0	24.0	40.0	**28.0**
	Probing	62.0	58.0	76.0	56.0	60.0	66.0	**82.0**	64.0	**66.0**
Med-Flamingo-9B	Prompting	26.0	20.0	28.0	36.0	40.0	38.0	40.0	36.0	33.0
	Probing	**80.0**	64.0	**80.0**	60.0	64.0	76.0	76.0	72.0	**72.0**
InternVL3-1B	Prompting	62.0	70.0	68.0	74.0	74.0	46.0	76.0	72.0	68.0
	Probing	**84.0**	74.0	76.0	70.0	78.0	62.0	74.0	78.0	**75.0**
BioMed-InternVL3-1B	Prompting	48.0	30.0	32.0	24.0	30.0	**8.0**	32.0	28.0	29.0
	Probing	62.0	50.0	60.0	50.0	50.0	44.0	50.0	56.0	53.0
LLaVA-7B	Prompting	12.0	16.0	8.0	**4.0**	**2.0**	10.0	12.0	12.0	**10.0**
	Probing	52.0	38.0	64.0	50.0	50.0	50.0	56.0	46.0	**51.0**
LLaVA-Med-7B	Prompting	20.0	20.0	16.0	14.0	10.0	24.0	10.0	8.0	15.0
	Probing	48.0	46.0	54.0	48.0	50.0	50.0	58.0	50.0	51.0
Qwen2-VL-2B	Prompting	64.0	62.0	66.0	64.0	66.0	64.0	58.0	76.0	65.0
	Probing	82.0	74.0	78.0	68.0	70.0	74.0	62.0	**82.0**	**74.0**
BioMed-Qwen2-VL-2B	Prompting	70.0	52.0	54.0	58.0	58.0	42.0	48.0	64.0	56.0
	Probing	70.0	52.0	78.0	60.0	50.0	54.0	54.0	60.0	60.0
Qwen2.5-VL-3B	Prompting	70.0	72.0	66.0	62.0	66.0	78.0	30.0	80.0	66.0
	Probing	80.0	**84.0**	**84.0**	76.0	70.0	**88.0**	58.0	82.0	**78.0**
MedVLThinker-RL-3B	Prompting	58.0	70.0	66.0	64.0	66.0	66.0	36.0	82.0	64.0
	Probing	74.0	68.0	80.0	64.0	70.0	70.0	56.0	80.0	70.0
Qwen2.5-VL-32B	Prompting	**8.0**	22.0	14.0	**4.0**	**2.0**	**6.0**	**0.0**	24.0	**10.0**
	Probing	**86.0**	80.0	**86.0**	76.0	76.0	78.0	72.0	80.0	**79.0**
MedVLThinker-RL-32B	Prompting	60.0	62.0	62.0	74.0	62.0	70.0	36.0	78.0	63.0
	Probing	68.0	68.0	82.0	64.0	70.0	70.0	72.0	76.0	71.0
Qwen3-VL-2B	Prompting	50.0	76.0	70.0	68.0	52.0	48.0	80.0	72.0	65.0
	Probing	82.0	78.0	**90.0**	**86.0**	76.0	70.0	**90.0**	**86.0**	**82.0**
MediX-R1-2B	Prompting	64.0	78.0	80.0	78.0	76.0	62.0	**88.0**	72.0	**75.0**
	Probing	70.0	72.0	84.0	76.0	58.0	72.0	84.0	76.0	74.0
Qwen3-VL-8B	Prompting	68.0	68.0	66.0	56.0	76.0	58.0	50.0	52.0	62.0
	Probing	**84.0**	**82.0**	**92.0**	**82.0**	**82.0**	**84.0**	**82.0**	**84.0**	**84.0**
MedMo-8B	Prompting	**98.0**	**98.0**	**100.0**	**98.0**	**100.0**	**100.0**	**100.0**	**100.0**	**99.0**
	Probing	92.0	98.0	**100.0**	98.0	96.0	**100.0**	**100.0**	94.0	97.0
Qwen3-VL-30B	Prompting	**6.0**	**6.0**	12.0	**4.0**	**0.0**	**2.0**	**0.0**	18.0	**6.0**
	Probing	**86.0**	**90.0**	**90.0**	**84.0**	**88.0**	82.0	**90.0**	**94.0**	**88.0**
MediX-R1-30B	Prompting	72.0	64.0	82.0	74.0	**84.0**	72.0	76.0	80.0	**76.0**
	Probing	62.0	62.0	72.0	58.0	64.0	64.0	70.0	70.0	65.0

Table 15: Prompting and probing accuracy (%) across eight medical imaging modalities on OmniMedVQA. Bold values highlight the most striking contrasts between the two paradigms, including near-zero prompting scores for large general-purpose models and the rare cases where prompting matches or exceeds probing.

ing. MediX-R1-30B also reverses the trend, with prompting (**76.0%**) exceeding probing (**65.0%**), consistent with the observation in Section 5.4 that reinforcement-learning-based adaptation can alter internal representations in ways that reduce linear separability at the layer level. Taken together, these modality-level results reinforce the central conclusion of the paper: the bottleneck in medical VQA lies in how knowledge is elicited, not in whether the model encodes it, and this failure of prompted generation is pervasive across all eight imaging modalities examined.

I Baseline Details

This section describes the baseline systems evaluated in our study, covering both multi-agent frameworks (§I.1) and specialized medical vision-language models (§I.2). All our experiments are performed on a server equipped with 2 NVIDIA H200 GPUs (140GB memory each) and an Intel(R)

Xeon(R) Gold 6442Y CPU.

I.1 Multi-Agent Frameworks

We evaluate three multi-agent frameworks that coordinate multiple vision-language agents for medical visual question answering.

MAM (Zhou et al., 2025). MAM (Modular Multi-Agent Framework) decomposes the medical diagnostic process into specialized roles, General Practitioner, Specialist Team, Radiologist, Medical Assistant, and Director, each embodied by a separate LLM-based agent. In our experiments, all agents are instantiated using **HuatuoGPT-Vision-7B** (Chen et al., 2024a).[34]

MMedAgent (Li et al., 2024). MMedAgent (Multi-modal Medical Agent) is the first agent framework designed specifically for the medical domain, enabling tool selection across six medical tasks spanning five modalities. The backbone vision-language model used by MMedAgent is **LLaVA-Med** (Li et al., 2023).[35]

UCAgents (Qianhan Feng, 2025). UCAgents employs a unidirectional convergence mechanism for visual evidence-anchored multi-agent medical decision-making. In our experiments, we instantiate UCAgents with **Qwen2.5-VL-72B-Instruct** (Bai et al., 2025b), which yielded the best performance among the model variants they evaluated.[36]

I.2 Medical Vision-Language Models

We evaluate four specialized Med-VLMs: InfiMed-RL-3B (Liu et al., 2025b), UniMedVL-14B (Ning et al., 2025), HuatuoGPT-Vision-34B (Chen et al., 2024a), and Aloe-Vision-72B-AR (Guasch-Martí et al., 2026). Following the official documentation and recommended configurations of each model, we describe the inference hyperparameters used for each below.

InfiMed-RL-3B (Liu et al., 2025b). InfiMed-RL-3B is built on top of Qwen2.5-VL-3B and runs in bfloat16 precision. We use greedy decoding, meaning the model always selects the most probable next token at each step, and cap generation at 256 tokens. Input image resolution follows the processor defaults recommended in the official documentation, with a minimum of 256×28^2 pixels and a maximum of 1280×28^2 pixels.

UniMedVL-14B (Ning et al., 2025). UniMedVL-14B is run in bfloat16 precision with greedy decoding and a maximum generation length of 128 tokens. A fixed random seed is set for full reproducibility, following the official inference configuration.

HuatuoGPT-Vision-34B (Chen et al., 2024a). HuatuoGPT-Vision-34B is loaded in float16 precision. Unlike the other Med-VLMs evaluated, it uses stochastic sampling rather than greedy decoding: the temperature is set to 0.2, which slightly sharpens the output distribution, and a repetition penalty of 1.2 is applied to discourage repetitive outputs. Generation is capped at 256 tokens. These settings correspond to the default generation configuration provided in the official model interface.

Aloe-Vision-72B-AR (Guasch-Martí et al., 2026). Aloe-Vision-72B-AR is run in bfloat16 precision with greedy decoding and a maximum of 32 new tokens, following the generation configuration recommended in the official model documentation.

I.3 Fine-Tuned Models

We fine-tuned two vision-language models: Qwen2.5-VL-7B-Instruct and Llama-3.2-11B-Vision-Instruct. Each model was fine-tuned in four settings: three single-domain settings using PATH-VQA, SLAKE, and VQA-RAD separately, and one multi-domain setting using the combined training data from all three benchmarks. This setup lets us compare in-domain specialization against broader medical adaptation under a consistent fine-tuning pipeline.

All runs used full fine-tuning rather than probe training or adapter-only updates, and the supervision format matched the repository's standard medical VQA prompting setup. Inputs were formatted as image-plus-question prompts with multiple-choice options, and the model was trained to generate the gold answer. Optimization used fused AdamW with learning rate 2×10^{-5}, weight decay 0.01, warmup ratio 0.03, bf16 precision, seed 42, and an effective batch size of 8. For Llama-3.2-11B-Vision-Instruct, we used per-device batch size 1 with gradient accumulation 8, while for Qwen2.5-VL-7B-Instruct, we used per-device batch size 8 without gradient accumulation.

[34] https://aclanthology.org/2025.findings-acl.1298.pdf#page=3

[35] https://aclanthology.org/2024.findings-emnlp.510.pdf#page=1

[36] https://arxiv.org/pdf/2512.02485#page=7

J Probe Ablation Details

J.1 Nonlinear Probe (MLP)

A natural question is whether the linear constraint of logistic regression is the bottleneck. We trained a small MLP probe on the same frozen embeddings for three representative models. The MLP does not consistently improve over logistic regression: it raises average accuracy for only one of the two models (Qwen3-VL-8B: 81.4% → 81.6%), while Llama-3.2-11B-Vision is unchanged or lower (81.3% → 80.9%). These results indicate that the clinical knowledge relevant to our benchmarks is largely linearly separable within a single transformer layer's activation space, validating logistic regression as the appropriate and parsimonious probe architecture. Table 16 provides full per-dataset results.

Model	Log. Acc.	MLP Acc.	Log. F1	MLP F1
Llama-3.2-11B-Vision	81.3	80.9	85.9	81.1
Qwen3-VL-8B	81.4	81.6	82.3	71.3

Table 16: Average accuracy and macro-F1 (%) across PATH-VQA, SLAKE, and VQA-RAD for the linear logistic probe and the nonlinear MLP probe.

J.2 PATH-VQA Sampling Sensitivity

PATH-VQA results throughout the paper are based on a stratified 500-example sample from the full 3,389-example test set. To confirm that our conclusions are not sensitive to a particular random draw, we repeated the evaluation with three independently resampled subsets (seeds 42, 43, 44). Accuracy standard deviations are small across all models: LLaVA-V0-7B 87.1 ± 0.9, Llama-3.2-11B-Vision 89.9 ± 0.5, Qwen3-VL-8B 91.7 ± 0.6. Best layers vary modestly across resamples (e.g., layers 16–27 for LLaVA-V0-7B), but the rankings among models remain stable. Table 17 reports best-layer probing accuracy on three independently resampled subsets, confirming that the stratified 500-example protocol is a stable proxy for the full test distribution.

Model	Acc.	Macro-F1	Best Layers
LLaVA-V0-7B	87.1 ± 0.9	87.0 ± 0.9	18 / 16 / 27
Llama-3.2-11B-Vision	89.9 ± 0.5	89.8 ± 0.5	20 / 13 / 23
Qwen3-VL-8B	91.7 ± 0.6	91.6 ± 0.6	21 / 25 / 23

Table 17: PATH-VQA sampling sensitivity: mean ± std across seeds 42, 43, and 44. Best Layers lists the optimal layer per seed in the same order.

K Probing vs. Prompting Across All Models and Benchmarks

Figures 28–30 report probing and prompting accuracy for all 24 evaluated models on PATH-VQA, SLAKE, and VQA-RAD. Across all three benchmarks, probing consistently and substantially outperforms prompting, confirming the main finding of the paper at scale: models encode clinically relevant knowledge that their generation-based interface fails to elicit.

On PATH-VQA (Figure 28), the probing advantage is universal and often dramatic. Med-Flamingo-9B exhibits the largest gap, with probing reaching 85.4% against a prompting accuracy of only 41.8%, a margin of 43.6 points. OpenFlamingo-9B follows a similar pattern (85.4% vs. 53.0%), confirming that Flamingo-family models are particularly susceptible to elicitation failure despite encoding strong medical signal. BioMed-Llama-3.2-11B achieves the highest probing accuracy at 88.4%, while MedMO-4B-Next leads under prompting at 69.4% — the only model that partially closes the gap between the two paradigms on this benchmark.

On SLAKE (Figure 29), the probing advantage persists across all models, but the prompting scores are generally higher than on PATH-VQA, suggesting that SLAKE questions are somewhat more amenable to generation-based decoding. Med-Flamingo-9B again shows the largest collapse under prompting (36.87%), compared to a probing accuracy of 79.52%. MedMO-4B-Next and MedMO-8B-Next both reach the highest probing scores at 90.12%, and MedMO-4B-Next also achieves the highest prompting score at 81.69%, nearly halving the typical gap. Among larger models, MedVLThinker-RL-32B (85.78% probing, 73.98% prompting) and Qwen2.5-VL-32B (87.47% probing, 77.59% prompting) show that reinforcement-learning-based training and scale can simultaneously improve both the richness of internal representations and their accessibility through prompting.

VQA-RAD (Figure 30) is the benchmark where the prompting gap narrows most for top-performing models. Qwen2.5-VL-32B achieves the highest probing accuracy at 88.59%, followed closely by MedVLThinker-RL-32B at 86.31%; the corresponding prompting scores are 82.51% and 77.19%, respectively, representing more modest gaps than are observed on the other two bench-

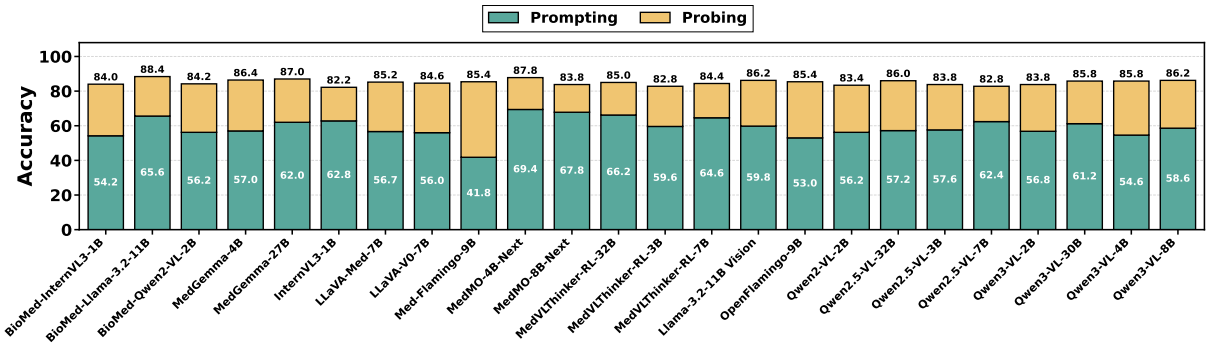

Figure 28: Probing vs. prompting accuracy (%) on PATH-VQA for all 24 evaluated models, sorted by probing score.

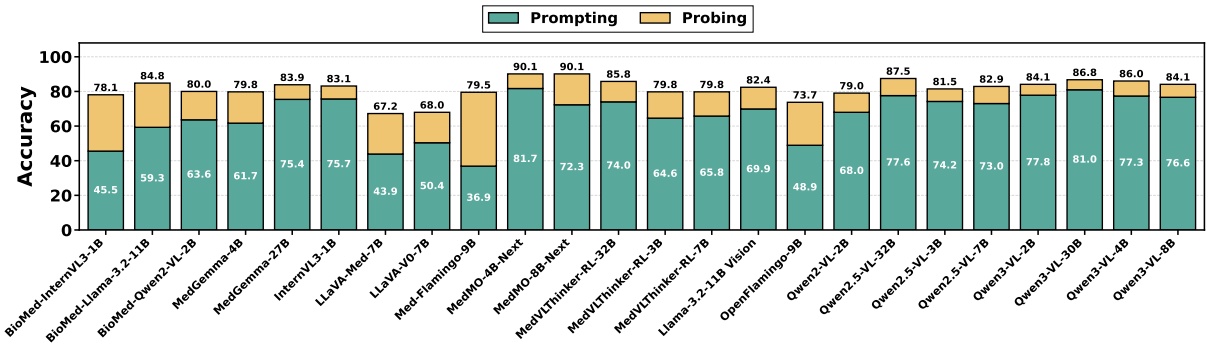

Figure 29: Probing vs. prompting accuracy (%) on SLAKE for all 24 evaluated models, sorted by probing score.

marks. Notably, MedMO-4B-Next is the single model where prompting (76.05%) marginally exceeds probing (73.76%) on VQA-RAD, a rare reversal that mirrors the MedMo-8B exception observed in the modality analysis (§15) and again points to dataset-specific prompt format alignment rather than a general superiority of generation-based decoding. Flamingo-family models (Med-Flamingo-9B, OpenFlamingo-9B) and older architectures (LLaVA-V0-7B, LLaVA-Med-7B) continue to show the largest probing-to-prompting gaps, confirming that elicitation failure is most severe in models whose training did not explicitly optimize for structured VQA generation.

Taken together, these per-benchmark figures reinforce the central conclusion at full model scale: probing reliably recovers latent clinical knowledge that prompted generation cannot surface, and this advantage holds across diverse model families, parameter counts, and medical VQA benchmarks.

L Use of AI Assistants

AI-based tools were used to assist with language refinement and clarity. All ideas, analyses, and conclusions presented in this work are solely those of the authors.

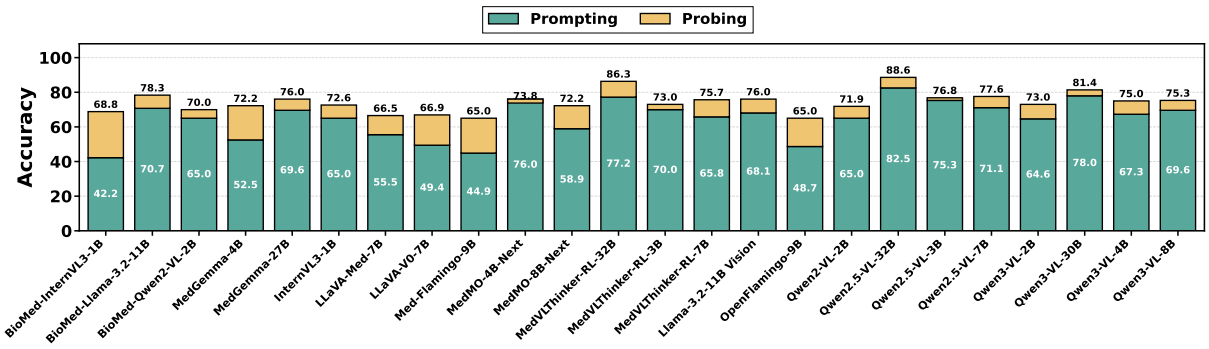

Figure 30: Probing vs. prompting accuracy (%) on VQA-RAD for all 24 evaluated models, sorted by probing score.