One VLM, Two Roles: Stage-Wise Routing and Specialty-Level Deployment for Clinical Workflows

Shayan Vassef SHAYAN@NIMBLEMIND.AI
Nimblemind, USA

Soorya Ram Shimgekar SOORYA@NIMBLEMIND.AI
Nimblemind, USA

Abhay Goyal ABHAY@NIMBLEMIND.AI
Nimblemind, USA

Christian Poellabauer CPOELLAB@FIU.EDU
Florida International University, FL, USA

Koustuv Saha KSAHA2@ILLINOIS.EDU
University of Illinois - Urbana Champaign, IL, USA

Pi Zonooz PI@NIMBLEMIND.AI
Nimblemind, USA

Navin Kumar NAVIN@NIMBLEMIND.AI
Nimblemind, USA

Abstract

Clinical ML workflows are often fragmented and inefficient: triage, task selection, and model deployment are handled by a patchwork of task-specific networks. These pipelines are rarely aligned with data-science practice, reducing efficiency and increasing operational cost. They also lack data-driven model identification (from imaging/tabular inputs) and standardized delivery of model outputs. We present a framework that employs a *single* vision–language model (VLM) in two complementary, modular roles. **First (Solution 1):** the VLM acts as an aware model-card matcher that routes an incoming image to the appropriate specialist model via a three-stage workflow (modality → primary abnormality → model-card ID). Reliability is improved by (i) stage-wise prompts enabling early termination via `None`/`Other` and (ii) a calibrated top-2 answer selector with a stage-wise cutoff. This raises routing accuracy by **+9** and **+11** percentage points on the training and held-out splits, respectively, compared with a baseline router, and improves held-out calibration (lower ECE). **Second (Solution 2):** we fine-tune the same VLM on specialty-specific datasets so that one model per specialty covers multiple downstream tasks, simplifying deployment while maintaining performance. Across gastroenterology, hematology, ophthalmology, pathology, and radiology, this single-model deployment **matches or approaches** specialized baselines.

Together, these solutions reduce data-science effort through more accurate selection, simplify monitoring and maintenance by consolidating task-specific models, and increase transparency via per-stage justifications and calibrated thresholds. Each solution stands alone, and in combination they offer a practical, modular path from triage to deployment.

Keywords: Healthcare AI, Vision–Language Models, MedGemma, Model Cards, MLOps, Early termination, Confidence thresholds, Selective prediction

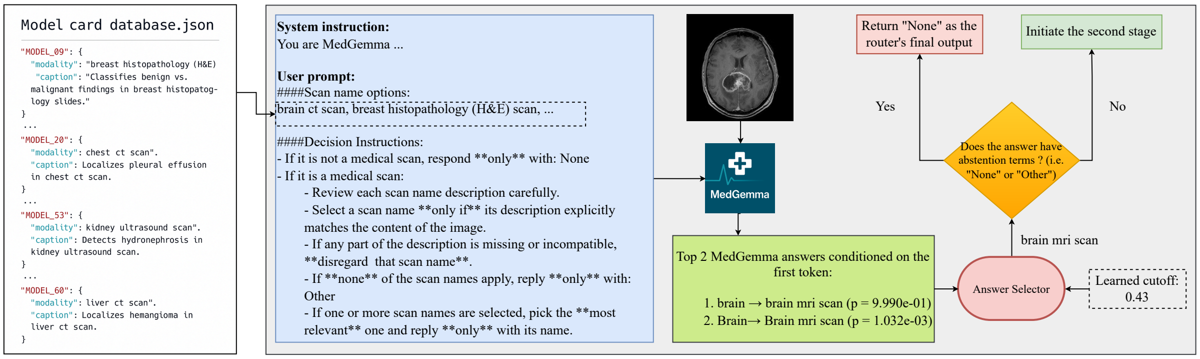
Data and Code Availability Data and code are available from the authors at reasonable request.

Institutional Review Board (IRB) All data is publicly available and de-identified; the study is IRB-exempt.

1. Introduction

Machine learning (ML) models in healthcare face significant operational friction when moving from research prototypes to clinical practice. An estimated

© 2025 S. Vassef, S.R. Shimgekar, A. Goyal, C. Poellabauer, K. Saha, P. Zonooz & N. Kumar.



```
Model card database.json
"MODEL_09": {
  "modality": "breast histopathology (H&E)
   "caption": "Classifies benign vs.
  malignant findings in breast histopatog-
  logy slides."
}
 ...
"MODEL_20": {
  "modality": chest ct scan".
  "caption": Localizes pleural effusion
  in chest ct scan.
}
 ...
"MODEL_53": {
  "modality": kidney ultrasound scan".
  "caption": Detects hydronephrosis in
  kidney ultrasound scan.
}
 ...
"MODEL_60": {
  "modality": liver ct scan".
  "caption": Localizes hemangioma in
  liver ct scan.
}
```

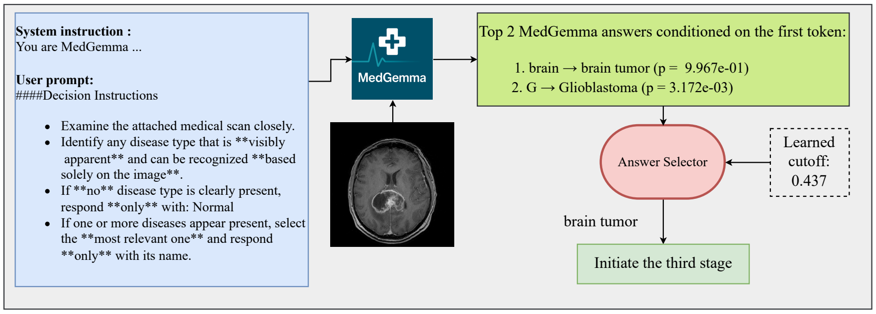
```
System instruction:
You are MedGemma ...

User prompt:
####Scan name options:
brain ct scan, breast histopathology (H&E) scan, ...

####Decision Instructions:
- If it is not a medical scan, respond **only** with: None
- If it is a medical scan:
      - Review each scan name description carefully.
      - Select a scan name **only if** its description explicitly
      matches the content of the image.
      - If any part of the description is missing or incompatible,
      **disregard  that scan name**.
      - If **none** of the scan names apply, reply **only** with:
      Other
      - If one or more scan names are selected, pick the **most
      relevant** one and reply **only** with its name.
```

Top 2 MedGemma answers conditioned on the first token:

1. brain → brain mri scan (p = 9.990e-01)
2. Brain→ Brain mri scan (p = 1.032e-03)

MedGemma
Answer Selector
Learned cutoff: 0.43
brain mri scan
Does the answer have abstention terms ? (i.e. "None" or "Other")
Yes
No
Return "None" as the router's final output
Initiate the second stage

Figure 1: The first stage in the router workflow (modality detection).



```
System instruction :
You are MedGemma ...

User prompt:
####Decision Instructions

  • Examine the attached medical scan closely.
  • Identify any disease type that is **visibly
     apparent** and can be recognized **based
    solely on the image**.
  • If **no** disease type is clearly present,
    respond **only** with: Normal
  • If one or more diseases appear present, select
    the **most relevant one** and respond
    **only** with its name.
```

Top 2 MedGemma answers conditioned on the first token:

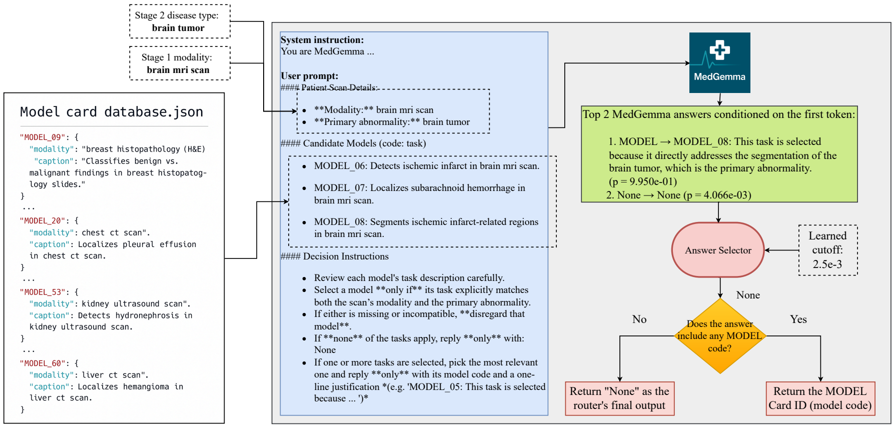
1. brain → brain tumor (p = 9.967e-01)
2. G → Glioblastoma (p = 3.172e-03)

Figure 2: The second stage in the router workflow (disease identification).

90% of medical AI models never reach clinical deployment Poddar et al. (2024), and production-ready models can take weeks to months to integrate, with over half of organizations reporting 8–90 days to deploy a single model Mage AI (2025). These delays stem from several bottlenecks, and we provide two key examples here. First, to select an appropriate model design (architecture, training objective, etc.) for a given dataset, the data scientist must identify the attributes that guide model design. In tabular data these attributes are explicit; in clinical images they are latent in the pixels—such as *modality* (e.g., CT) and *clinical indication* (e.g., acute intracranial hemorrhage)—and must first be extracted, standardized, and quality-checked before they can inform design choices. As a result, streamlined model selection for clinical imaging is often overlooked and warrants dedicated attention Willemink et al. (2020). Second, health systems tend to integrate, validate, monitor, and recalibrate AI solutions one by one, so developing and maintaining task-specific models multiplies the cost burden and slows adoption Brady et al. (2024).

We address these operational issues with two linked—but independently evaluated—solutions: the first assists data scientists in selecting the appropriate model, and the second shortens time-to-deployment by unifying task-specific AI models into broader, specialty-level models that support multiple tasks. While Solution 1 could inform Solution 2 in future systems, *our implementation treats them separately*; Solution 2 does not consume outputs from Solution 1.

Solution 1: Model-card matching Early medical vision–language efforts aligned images and text to enable cross-modal understanding and transfer. Recently, MedGemma Sellergren et al. (2025) and UMIT Yu et al. (2025) demonstrated that medical VLMs can approach specialized systems while maintaining generality, motivating workflows that both interpret the input and decide what to do next. Accordingly, we adopt a medical VLM as an aware, auditable selector that reads an input image and chooses an appropriate model card via a three-stage pipeline (MODALITY → PRIMARY ABNORMALITY → MODEL-CARD ID). At each stage, the VLM proposes top can-



```
"MODEL_09": {
  "modality": "breast histopathology (H&E)
   "caption": "Classifies benign vs.
  malignant findings in breast histopatog-
  logy slides."
}
 ...
"MODEL_20": {
  "modality": chest ct scan".
  "caption": Localizes pleural effusion
  in chest ct scan.
}
 ...
"MODEL_53": {
  "modality": kidney ultrasound scan".
  "caption": Detects hydronephrosis in
  kidney ultrasound scan.
}
 ...
"MODEL_60": {
  "modality": liver ct scan".
  "caption": Localizes hemangioma in
  liver ct scan.
}
```

```
System instruction:
You are MedGemma ...

User prompt:
#### Patient Scan Details:
  • **Modality:** brain mri scan
  • **Primary abnormality:** brain tumor
#### Candidate Models (code: task)
  • MODEL_06: Detects ischemic infarct in brain mri scan.
  • MODEL_07: Localizes subarachnoid hemorrhage in
    brain mri scan.
  • MODEL_08: Segments ischemic infarct-related regions
    in brain mri scan.
#### Decision Instructions
  • Review each model's task description carefully.
  • Select a model **only if** its task explicitly matches
    both the scan’s modality and the primary abnormality.
  • If either is missing or incompatible, **disregard that
    model**.
  • If **none** of the tasks apply, reply **only** with:
    None
  • If one or more tasks are selected, pick the most relevant
    one and reply **only** with its model code and a one-
    line justification *(e.g. 'MODEL_05: This task is selected
    because ... ')*
```

1. MODEL → MODEL_08: This task is selected because it directly addresses the segmentation of the brain tumor, which is the primary abnormality. (p = 9.950e-01)
2. None → None (p = 4.066e-03)

Answer Selector
Learned cutoff: 2.5e-3
None
No
Does the answer include any MODEL code?
Yes
Return "None" as the router's final output
Return the MODEL Card ID (model code)

Figure 3: The third (final) stage in the router workflow (model-card matching).

didates, and an answer selector applies a calibrated threshold that either selects the most likely candidate (our baseline) or, if warranted, the second most likely candidate. Decisions at each stage (VLM output, any abstention, and routing to the next stage) are visible to users and logged in a consistent format, yielding transparent selection tied to model-card semantics Mitchell et al. (2019) and safe selective prediction Geifman and El-Yaniv (2019). This process accelerates development: we posit that choosing the appropriate model for a given clinical image hinges on two key pieces of information—imaging modality and disease type—details that are not always apparent to data scientists.

Solution 2: Specialty-specific deployment Prior work shows that domain-pretrained medical encoders, when fine-tuned on target tasks, can match strong baselines across multiple datasets Mei et al. (2022); Zhang et al. (2022). In healthcare, specialty-level foundation models have also demonstrated broad coverage within a specialty—outperforming narrow, dataset-specific models across many tasks (e.g., pathology’s Prov-GigaPath achieves SOTA on 25/26 tasks across 31 tissue types) Xu et al. (2024). Operationally, fewer, broader specialty models reduce the number of artifacts a health system must validate, secure, and monitor, lowering deployment burden compared with one-model-per-dataset estates Brady et al. (2024). In line with past work, we fine-tune the *same* calibrated VLM at the *specialty* level—gastroenterology, hematology, ophthalmology, pathology, and radiology—rather than per dataset. Our goal is to match task-specific model performance while keeping everything on a single, easy-to-maintain deployment stack. To adapt the VLM to multiple AI tasks, one must include task-specific prompts in the VLM’s training data (e.g., a task-specific user prompt for colonoscopy lesion classification and a task-specific user prompt for polyp bounding-box detection) to build a specialty model that supports both classification and bounding-box (BBox) prediction. Importantly, in our experiments, *Solution 2 is trained and evaluated independently of Solution 1.*

Taken together, these solutions connect selection and deployment in a calibrated workflow—a VLM that can *route and execute.* Compared with recent multi-agent pipelines (Shimgekar et al., 2025a,b), this minimalist design may reduce data-science workload and human error. As an example, AI-driven triage has been shown to cut chest-radiograph reporting delays from 11.2 to 2.7 days in simulation while maintaining diagnostic performance Annarumma et al. (2019). Similarly, our approach may shorten the path from triage to clinical use and increase throughput without sacrificing performance.

2. Related Work

Automated matching in healthcare Selecting the appropriate model based on a dataset has been explored at multiple layers of care delivery, such as patient-facing systems, specialty care, etc. Patient-facing systems (e.g., symptom checkers like Buoy Health buoy health (2025)) narrow likely conditions and suggest next steps. Platforms such as Garner

Health Garner (2025) match patients to specialists using outcomes data. Similarly, LLM-driven controllers (HuggingGPT Shen et al. (2023)) show how a coordinator can parse requests, consult model descriptions, and delegate to expert models. This pattern maps naturally to healthcare *model cards* as standardized summaries of intended use and performance Mitchell et al. (2019). Clinical imaging "orchestration" seeks to run the appropriate algorithm on the selected scan Aidoc (2024). However, prior work does not provide calibrated, stage-wise routing that is explicitly grounded in model-card semantics, with visible abstention and early termination and per-stage decision logs for audit. In the first solution, we address this gap with a three-stage router that makes every decision—VLM output, abstention, and route—user-visible and logged, cutting routing errors while preserving safety via careful prompt design and stage-specific selection logic.

Adaptation across tasks A parallel line of work reduces packaging and integration friction for *multiple* models: MONAI Project MONAI (2023) deploys standardized applications, packaging and interfaces, and self-configuring pipelines like nnU-Net Isensee et al. (2021) that adapt architectures to new datasets with minimal manual tuning. In practice, however, maintaining one network per task is inefficient: each model has to be validated, integrated, monitored, and updated separately at the site level, duplicating effort and slowing adoption Boverhof et al. (2024); Brady et al. (2024). Multi-agent or model-zoo deployments likewise increase the number of components to secure, monitor, and recalibrate over time, compounding technical debt Sculley et al. (2015). Unlike UMIT Yu et al. (2025)—which initializes Qwen2-VL Wang et al. (2024) and follows a two-stage pipeline (full-parameter feature alignment followed by instruction fine-tuning) to reach competitive results across many datasets and modalities—there is no evidence that a single calibrated VLM can be *reused* across specialties within one unified deployment stack where onboarding a new task changes only the *user prompt*. Our second solution targets this gap: we fine-tune one medical VLM across datasets within a specialty so a single component serves multiple imaging tasks, keeping interfaces and calibration fixed while matching or approaching benchmark performance.

3. Solution 1: Model–Card Matching with Early Termination

To integrate the medical VLM as an automated model selector, we employ a three-stage prompting pipeline. At each stage, the model receives a stage-specific instruction and context and returns a textual answer. The stages progressively narrow the information needed to select the appropriate model card. After each response, an **answer-selection** step ranks candidate answers by likelihood, retains the top two, and chooses between them; for Stage 1 only, it may also terminate early. Figures 1, 2, and 3 illustrate the overall workflow in order; Algorithm 1 provides pseudocode for the answer-selection strategy; and Algorithm 2 explains how we derive stage-wise cutoffs that parameterize Algorithm 1. Below, we first outline the prerequisites and then describe the selection process for each stage.

Prerequisite 1: Backbone VLM choice In Section 1 (Solution 1) we described a generic medical VLM; here, we *adopt* MedGemma. MedGemma was pretrained on a multimodal medical mixture spanning radiology (CXR/CT/MRI), ophthalmology, dermatology, and histopathology[1] and reports competitive benchmarks relative to prior work Sellergren et al. (2025).

Prerequisite 2: Model-card repository To match a clinical image to the correct model, we maintain a diverse repository of model cards spanning multiple modalities (e.g., CT, MRI) and AI tasks (e.g., classification, grading). This repository currently includes **84** model cards over **26** unique modalities: brain CT scan, brain MRI scan, breast histopathology (H&E) scan, breast mammography scan, breast ultrasound scan, cervix cytology scan, cervix histology scan, cervix histopathology (H&E) scan, chest CT scan, colonoscopy scan, colon histopathology (H&E) scan, esophageal gastroscopy scan, fundus photography scan, gallbladder ultrasound scan, kidney CT scan, kidney histopathology scan, kidney MRI scan, kidney ultrasound scan, liver CT scan, liver ultrasound scan, lung histopathology (H&E) scan, prostate histopathology (H&E) scan, skin dermatology scan, skin dermoscopy scan, skin histopathology (H&E) scan, and a skin multimodal

[1] Example datasets: MIMIC-CXR 231,483 images; EyePACS 199,258; internal dermatology 51,049; PAD-UFES-20 2,047; internal histopathology 32.55M patches; CT-US1 59,979 slices; MRI-US1 47,622 Sellergren et al. (2025).

panel (dermoscopy+RCM+H&E). Rather than storing task-specific models, each candidate is indexed by a string key—the *model-card ID* (a.k.a. model code; e.g., `MODEL_04`, which might refer to an ophthalmology model)—and by two minimal descriptors that make it discoverable by the router: (i) a short *task caption* describing what the model does (e.g., "segments tumor regions in histopathology images"), and (ii) the *modality*, i.e., the scan type on which the model was trained (e.g., "breast histopathology scan").

Stage 1: Modality identification (Figure 1). The goal of the first stage is to determine the imaging modality or scan type of the input image. We prompt MedGemma with a system message that casts it as a medical-imaging specialist and supply a list of admissible modalities drawn from the model-card repository. Concretely, we iterate over all model cards, extract each card's `modality` field, deduplicate the values to form a unique set, and inject that set into the prompt. The prompt instructs: (1) if the image is *not* a medical scan, respond with `None`; (2) if it *is* a medical scan, choose the single most appropriate scan name from the list that matches the image content; and (3) if none of the listed modalities perfectly fits, respond with `Other`. This prompting strategy explicitly conditions the model to use the tokens `None` or `Other` when a confident match is absent. For example, given an endoscopic image of the colon, the correct output might be "colon colonoscopy scan"; for a non-medical photograph (e.g., a picture of a cat), the model should output `None`. Early termination occurs only at this stage: if the selected token is `None` or `Other`, the pipeline halts.

Stage 2: Primary finding detection (Figure 2). In the second stage, the model examines the image (unaware of the router's result for Stage 1) and attempts to identify the primary abnormality or finding, if any. We prompt MedGemma with an instruction such as: "Identify the single most likely disease or finding apparent in this scan. If no disease is present, respond only with: 'Normal.'" This stage is essentially an image-based diagnosis step for a high-level finding. For example, given a colonoscopy image, the model might answer "Polyp" if one is seen; given a chest X-ray, it might answer "Normal" if nothing abnormal stands out, or a broad condition like "Pneumonia" if detected. The prompting ensures that the special token "Normal" is used to indicate the absence of any visible pathology.

Stage 3: Model-card selection (Figure 3). In the final stage, the model selects an appropriate card from the repository. We iterate over all cards, filter to those whose `modality` matches the Stage 1 choice, and pass the resulting list of `(model_card_id, task_caption)` pairs to the prompt. For example, if Stage 1 predicts `breast histopathology scan`, the candidate set might be: [`MODEL_01: Classifies benign vs. malignant findings in breast histopathology slides.`, `MODEL_04: ...`].

To provide sufficient context for disambiguation, we include the Stage 1 modality and the Stage 2 primary abnormality as structured fields in the prompt (e.g., *Patient scan details*: breast histopathology scan; *Primary abnormality*: carcinoma). To make this a pure selection task, we mask the input image so that MedGemma acts as a language model solving a multiple-choice problem over the candidate list. The prompt instructs the model to: (i) review each candidate's task caption; (ii) select a model only if its task explicitly matches *both* the identified modality and abnormality; (iii) reply `None` if no option fits; and (iv) when a match exists, return the model's code (card ID) plus a one-line justification of the fit.

Cutoff Tuning and Answer Selection. Even with careful prompting (e.g., `return the best-matched`, restrictive cues like `only`, and explicit abstentions such as `None`), the router can still pick suboptimal first tokens or miss safe abstentions. Empirically, we observe that the *second-most likely* first token can sometimes yield a more faithful full answer. We therefore arbitrate between the top two candidates using a simple, stage-specific *cutoff* on the runner-up probability.

Selector logic. At each stage, the model forms a distribution over the first token. We take the top two tokens and decode full answers conditioned on each, producing $\{(a_i, p_i)\}_{i=1}^{2}$ with $p_1 \geq p_2$. This pair is passed to a transparent selector that prefers the runner-up only when its probability is sufficiently large. The rule returns both the chosen answer $\hat{a}$ and the index $y \in \{\textsc{first}, \textsc{second}\}$. When applying Algorithm 1 **in Stage 1 only**, if the selected token is literally `None` or `Other`, we halt and return `None` as the router's final output (early termination to avoid extra compute).

Learning the cutoff in a supervised fashion. We gathered 102 diverse (image, per-stage top-2 predictions, label *(First/Second)*) pairs over the training

Algorithm 1: Answer selector (per stage)

```
Given   $\mathcal{C} = \{(a_i, p_i)\}_{i=1}^{|V|}$ with $p_1 \geq p_2$, cutoff $\tau$.
Output  $\hat{a}$ (selected answer), $y \in \{\textsc{first}, \textsc{second}\}$.

1. Set $\hat{a} \leftarrow a_1$ and $y \leftarrow$ FIRST.
2. If $p_2 \geq \tau$ and $a_2 \neq a_1$ then
       set $\hat{a} \leftarrow a_2$ and $y \leftarrow$ SECOND (prefer runner-up when $p_2$ is large).
```

set and 18 pairs over the held-out set. All samples were collected from Wikimedia Commons and share a Creative Commons Attribution–ShareAlike (CC BY-SA) license. For each stage $k \in \{1, 2, 3\}$, the training split provides the correct index (FIRST vs. SECOND). We learn a single *cutoff* τ_k via a one-pass sweep over the sorted runner-up probabilities p_2, which returns *all* optimal intervals and a canonical midpoint for deployment. Applying Algorithm 2 with the p_2-based selector (Alg. 1) yields the optimal intervals and representative cutoffs in Table 1. Intervals are open on the right; wide plateaus indicate robustness.

Algorithm 2: Cutoff sweep (training split, stage k)

```
Given   Training rows for stage $k$ with runner-up
        probabilities $p_2$ and labels $y \in \{\textsc{first}, \textsc{second}\}$.
Output  All optimal cutoff intervals for $\tau_k$; canonical $\tau_k^\star$
        (midpoint of the widest interval).

1. Partition events: $A \leftarrow \{p_2 : y = \textsc{second}\}$,  $B \leftarrow \{p_2 : y = \textsc{first}\}$.
2. Sort once: merge $A \cup B$ ascending, consuming equal values together (event list).
3. Initialize score: place $\tau$ just left of 0; $S_0 \leftarrow |A|$.
4. One-pass sweep (left→right):
   (a) Crossing an $A$-value: a previously correct flip-to-second becomes incorrect $\Rightarrow S \leftarrow S - 1$.
   (b) Crossing a $B$-value: a previously incorrect stay-with-first becomes correct $\Rightarrow S \leftarrow S + 1$.
5. Track maxima: record $S_{\max}$ and every open interval between successive events where $S = S_{\max}$.
6. Return: these intervals are optimal; set $\tau_k^\star$ to the midpoint of the widest optimal interval.
```

Table 1: Optimal cutoff intervals and representative $\tau_k^\star$ learned on the training split.

Stage	Optimal interval(s)	$\tau_k^\star$
Stage 1 — Modality	[0.417100, 0.434750)	**0.425925**
Stage 2 — Disease	[0.436800, 0.437400)	**0.437100**
Stage 3 — Final	[0.001100, 0.001334); [0.001985, 0.003030)	**0.002508**

Per-stage accuracy (training and held-out). We compare a simple baseline (always choose FIRST) with the cutoff router (Alg. 1 using $\tau_k^\star$ from Alg. 2). The p_2-based selector is transparent and stage-local. Cutoffs learned by a single sweep improve final-stage correctness on both splits (approximately +10 pp on training split; approximately +6 pp on held-out split), with modest gains at earlier stages. The inference-time Stage 1 abstention guard reduces unnecessary computation without affecting cutoff learning or the evaluations above.

Table 2: Per-stage accuracy with and without cutoff.

Split	Stage / Task	Baseline acc	Cutoff acc (ours)
Training	Stage 1 — Modality	0.9216	**0.9412**
	Stage 2 — Disease	0.7647	**0.7745**
	Stage 3 — Final	0.4608	**0.5588**
Held-out	Stage 1 — Modality	0.8889	**0.8889**
	Stage 2 — Disease	0.6667	**0.6667**
	Stage 3 — Final	0.5556	**0.6111**

3.1. Router Results: Cutoff vs. Baseline

Setup. We compare the router's *final decision stage* with and without a cutoff on the tuning splits (**training**: 102, **held-out**: 18). For each image, the router emits a *predicted label* (either a `MODEL_#` or `None`) together with a *predicted confidence* in $[0, 1]$, i.e., the probability the router assigns to the emitted label at decision time. Accuracy is measured by exact match between the *predicted label* and the *ground-truth label*. Calibration is summarized with the **Expected Calibration Error (ECE)** using 10 equal-width, right-closed bins:

$$\mathrm{ECE} \;=\; \sum_{b=1}^{10} \frac{n_b}{N} \left|\mathrm{acc}_b - \mathrm{conf}_b\right| \quad \text{(lower is better).}$$

Table 3: Final decision stage results. Accuracy is exact match; ECE uses 10 equal-width, right-closed bins.

Split / Method	Accuracy	ECE
Training — Baseline (no cutoff)	0.392	0.415
Training — Cutoff (ours)	**0.480**	0.490
Held-out — Baseline (no cutoff)	0.444	0.390
Held-out — Cutoff (ours)	**0.556**	**0.372**

Reliability curves. Figures 4 and 5 plot *accuracy vs. confidence* per bin for the cutoff router and the baseline, along with the ideal $y{=}x$ diagonal. On the held-out split, the cutoff curve lies closer to the diagonal (lower ECE). On the training split, the cutoff router improves accuracy but is mildly under-calibrated at mid/high confidence, increasing ECE. Overall, Cutoff improves end-task correctness on both splits (+8.8 pp on training split;

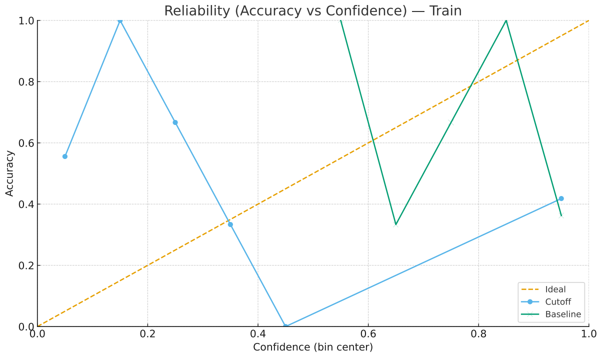


Figure 4: Reliability (Accuracy vs. Confidence) on **training split**. Dashed line: ideal $y=x$. The cutoff router improves accuracy but is slightly under-calibrated in mid/high-confidence bins.

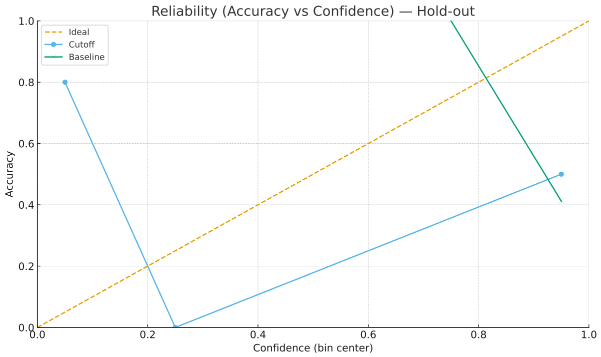


Figure 5: Reliability (Accuracy vs. Confidence) on **held-out**. The cutoff router improves both accuracy and calibration (lower ECE) compared to the baseline.

+11.1 pp on held-out split) and achieves better held-out calibration (ECE 0.372 vs. 0.390). A practical advantage, visible in Figures 4 and 5, is that when confidence exceeds 0.5, accuracy under Cutoff tends to increase monotonically with confidence; in contrast, the baseline exhibits mixed behavior, where confidence does not reliably track accuracy. Furthermore, on the held-out split the cutoff curve closely mirrors the shape observed on the training split—aside from a slight deviation in the first (lowest-confidence) range—indicating that the reliability profile measured on the training data largely transfers to unseen data. In practice, this suggests that development-time reliability estimates are predictive of deployment-time reliability.

4. Solution 2: Specialty-Specific Model for Downstream Deployment

Building on the rationale in Section 1 (Solution 2), we fine-tune the same backbone VLM adopted in Section 3—MedGemma—across five specialties (gastroenterology, hematology, ophthalmology, pathology, and radiology) to simplify maintenance and deployment rather than pursuing state-of-the-art performance on each task. Importantly, Solution 2 is implemented and evaluated independently; it does not consume outputs from Solution 1, though the two are conceptually complementary.

For each specialty we compare: (a) an external task-specific benchmark (typically the published state-of-the-art for that dataset), (b) the base MedGemma in zero-shot mode (no domain adaptation), and (c) MedGemma after specialty-specific fine-tuning (denoted "S–S" in the table). This tests whether a single specialty-tuned MedGemma can substitute for the external benchmark while retaining a single deployment stack. If successful, a fixed VLM plus targeted prompt engineering and supervised fine-tuning may reduce the need to design, validate, and operate separate architectures—saving compute and shortening triage-to-deployment cycles Annarumma et al. (2019). All results are summarized in Table 4.

Datasets For gastroenterology, we use the dataset introduced by Li et al. (2021), an annotated compilation of three public sources: CVC-ColonDB Bernal et al. (2012), GLRC Mesejo et al. (2016), and KUMC (80 colonoscopy video sequences from the University of Kansas Medical Center). For hematology, we use BloodMNIST Acevedo et al. (2020), organized into eight blood cell categories. For ophthalmology, we employ OCTMNIST Kermany et al. (2018) (four retinal OCT classes) and RetinaMNIST Dataset (2020) (five retinopathy grades). For pathology, we use TissueMNIST Ljosa et al. (2012) (eight kidney cortex cell types), PathMNIST Kather et al. (2019) (nine colorectal tissue types), and a breast cancer histopathology set following Cruz-Roa et al. (2014). Lastly, for radiology, we use BreastMNIST Al-Dhabyani et al. (2020) (binary classification of breast images: normal vs. malignant), DeepLesion Yan et al. (2017) (eight different body organs), Organ(A-B-C)MNIST Bilic et al. (2023); Xu et al. (2019) (11 body organs in axial, coronal, and sagittal views, respectively), PneumoniaMNIST Kermany (2018); Kermany et al. (2018) (pneumonia vs. normal chest images), and ChestMNIST Wang et al. (2017) (14 disease labels in X-ray images). Train/validation/test sizes are summarized in Table 4.

Fine-tuning results For each *specialty-specific* case (five in total), we use a parameter-efficient fine-tuning (PEFT) method, QLoRA Dettmers et al. (2023). As shown in Table 4, specialty-specific fine-tuning substantially narrows the gap to specialist benchmarks. While the base MedGemma performs modestly in zero-shot mode, targeted specialty adaptation yields competitive results across modalities and objectives. This supports the premise that one VLM can act as a generalist engine per specialty, reducing the need for separate per-task architectures and thereby saving compute and shortening triage-to-deployment cycles. We also provide public example inference results for each specialty-specific fine-tuning case here.

5. Discussion

We presented two linked clinical solutions: (1) a single, calibrated medical VLM that runs an aware model-card–matching workflow, routing an input image to a model-card ID—or abstaining and terminating when appropriate—and (2) the same VLM used as a specialty-specific downstream model with performance competitive with task-specific models. The solutions are complementary but *decoupled* in our implementation: Solution 2 neither requires nor consumes Solution 1 outputs. For **Solution 1**, the workflow is designed to streamline selection for data scientists by providing: *safety and trust* (stage-wise decisions are visible throughout the pipeline), *robustness* (fewer incorrect selections via the combined use of stage-wise prompts and answer-selector logic), *scalability* (flexibility to add stages, descriptors, or model cards, or to swap in a more mature selector), and *speed* (a minimalist design with fewer moving parts than recent multi-agent pipelines (Shimgekar et al., 2025a,b)). For **Solution 2**, specialty-specific models offer a practical first step toward automated deployment. We make this step concrete by shifting effort from designing new architectures to engineering task-specific prompts that align the VLM's outputs with downstream objectives. In practice, this process can increase throughput for data scientists and clinicians without sacrificing performance.

Limitations and future work. MedGemma is pretrained on a fixed set of modalities and tasks; accordingly, it may not generalize to imaging types or findings outside its training distribution. In short, its inference and selection decisions are tied to its training data: if a modality or pathology is unseen or underrepresented, the model may fail to recognize it and either choose an imperfect model or default to "Other/None." This risk can be mitigated by enlarging the pretraining corpus or by adopting continual-learning methods that add new domains while preserving prior knowledge (e.g., Li and Hoiem (2017)).

While our system outputs justifications for model selection, the interpretability of the VLM's internal reasoning remains limited (a common issue with large neural models). Techniques such as chain-of-thought prompting or explicability constraints could be applied so the model provides brief, stage-wise reasoning—further improving transparency (see Wei et al. (2022)). Another design choice concerns the selection of thresholds τ_1, τ_2, τ_3. We derive the optimal cutoff thresholds using Algorithm 2, which maximizes the per-stage score given (i) our stage-specific prompting strategy; (ii) the diversity and structure of the model-card repository defined in Section 3; and (iii) the workflow's preference for the true final output—either `None` or a model-card ID—across different input images. In practice, these thresholds may require adjustment across clinical contexts. For high-accuracy applications using the same modality and disease labels as the training set, a workflow designer might prefer `None` over the closest model-card match, resulting in different labels for Stage 3 tuning.

Finally, our pipeline has three stages because repository lookup is primarily based on modality and a single primary finding. As we incorporate more complex model cards (e.g., some models might handle combinations of findings or require patient metadata), the prompt design may need to extend to additional stages or a more free-form query. Fortunately, the answer-selector concept generalizes to any number of stages by examining the confidence of alternative responses. We outline two complementary directions for deployment: (1) automate fine-tuning end to end—so deployment proceeds with minimal data-scientist intervention—by automatically selecting the dataset linked to the model card chosen in Solution 1 and automatically generating the prompts required for fine-tuning; and (2) reduce inference latency by applying pruning methods tailored to large language models (e.g., SparseGPT Frantar and Alistarh (2023)).

Table 4: Performance summary across specialties

Task	**External benchmark**	**MedGemma**	**(S–S) fine-tuned**
Gastroenterology — *PolypSet* (Training: 27,048; Test: 4,719; Val: 4,214)			
Polyp Classification (Acc.)	**81**%	61%	79%
Polyp Classification (F1 Macro)	**80**%	38%	77%
Polyp Classification (F1 Weighted)	**81**%	47%	79%
BBox Localization Acc.	**78**%	1%	—
Region Localization Acc.	—	32%	**46**%
Hematology — *BloodMNIST* (Training: 11,959; Test: 3,421; Val: 1,712)			
Cell-type Classification (Acc.)	97%	20%	**98**%
Cell-type Classification (F1 Macro)	—	8%	**98**%
Cell-type Classification (F1 Weighted)	—	10%	**98**%
Ophthalmology — *OCTMNIST* (Training: 97,477; Test: 1,000; Val: 10,832)			
Retinal OCT Classification (Acc.)	78%	26%	**93**%
Retinal OCT Classification (F1 Macro)	—	38%	**77**%
Retinal OCT Classification (F1 Weighted)	—	47%	**79**%
Ophthalmology — *RetinaMNIST* (Training: 1,080; Test: 400; Val: 120)			
Diabetic Retinopathy Grading (Acc.)	53%	55%	**74**%
Diabetic Retinopathy Grading (F1 Macro)	—	37%	**59**%
Diabetic Retinopathy Grading (F1 Weighted)	—	52%	**72**%
Pathology — *TissueMNIST* (Training: 165,466; Test: 47,280; Val: 23,640)			
Kidney Cortex Classification (Acc.)	**70.3**%	21%	70.1%
Kidney Cortex Classification (F1 Macro)	—	5%	**60**%
Kidney Cortex Classification (F1 Weighted)	—	9%	**69**%
Pathology — *PathMNIST* (Training: 89,996; Test: 7,180; Val: 10,004)			
Tissue Type Classification (Acc.)	91.1%	38%	**96**%
Tissue Type Classification (F1 Macro)	—	28%	**94**%
Tissue Type Classification (F1 Weighted)	—	29%	**96**%
Pathology — *BreastHIST* (Training: 214,969; Test: 31,376; Val: 31,179)			
Ductal Carcinoma Classification (Acc.)	84.3%	69%	**87**%
Ductal Carcinoma Classification (F1 Macro)	—	66%	**84**%
Ductal Carcinoma Classification (F1 Weighted)	—	70%	**87**%
Radiology — *BreastMNIST* (Training: 546; Test: 156; Val: 78)			
Breast Lesion Classification (Acc.)	**86.3**%	73%	74%
Breast Lesion Classification (F1 Macro)	—	42%	**52**%
Breast Lesion Classification (F1 Weighted)	—	62%	**67**%
Radiology — *DeepLesion* (Training: 7,859; Test: 984; Val: 973)			
BBox-Guided Region Classification (Acc.)	—	28%	**90**%
BBox-Guided Region Classification (F1 Macro)	—	19%	**87**%
BBox Guided Region Classification (F1 Weighted)	—	23%	**90**%
Radiology — *OrganAMNIST* (Training: 34,859; Test: 17,778; Val: 6,491)			
Organ Classification (Axial) (Acc.)	**95**%	9%	94%
Organ Classification (Axial) (F1 Macro)	—	4%	**92**%
Organ Classification (Axial) (F1 Weighted)	—	6%	**94**%
Radiology — *OrganCMNIST* (Training: 12,975; Test: 8,216; Val: 2,392)			
Organ Classification (Coronal) (Acc.)	**92**%	10%	88%
Organ Classification (Coronal) (F1 Macro)	—	7%	**86**%
Organ Classification (Coronal) (F1 Weighted)	—	9%	**88**%
Radiology — *OrganSMNIST* (Training: 13,932; Test: 8,827; Val: 2,452)			
Organ Classification (sagittal) (Acc.)	**81**%	7%	75%
Organ Classification (sagittal) (F1 Macro)	—	3%	**93**%
Organ Classification (sagittal) (F1 Weighted)	—	3%	**95**%
Radiology — *PneumoniaMNIST* (Training: 4,708; Test: 624; Val: 524)			
Chest Pneumonia Classification (Acc.)	95%	28%	**95**%
Chest Pneumonia Classification (F1 Macro)	—	73%	**94**%
Chest Pneumonia Classification (F1 Weighted)	—	74%	**94**%
Radiology — *ChestMNIST* (Training: 78,468; Test: 22,433; Val: 11,219)			
Chest Multi-label Classification (Acc.)	**95**%	86%	92%
Chest Multi-label Classification (F1 Macro)	—	**16**%	14%
Chest Multi-label Classification (F1 Weighted)	—	40%	**40**%

References

Andrea Acevedo, Anna Merino, Santiago Alférez, Ángel Molina, Laura Boldú, and José Rodellar. A dataset of microscopic peripheral blood cell images for development of automatic recognition systems. *Data in brief*, 30:105474, 2020.

Aidoc. Intelligent orchestration for radiology ai. https://www.aidoc.com/ai-orchestration/, 2024. Accessed 2025-08-10.

Walid Al-Dhabyani, Mohammed Gomaa, Hussien Khaled, and Aly Fahmy. Dataset of breast ultrasound images. *Data in brief*, 28:104863, 2020.

Mauro Annarumma, Steve Withey, Richard Bakewell, Emanuele Pesce, Via Goh, and Giovanni Montana. Automated triaging of adult chest radiographs with deep artificial neural networks. *The Lancet Digital Health*, 1(7):e307–e316, 2019. doi: 10.1016/S2589-7500(19)30117-0. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6438359/.

Jorge Bernal, Javier Sánchez, and Fernando Vilarino. Towards automatic polyp detection with a polyp appearance model. *Pattern Recognition*, 45 (9):3166–3182, 2012.

Patrick Bilic, Patrick Christ, Hongwei Bran Li, Eugene Vorontsov, Avi Ben-Cohen, Georgios Kaissis, Adi Szeskin, Colin Jacobs, Gabriel Efrain Humpire Mamani, Gabriel Chartrand, et al. The liver tumor segmentation benchmark (lits). *Medical image analysis*, 84:102680, 2023.

Bart-Jan Boverhof, W Ken Redekop, Daniel Bos, Martijn PA Starmans, Judy Birch, Andrea Rockall, and Jacob J Visser. Radiology ai deployment and assessment rubric (radar) to bring value-based ai into radiological practice. *Insights into Imaging*, 15(1):34, 2024.

Adrian P Brady, Bibb Allen, Jaron Chong, Elmar Kotter, Nina Kottler, John Mongan, Lauren Oakden-Rayner, Daniel Pinto Dos Santos, An Tang, Christoph Wald, et al. Developing, purchasing, implementing and monitoring ai tools in radiology: practical considerations. a multi-society statement from the acr, car, esr, ranzcr & rsna. *Canadian Association of Radiologists Journal*, 75 (2):226–244, 2024.

buoy health. buoy health solutions. https://www.buoyhealth.com, 2025. Accessed: 2025-08-12.

Angel Cruz-Roa, Ajay Basavanhally, Fabio González, Hannah Gilmore, Michael Feldman, Shridar Ganesan, Natalie Shih, John Tomaszewski, and Anant Madabhushi. Automatic detection of invasive ductal carcinoma in whole slide images with convolutional neural networks. In *Medical imaging 2014: Digital pathology*, volume 9041, page 904103. SPIE, 2014.

DC Dataset. The 2nd diabetic retinopathy–grading and image quality estimation challenge, 2020.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms, 2023. *URL https://arxiv.org/abs/2305.14314*, 2, 2023.

Elias Frantar and Dan Alistarh. Sparsegpt: Massive language models can be accurately pruned in one-shot. In *International conference on machine learning*, pages 10323–10337. PMLR, 2023.

Garner. Garner solutions. https://www.getgarner.com, 2025. Accessed: 2025-08-12.

Yonatan Geifman and Ran El-Yaniv. Selectivenet: A deep neural network with an integrated reject option. In *ICML*, pages 2151–2159, 2019.

Fabian Isensee, Paul F. Jäger, Simon A. A. Kohl, Jens Petersen, and Klaus H. Maier-Hein. nnu-net: a self-configuring method for deep learning-based biomedical image segmentation. *Nature Methods*, 18:203–211, 2021. doi: 10.1038/s41592-020-01008-z.

Jakob Nikolas Kather, Johannes Krisam, Pornpimol Charoentong, Tom Luedde, Esther Herpel, Cleo-Aron Weis, Timo Gaiser, Alexander Marx, Nektarios A Valous, Dyke Ferber, et al. Predicting survival from colorectal cancer histology slides using deep learning: A retrospective multicenter study. *PLoS medicine*, 16(1):e1002730, 2019.

Daniel Kermany. Large dataset of labeled optical coherence tomography (oct) and chest x-ray images. *(No Title)*, 2018.

Daniel S Kermany, Michael Goldbaum, Wenjia Cai, Carolina CS Valentim, Huiying Liang, Sally L Baxter, Alex McKeown, Ge Yang, Xiaokang Wu, Fangbing Yan, et al. Identifying medical diagnoses and treatable diseases by image-based deep learning. *cell*, 172(5):1122–1131, 2018.

Kaidong Li, Mohammad I Fathan, Krushi Patel, Tianxiao Zhang, Cuncong Zhong, Ajay Bansal, Amit Rastogi, Jean S Wang, and Guanghui Wang. Colonoscopy polyp detection and classification: Dataset creation and comparative evaluations. *Plos one*, 16(8):e0255809, 2021.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

Vebjorn Ljosa, Katherine L Sokolnicki, and Anne E Carpenter. Annotated high-throughput microscopy image sets for validation. *Nature methods*, 9(7):637, 2012.

Mage AI. How long does it take to build an ml model? https://m.mage.ai/how-long-does-it-take-to-build-an-ml-model-d68b8afa50a5, 2025. Accessed: 2025-08-12.

Xueyan Mei, Philip M. Robson, Yang Yang, Zelong Liu, et al. Radimagenet: An open radiologic deep learning research dataset for effective transfer learning. *Radiology: Artificial Intelligence*, 4(5): e210315, 2022. doi: 10.1148/ryai.210315. URL https://pubmed.ncbi.nlm.nih.gov/36204533/.

Pablo Mesejo, Daniel Pizarro, Armand Abergel, Olivier Rouquette, Sylvain Beorchia, Laurent Poincloux, and Adrien Bartoli. Computer-aided classification of gastrointestinal lesions in regular colonoscopy. *IEEE transactions on medical imaging*, 35(9):2051–2063, 2016.

Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. Model cards for model reporting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT* 2019)*, pages 220–229. ACM, 2019. doi: 10.1145/3287560.3287596.

Mukund Poddar, Jayson S. Marwaha, William Yuan, Santiago Romero-Brufau, and Gabriel A. Brat. An operational guide to translational clinical machine learning in academic medical centers. *npj Digital Medicine*, 7(1):129, 2024. doi: 10.1038/s41746-024-01094-9.

Project MONAI. Monai deploy app sdk. https://github.com/Project-MONAI/monai-deploy-app-sdk, 2023. Accessed 2025-08-10.

D. Sculley, G. Holt, D. Golovin, E. Davydov, T. Phillips, D. Ebner, V. Chaudhary, M. Young, J.-F. Crespo, and D. Dennison. Hidden technical debt in machine learning systems. In *NeurIPS*, 2015.

Andrew Sellergren, Sahar Kazemzadeh, Tiam Jaroensri, Atilla Kiraly, Madeleine Traverse, Timo Kohlberger, Shawn Xu, Fayaz Jamil, Cían Hughes, Charles Lau, et al. Medgemma technical report. *arXiv preprint arXiv:2507.05201*, 2025. URL https://arxiv.org/abs/2507.05201.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face. *Advances in Neural Information Processing Systems*, 36:38154–38180, 2023.

Soorya Ram Shimgekar, Abhay Goyal, Shayan Vassef, Koustuv Saha, Christian Poellabauer, Xavier Vautier, Pi Zonooz, and Navin Kumar. Nimblelabs: Accelerating healthcare ai development through agentic ai. *Preprints*, August 2025a. doi: 10.20944/preprints202508.1713.v1. URL https://doi.org/10.20944/preprints202508.1713.v1.

Soorya Ram Shimgekar, Shayan Vassef, Abhay Goyal, Navin Kumar, and Koustuv Saha. Agentic ai framework for end-to-end medical data inference. *arXiv preprint arXiv:2507.18115*, 2025b. URL https://arxiv.org/abs/2507.18115.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Xiaosong Wang, Yifan Peng, Le Lu, Zhiyong Lu, Mohammadhadi Bagheri, and Ronald M Summers. Chestx-ray8: Hospital-scale chest x-ray database and benchmarks on weakly-supervised classification and localization of common thorax diseases. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2097–2106, 2017.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed H. Chi, Quoc V. Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*, 2022. URL https://arxiv.org/abs/2201.11903.

Martin J. Willemink, Wojciech A. Koszek, Cailin Hardell, Jie Wu, Dominik Fleischmann, Hugh Harvey, Les R. Folio, Ronald M. Summers, Daniel L. Rubin, and Matthew P. Lungren. Preparing medical imaging data for machine learning. *Radiology*, 295(1):4–15, 2020. doi: 10.1148/radiol.2020192224.

Hanwen Xu, Naoto Usuyama, Jaspreet Bagga, Sheng Zhang, Rajesh Rao, Tristan Naumann, Cliff Wong, Zelalem Gero, Javier González, Yu Gu, et al. A whole-slide foundation model for digital pathology from real-world data. *Nature*, 630(8015):181–188, 2024.

Xuanang Xu, Fugen Zhou, Bo Liu, Dongshan Fu, and Xiangzhi Bai. Efficient multiple organ localization in ct image using 3d region proposal network. *IEEE transactions on medical imaging*, 38(8):1885–1898, 2019.

Ke Yan, Xiaosong Wang, Le Lu, and Ronald M Summers. Deeplesion: Automated deep mining, categorization and detection of significant radiology image findings using large-scale clinical lesion annotations. *arXiv preprint arXiv:1710.01766*, 2017.

Haiyang Yu, Siyang Yi, Ke Niu, Minghan Zhuo, and Bin Li. Umit: Unifying medical imaging tasks via vision–language models. *arXiv preprint arXiv:2503.15892*, 2025. URL https://arxiv.org/abs/2503.15892.

Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D. Manning, and Curtis P. Langlotz. Contrastive learning of medical visual representations from paired images and text. In *Proceedings of the Machine Learning for Healthcare Conference*, volume 182 of *Proceedings of Machine Learning Research*, pages 1–24, 2022. URL https://proceedings.mlr.press/v182/zhang22a/zhang22a.pdf.