Retrieval-Augmented Interpretable Learning: Towards Task-Specific Zero-Shot Models in Healthcare

Sazan Mahbub[c] **Caleb Ellington**[c,g] **Zhiyuan Li**[w] **Yixin Yang**[w]
Souvik Kundu[i] **Ben Lengerich**[w] **Eric P. Xing**[c,m,g]

[c]Carnegie Mellon University [w]University of Wisconsin–Madison
[m]Mohamed bin Zayed University of AI [g]GenBio AI [i]Intel

Abstract

We introduce **Retrieval-Augmented Interpretable Learning (RAIL)**, a probabilistic meta-learning framework for zero-shot generation of task-specific interpretable models that synthesizes coefficient-space structure from natural-language task descriptions and a memory of previously learned task-specific predictors. RAIL retrieves related source tasks, transfers structure through coefficient space, and generates a new predictor in the original diagnostic-feature space, enabling zero-shot and few-shot clinical procedure prediction with feature-level explanations. Its probabilistic formulation provides uncertainty over retrieval, model coefficients, and predictions, supporting reliability-aware deployment: uncertain predictions or unstable explanations can be flagged for additional clinical review rather than treated as automatic decisions. This makes RAIL particularly suited for healthcare settings, where prediction tasks are highly long-tailed, new clinical targets arise frequently, and models must remain inspectable, uncertainty-aware, and compatible with human oversight. Across long-tailed clinical procedure prediction tasks, RAIL maintains reliable performance across data-availability regimes: it achieves 73.4% accuracy in the *held-out zero-shot* settings, where no supervised task-specific model can be trained, and remains near 73.2% accuracy in the extreme few-shot regime with only 2–4 examples, where supervised task-specific models perform close to chance. RAIL further benefits from clinically informed task representations and yields retrieval, uncertainty, and coefficient-level diagnostics that make model behavior more transparent. These results suggest a path toward scalable clinical prediction systems that can adapt to new tasks while preserving interpretability and reliability.

1 Introduction

Recent advances in clinical predictive modeling have highlighted the need for models that are not only accurate, but also adaptive, interpretable, and reliable under limited supervision. In many healthcare settings, one needs task-specific predictors for deciding whether different procedures, medications, or interventions should be administered to a patient, conditioned on a shared set of diagnostic measurements. However, clinical task distributions are naturally long-tailed: common procedures may have abundant labels, while many clinically meaningful or newly considered tasks have only a few examples, and some have no task-specific labels at all. This makes independently training a supervised model for every task unreliable in precisely the regimes where adaptable clinical prediction is most needed.

A growing body of work has studied meta-models and task-conditioned predictors that dynamically generate or adapt models for specific contexts or tasks [1, 2, 3, 4]. In parallel, pretrained language models have been explored as sources of prior knowledge for downstream decision-making [5],

Correspondence to: smahbub@cs.cmu.edu, lengerich@wisc.edu, epxing@cs.cmu.edu.

Preprint.

including reinforcement learning [6, 7, 8], feature selection [9], causal discovery [10, 11], and healthcare retrieval-augmented prediction pipelines [12]. These approaches suggest that task descriptions and pretrained representations can provide useful priors for adaptation. Yet, many language-model-based or black-box transfer systems do not directly produce transparent, task-specific predictive models. Their outputs often remain embedded in latent representations or natural-language rationales, limiting their use when clinicians need inspectable feature-level weights, uncertainty estimates, and signals for when additional human review is warranted.

This motivates our central question: *Can we generate an interpretable task-specific clinical prediction model in a zero-shot manner, using only a natural-language task description and a memory of previously learned interpretable predictors?* Such a framework would combine the adaptability of meta-learning and retrieval-augmented modeling with the transparency of classical interpretable predictors. It would also support clinical oversight by exposing which features drive a prediction, when the generated model is uncertain, and when predictions should be flagged for additional review.

We introduce **Retrieval-Augmented Interpretable Learning (RAIL)**, a probabilistic meta-learning framework for *zero-shot generation* of task-specific interpretable models, synthesizing coefficient-space structure from natural-language task descriptions and a memory of previously learned linear predictors. RAIL represents each prior task using both a language embedding of its task description and the coefficients of an interpretable model trained for that task. Given a new target task, RAIL retrieves related source tasks, transfers structure through coefficient space, and generates a new predictor in the original diagnostic-feature space. The resulting model is directly inspectable: each coefficient corresponds to an input feature, enabling feature-level interpretation rather than post-hoc explanation of an opaque predictor.

The key idea is that task descriptions and task-specific coefficients encode complementary forms of transferable knowledge. Language embeddings provide semantic alignment across procedures and clinical targets, while learned coefficients encode how diagnostic features were used by prior interpretable predictors. RAIL combines these signals through a probabilistic retrieval formulation: retrieved tasks define a retrieval-conditioned prior over target-task coefficients, which directly yields a generated predictor in the zero-shot regime and can be further adapted when limited target-task labels are available. Because RAIL is probabilistic, it also produces uncertainty over retrieval, coefficients, and predictions. These uncertainty signals can identify ambiguous retrieval neighborhoods, unstable feature-level explanations, and individual predictions that may require clinician review.

We evaluate RAIL on long-tailed clinical procedure prediction tasks, where each task asks whether a procedure should be administered to a patient given diagnostic results. Our experiments show that RAIL is especially useful in low-data and zero-shot regimes, where task-specific supervision is scarce or unavailable. Quantitatively, RAIL remains stable as supervision decreases: it obtains 73.4% accuracy on held-out zero-shot tasks using only task descriptions and retrieved prior predictors, and achieves 73.2% accuracy in the extreme few-shot regime with only 2–4 labeled examples. In contrast, supervised task-specific baselines degrade substantially in this regime, highlighting the value of transferring coefficient-space structure from related source tasks. Controlled baselines and ablations show that performance depends on clinically informed task representations, meaningful retrieval, and learned coefficient synthesis. Retrieval diagnostics further show that RAIL retrieves coherent and oracle-consistent task neighborhoods, while embedding perturbation studies confirm that task-representation quality is central to model generation. Finally, uncertainty analyses show that predictive confidence tracks empirical accuracy, uncertainty identifies likely failures, selective prediction reduces risk, and coefficient-level uncertainty helps distinguish stable from unstable explanations. Our contributions are as follows:

- We introduce **RAIL**, a probabilistic meta-learning framework for *zero-shot generation* of task-specific interpretable clinical prediction models from natural-language task descriptions and a memory of prior interpretable predictors.
- We formulate model generation as retrieval-conditioned coefficient-space transfer, producing predictors in the original diagnostic-feature space and preserving direct feature-level interpretability.
- We develop a probabilistic formulation that supports zero-shot generation, optional few-shot adaptation, and uncertainty estimation over retrieval, coefficients, and predictions.
- We evaluate RAIL on long-tailed clinical procedure prediction tasks, with baselines, retrieval ablations, text-encoder comparisons, oracle retrieval diagnostics, and embedding perturbation studies showing when and why retrieval-augmented generation succeeds.

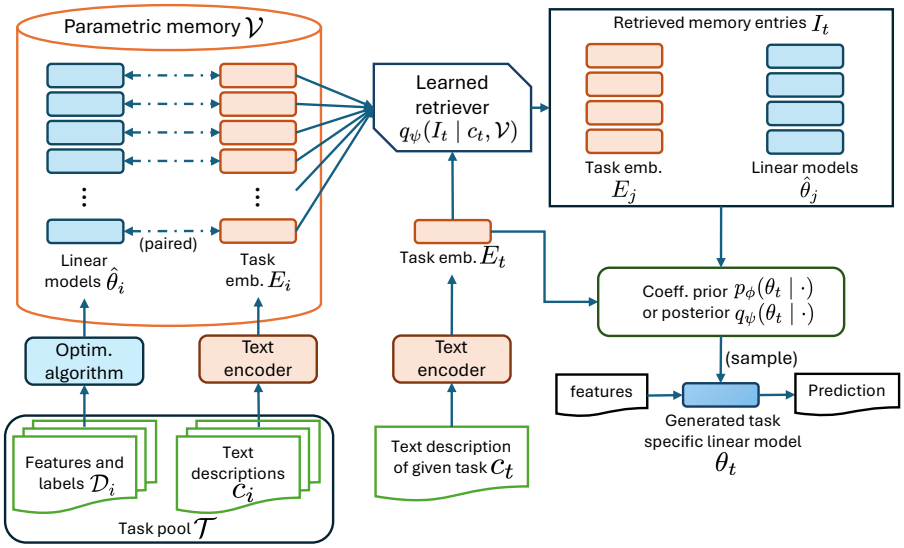


Figure 1: Overview of RAIL. Retrieved task embeddings and learned linear models from the parametric memory condition the generation of a task-specific interpretable predictor.

- We demonstrate that RAIL provides reliability and interpretability diagnostics, including uncertainty-based failure detection, selective prediction, and coefficient-level uncertainty for identifying stable versus unstable explanations.

2 RAIL: Retrieval-Augmented Interpretable Learning

Goal. Given a new task t described only by natural language c_t, our goal is to synthesize an interpretable task-specific predictor $f_t(x) = \sigma(\theta_t^\top x)$ without training a new black-box model from scratch. RAIL assumes access to a memory of prior tasks, where each task is represented by both a language embedding and the coefficients of an interpretable linear model. The embeddings provide semantic alignment across tasks, while the coefficients provide transferable parameter-level structure in the original feature space. RAIL is primarily designed for zero-shot model synthesis: at inference time, no labeled examples from task t are required. When limited target-task labels are available, the same formulation supports few-shot posterior adaptation by combining the retrieval-conditioned prior with a task-specific estimator.

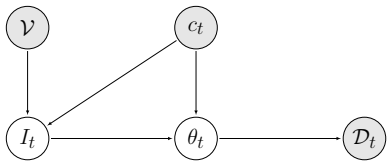


Figure 2: PGM for RAIL. Observed variables are shaded.

2.1 Parametric memory: semantic keys and interpretable values

We assume training tasks $\{\mathcal{T}_i = (\mathcal{D}_i, c_i)\}_{i=1}^n$, where $\mathcal{D}_i = \{(x_{ij}, y_{ij})\}_{j=1}^{m_i}$ share a common feature space $\mathcal{X} \subset \mathbb{R}^f$, and each task has a natural-language description c_i. For each task i, we train an interpretable linear model, e.g., logistic regression, to obtain coefficients $\hat{\theta}_i \in \mathbb{R}^f$. We also compute a task embedding $E_i = \mathrm{LM}(c_i) \in \mathbb{R}^e$ using a frozen language-model encoder. The resulting parametric memory is

$$\mathcal{V} = \{(\hat{\theta}_i, E_i)\}_{i=1}^n, \tag{1}$$

where embeddings serve as semantic retrieval keys and coefficients serve as interpretable parameter-space values.

2.2 Latent-variable formulation

RAIL introduces two latent quantities. The retrieval latent $I_t \subseteq \mathcal{V}$ is a retrieved subset of memory entries selected from the full parametric memory, with $|I_t| = S$. Each element of I_t is a source-task memory entry $(\hat{\theta}_j, E_j)$. The task-parameter latent $\theta_t \in \mathbb{R}^f$ is the coefficient vector of the interpretable predictor for task t.

2.3 Generative model

The parametric memory $\mathcal{V}$ and task description c_t are observed conditioning variables. RAIL models uncertainty over both the retrieved memory subset and the task-specific coefficients:

$$p_\phi(\mathcal{D}_t, \theta_t, I_t \mid c_t, \mathcal{V}) = p(\mathcal{D}_t \mid \theta_t)\; p_\phi(\theta_t \mid I_t, c_t)\; p(I_t \mid c_t, \mathcal{V}). \tag{2}$$

The overview and the probabilistic graphical model are shown in Fig. 1 and Fig. 2, respectively.

Likelihood. For binary prediction, we use the logistic GLM likelihood

$$p(y \mid x, \theta) = \mathrm{Bernoulli}\big(\sigma(\theta^\top x)\big), \tag{3}$$

where $(x, y) \in \mathcal{D}_t$. For regression, the likelihood can analogously be written as $p(y \mid x, \theta) = \mathcal{N}(y; \theta^\top x, \sigma^2)$.

Retrieval prior. The prior $p(I_t \mid c_t, \mathcal{V})$ is induced by cosine similarity between the target task embedding $E_t = \mathrm{LM}(c_t)$ and memory entries. Let $s_{t,j}^{\mathrm{prior}}$ denote the cosine-similarity score for memory entry j. In implementation, we retrieve a candidate subset of size S using these scores and define normalized prior weights within this candidate set:

$$\pi_{t,j}^{\mathrm{prior}} = \frac{\exp(s_{t,j}^{\mathrm{prior}}/\tau_p)}{\sum_{\ell=1}^{S} \exp(s_{t,\ell}^{\mathrm{prior}}/\tau_p)}. \tag{4}$$

This cosine-based prior provides a semantic anchor for the learned retriever.

Retrieval-conditioned coefficient prior. For each retrieved memory entry $(\hat{\theta}_j, E_j) \in I_t$, define

$$K_j = (\hat{\theta}_j \,\|\, E_j), \qquad V_j = \hat{\theta}_j. \tag{5}$$

Conditioned on I_t and c_t, a multi-head cross-attention generator aggregates the retrieved coefficient values and outputs a Gaussian prior over the target-task coefficients:

$$p_\phi(\theta_t \mid I_t, c_t) = \mathcal{N}\big(\theta_t; \mu_{\phi,t}, \mathrm{diag}(\sigma_{\phi,t}^2)\big), \tag{6}$$

where $(\mu_{\phi,t}, \sigma_{\phi,t}^2)$ are produced by neural heads applied to the attention output [13]. We use a diagonal covariance and parameterize $\log \sigma_{\phi,t}^2$.

2.4 Learned retrieval and coefficient inference

We use the factorized variational family

$$q_\psi(I_t, \theta_t \mid \mathcal{D}_t, c_t, \mathcal{V}) = q_\psi(I_t \mid c_t, \mathcal{V})\; q_\psi(\theta_t \mid \mathcal{D}_t, I_t, c_t). \tag{7}$$

The retrieval factor $q_\psi(I_t \mid c_t, \mathcal{V})$ is restricted to be label-free, so that the learned retriever can be used in zero-shot inference. It is trained jointly with the coefficient generator to improve over the cosine-similarity prior, while remaining semantically anchored to it through the retrieval KL term in Eq. (15). The coefficient factor $q_\psi(\theta_t \mid \mathcal{D}_t, I_t, c_t)$ is the variational posterior used when target-task labels are available.

Amortized learned retriever. For each candidate memory entry, we compute a learned retrieval score from the target embedding and candidate key:

$$s_{t,j}^{\mathrm{post}} = g_\psi(E_t, K_j). \tag{8}$$

Let $\mathcal{K}_t$ denote the top-k candidates under these scores, with $k = |\mathcal{K}_t| < S$. To approximate near-discrete retrieval while preserving differentiability, we sharpen only positive top-k scores:

$$\tilde{s}_{t,j}^{\mathrm{post}} = \begin{cases} s_{t,j}^{\mathrm{post}}/\tau_q, & j \in \mathcal{K}_t \text{ and } s_{t,j}^{\mathrm{post}} > 0, \\ s_{t,j}^{\mathrm{post}}, & \text{otherwise}, \end{cases} \qquad 0 < \tau_q \le 1. \tag{9}$$

We then define the learned retrieval distribution

$$q_\psi(I_t \mid c_t, \mathcal{V}) = \mathrm{Cat}\big(\pi_t^{\mathrm{post}}\big), \qquad \pi_{t,j}^{\mathrm{post}} = \frac{\exp(\tilde{s}_{t,j}^{\mathrm{post}})}{\sum_{\ell=1}^{S} \exp(\tilde{s}_{t,\ell}^{\mathrm{post}})}. \tag{10}$$

Although I_t denotes the retrieved memory subset, the implementation uses a tractable categorical relaxation over the S candidate entries in the retrieved set. During training, the soft weights π_t^{post} are used as differentiable retrieval weights inside the attention generator. During inference, retrieval is performed using $q_\psi(I_t \mid c_t, \mathcal{V})$ rather than the cosine-similarity prior.

Coefficient posterior. We use a Gaussian posterior with diagonal covariance:

$$q_\psi(\theta_t \mid \mathcal{D}_t, I_t, c_t) = \mathcal{N}\big(\theta_t; \mu_{q,t}, \mathrm{diag}(\sigma^2_{q,t})\big). \tag{11}$$

For stability, our implementation shares posterior and prior variances, $\sigma^2_{q,t} \equiv \sigma^2_{\phi,t}$, and constructs the posterior mean as

$$\mu_{q,t} = \alpha \odot \mu_{\phi,t} + (1-\alpha) \odot \hat{\theta}^{\mathrm{self}}_t, \tag{12}$$

where $\hat{\theta}^{\mathrm{self}}_t$ is an optional task-specific estimator, such as a logistic-regression model trained on $\mathcal{D}_t$, and $\alpha \in (0,1)^f$ is a learned per-coordinate gate. Thus, the posterior interpolates between the retrieval-conditioned prior and the task-specific estimator, relying more on the prior when target-task supervision is scarce.

Reparameterization and prediction. We sample

$$\theta_t = \mu_{q,t} + \sigma_{q,t} \odot \varepsilon, \qquad \varepsilon \sim \mathcal{N}(0, I), \tag{13}$$

and predict with the GLM likelihood, e.g., $\sigma(\theta_t^\top x)$ for binary classification.

2.5 Variational objective with learned retrieval

The task evidence marginalizes over both the retrieved memory index and the target-task coefficients:

$$p_\phi(\mathcal{D}_t \mid c_t, \mathcal{V}) = \sum_{I_t} \int p_\phi(\mathcal{D}_t, \theta_t, I_t \mid c_t, \mathcal{V})\, d\theta_t. \tag{14}$$

Using the restricted variational family in Eq. (7), we optimize the negative ELBO:

Training objective.

$$\begin{aligned} \mathcal{J}_t(\phi, \psi) = &-\mathbb{E}_{q_\psi(I_t \mid c_t, \mathcal{V})} \mathbb{E}_{q_\psi(\theta_t \mid \mathcal{D}_t, I_t, c_t)} \big[\log p(\mathcal{D}_t \mid \theta_t)\big] \\ &+ \mathbb{E}_{q_\psi(I_t \mid c_t, \mathcal{V})} \Big[\mathrm{KL}(q_\psi(\theta_t \mid \mathcal{D}_t, I_t, c_t) \,\|\, p_\phi(\theta_t \mid I_t, c_t))\Big] \\ &+ \mathrm{KL}(q_\psi(I_t \mid c_t, \mathcal{V}) \,\|\, p(I_t \mid c_t, \mathcal{V}))\,. \end{aligned} \tag{15}$$

The likelihood term trains generated coefficients to explain target-task labels, the coefficient KL anchors the adapted posterior to the retrieval-conditioned prior, and the retrieval KL regularizes the learned retriever toward the semantic similarity prior. Thus, downstream prediction encourages retrieval of memory entries useful for coefficient synthesis, while the KL terms prevent unconstrained drift. A full derivation and additional theoretical analyses are given in Appendix F.

3 Results and Discussion

We evaluate RAIL on clinical procedure prediction tasks, where each task predicts whether a procedure should be administered to a patient given diagnostic measurements. This setting is naturally long-tailed: common procedures have abundant supervision, while many clinically meaningful procedures have few examples or are evaluated zero-shot. Our experiments assess: (i) low-data and zero-shot predictive performance, (ii) the role of meaningful retrieval, (iii) whether task embeddings recover transferable parameter neighborhoods, and (iv) whether RAIL's uncertainty estimates support reliability and interpretability.

3.1 Experimental Setup

Dataset. We construct clinical procedure prediction tasks from MIMIC-IV [14]. Each task predicts whether a procedure is administered from 217 diagnostic laboratory features, using the procedure definition as the task description. The dataset contains 7,487 procedure tasks: 4,412 with at least two positive examples for supervised/few-shot evaluation and 3,075 held-out tasks for zero-shot evaluation. Dataset construction, normalization, and task-regime details are given in Appendix B.

Tasks and regimes. Each procedure defines a binary prediction task t with diagnostic features $x \in \mathbb{R}^f$ (where $f = 217$), a natural-language task description c_t, and task-specific labels. Since all tasks share the same diagnostic-feature space, task-specific linear models can be represented in a common coefficient space. We evaluate across sample-scarcity regimes with $50+$, 10–49, 5–9, and 2–4 examples, and a held-out zero-shot regime. In zero-shot evaluation, target tasks are completely excluded from the retrieval memory: their descriptions, patient samples, labels, and task-specific coefficients are unseen during inference. RAIL receives only c_t, retrieves source tasks distinct from the target task, and deploys the retrieval-conditioned prior mean $\mu_{\phi,t}$ as the synthesized model.

Baselines, ablations, and metrics. We compare against supervised task-specific references, direct retrieval-transfer baselines, retrieval perturbations, and task-text encoder variants. Supervised references use target-task labels and are therefore not zero-shot methods. Retrieval-transfer baselines include Top-1 retrieved LR and Top-k coefficient averaging. Retrieval ablations replace normal retrieval with random or semantically distant source tasks. For task-text representations, we compare DistilBERT-base-uncased with the larger medical-domain MedEmbed-large-v0.1. We report task-averaged accuracy as the primary predictive metric, and use retrieval diagnostics, uncertainty-based failure detection, selective prediction, and coefficient-level uncertainty to evaluate interpretability and reliability.

3.2 Long-Tailed Task Distribution Motivates Retrieval-Augmented Model Generation

Clinical procedure tasks are highly imbalanced in sample availability: a small number of procedures have abundant supervision, while many have only a few labeled examples. This long-tail structure makes direct task-specific training unreliable in the low-data tail and motivates retrieval-augmented model generation, where related prior tasks provide parameter-level structure for new or data-scarce tasks.

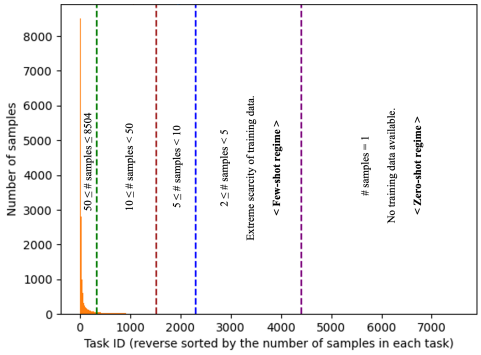


Figure 3: Long-tailed task distribution. Procedure tasks are highly imbalanced in sample availability.

Fig. 3 shows the long-tailed task distribution. Fig. 4 and Appendix Tab. 3 compare RAIL with the task-specific logistic regression across sample regimes. Here, 'LR oracle' denotes an LR model fit separately for each target task. In high-resource tasks, the supervised reference remains competitive, as expected. As target-task supervision decreases, however, the LR degrades sharply, dropping from approximately 0.76 to 0.42 in F1 and from 0.76 to 0.55 in accuracy between the $50+$ and 2–4 regimes. In contrast, RAIL remains comparatively stable, with F1 between 0.70 and 0.75 and accuracy between 0.72 and 0.75, indicating that retrieval-conditioned model generation provides useful prior structure when direct supervision is scarce.

3.3 Baselines and Main Ablations

We compare RAIL with supervised references, a retrieval-free meta-learning baseline, retrieval perturbations, and direct retrieval-transfer baselines. In addition to accuracy, Tab. 1 reports whether each method exposes clinically useful diagnostics: an inspectable predictor, task-text conditioning, retrieval traceability, and coefficient uncertainty.

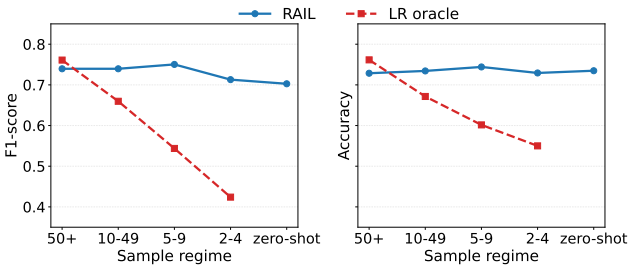


Figure 4: RAIL remains stable across sample-scarcity regimes, while the supervised task-specific LR oracle degrades in the low-data tail.

Tab. 1 shows that full RAIL provides the strongest combination of predictive performance and clinical inspectability. MedEmbed improves over DistilBERT, suggesting that clinically informed task-text representations provide stronger priors for procedure prediction. The

Table 1: Baseline and main ablation comparison. Diagnostic columns indicate whether a method provides an inspectable predictor, uses task-description embeddings, exposes retrieved source-task traces, or estimates coefficient uncertainty. For repeated runs, we report mean ± standard deviation over five independent runs.

Variant	Inspectable	Task text	Retrieval trace	Coeff. uncertainty	Few-shot acc.	Zero-shot acc.
Supervised references						
Task-specific MLP	×	×	×	×	0.533	N/A
Task-specific XGBoost	×	×	×	×	0.500	N/A
Task-specific Random Forest	×	×	×	×	0.573	N/A
Task-specific Logistic Regression	✓	×	×	×	0.550	N/A
Task-specific Decision Tree	✓	×	×	×	0.531	N/A
RAIL variants						
Full RAIL (w/ MedEmbed)	✓	✓	✓	✓	$\mathbf{0.732} \pm 0.009$	$\mathbf{0.734} \pm 0.011$
Full RAIL (w/ DistilBERT)	✓	✓	✓	✓	0.721 ± 0.010	0.718 ± 0.013
Without retrieval (w/ MedEmbed)	✓	✓	×	✓	0.703 ± 0.012	0.701 ± 0.014
Retrieval perturbation ablations						
RAIL + random retrieval	✓	✓	✓	✓	0.657 ± 0.019	0.659 ± 0.021
RAIL + distant retrieval	✓	✓	✓	✓	0.579 ± 0.008	0.587 ± 0.010
Retrieval-transfer baselines						
Top-1 retrieved LR	✓	✓	✓	×	0.587	0.585
Top-k coefficient average ($k = 100$)	✓	✓	✓	×	0.656	0.652

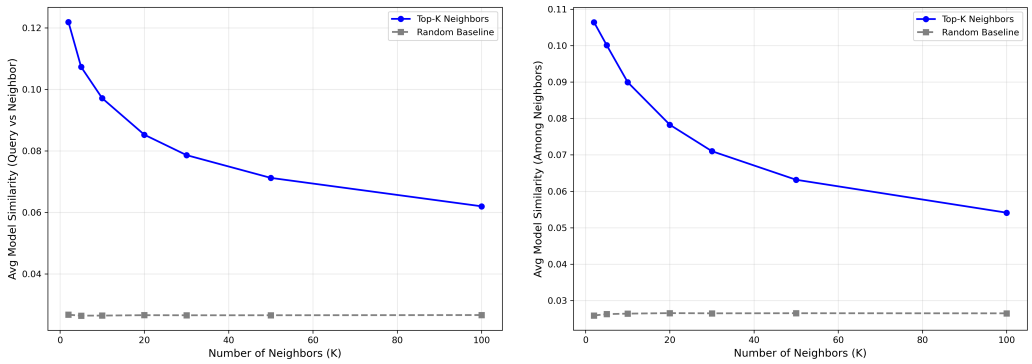
retrieval-free variant is competitive, empirically proving that task descriptions carry useful prior; however, full RAIL improves over it by grounding model generation in retrieved source-task coefficients and exposing a retrieval trace. Random and distant retrieval substantially degrade performance, showing that retrieval must be semantically meaningful. Direct retrieval-transfer baselines also underperform full RAIL, indicating that learned coefficient synthesis is more effective than direct model reuse or naive coefficient averaging. Thus, RAIL's benefit is not only higher low-data and zero-shot accuracy, but also the combination of feature-level inspectability, retrieval-grounded traceability, and uncertainty-aware diagnostics.

3.4 Stress-Testing Task-Embedding Space

Beyond the main predictive ablations in Tab. 1, we stress-test the task-representation space through perturbations, compression, and randomization. Additional robustness results are provided in Appendix Fig. 10. These analyses show that Gaussian noise degrades few-shot and zero-shot performance, compact embeddings preserve useful retrieval structure, and random or shuffled embeddings collapse retrieval quality. Together, these results indicate that RAIL depends on semantic organization in the task-embedding space, while useful retrieval structure can still be compressed.

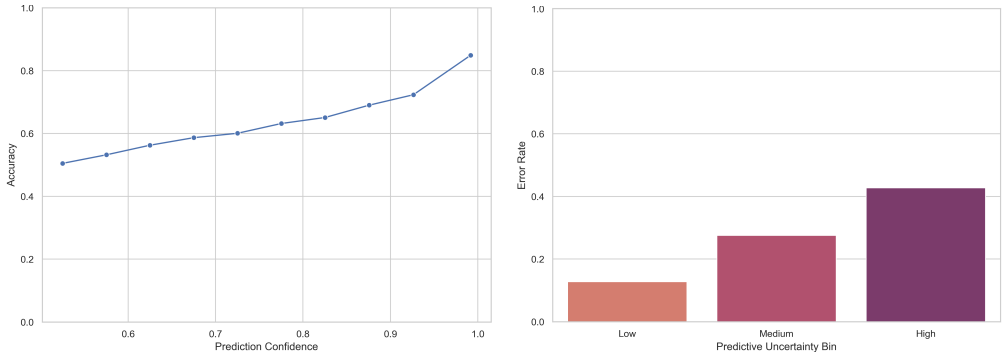
3.5 Retrieval and Embedding Diagnostics

RAIL assumes that semantically related task descriptions retrieve source tasks with transferable coefficient structure. We test this through embedding–model alignment and oracle retrieval consistency.



(a) Query vs. retrieved neighbors. (b) Pairwise similarity among neighbors.

Figure 5: Embedding–model alignment. Top-k retrieved tasks are closer to the query and more coherent among themselves than random neighbors.



(a) Confidence vs. accuracy.

(b) Error vs. uncertainty.

Figure 7: Predictive uncertainty diagnostics. **(a)** Predictive confidence tracks empirical accuracy. **(b)** Higher predictive uncertainty corresponds to higher error rates.

Embedding–model alignment. We first ask whether nearest neighbors in task-embedding space are also close in coefficient space. Fig. 5 compares top-k retrieved neighbors with random neighbors. Across embedding variants, retrieved neighbors are substantially more similar to the query task than random neighbors, and retrieved neighborhoods are more internally coherent. Similarity decreases as k grows, as broader neighborhoods include less related tasks. This supports RAIL's central mechanism: local neighborhoods in task-embedding space contain transferable parameter information.

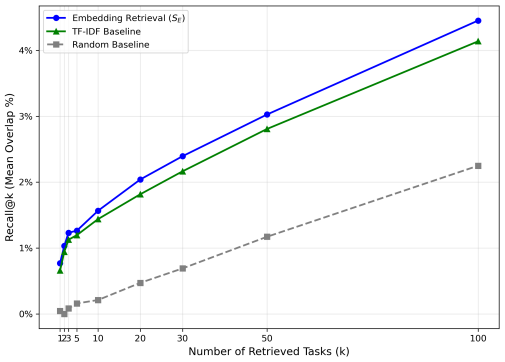


Figure 6: Oracle retrieval consistency. Embedding retrieval better matches oracle-relevant tasks than random retrieval and TF-IDF.

Oracle retrieval consistency. We further compare embedding retrieval with TF-IDF and random retrieval using an oracle-overlap criterion. As shown in Fig. 6, embedding retrieval achieves higher oracle overlap across retrieval depths than random retrieval and is competitive with or better than TF-IDF. Thus, task embeddings capture similarity relevant for model transfer rather than only lexical overlap.

3.6 Uncertainty Diagnostics and Selective Prediction

RAIL provides retrieval, coefficient, and predictive uncertainty. We focus on predictive uncertainty for reliability analysis: we sample $K = 100$ coefficient vectors from the posterior, compute the induced predictive probabilities, and use the binary entropy of the posterior-mean probability as the uncertainty score. Prediction confidence is $\max(\bar{p}, 1 - \bar{p})$.

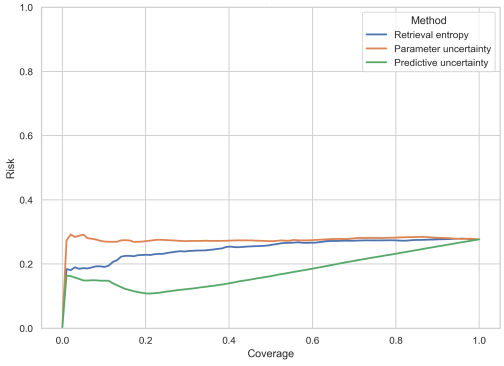


Figure 8: Selective prediction. Lower-uncertainty predictions yield lower risk at reduced coverage.

Prediction confidence aligns with empirical correctness (Fig. 7a and 7b): accuracy increases monotonically from low- to high-confidence bins, while error rate increases from low- to high-uncertainty bins. We further define a sample-level failure as $\mathbf{1}[\hat{y} \neq y]$ and evaluate uncertainty scores using AUROC and AUPRC. Predictive uncertainty is the strongest failure-detection signal, achieving AUROC 0.683 and AUPRC 0.419, outperforming retrieval entropy and parameter uncertainty; ROC and precision–recall curves are provided in Appendix Fig. 11. Thus, output-level uncertainty is most aligned with prediction-level failure, while retrieval and coefficient uncertainty provide complementary task- and model-level diagnostics.

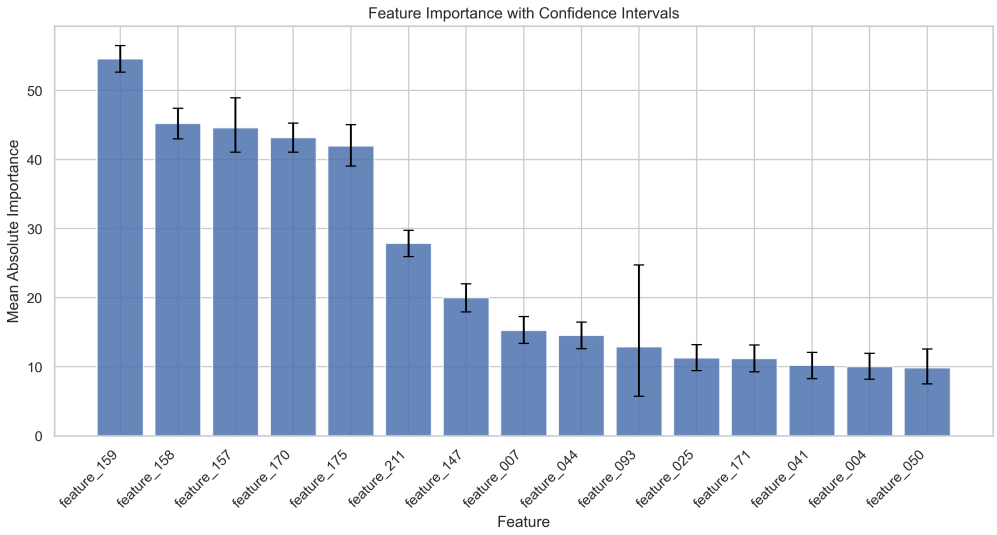


Figure 9: Feature-level interpretability with uncertainty. RAIL ranks diagnostic features by posterior coefficient magnitude and reports uncertainty intervals.

Finally, we evaluate uncertainty-based abstention by ranking examples by increasing uncertainty and retaining the lowest-uncertainty subset. Fig. 8 shows that risk decreases as coverage decreases, suggesting that uncertain predictions can be flagged for expert review rather than used automatically.

3.7 Coefficient-Level Interpretability with Uncertainty

RAIL generates linear models in the original diagnostic-feature space, so each coefficient corresponds to an inspectable input feature. Its posterior over coefficients further enables uncertainty-aware interpretation.

For each task and feature, we record posterior coefficient mean, magnitude, variance, and standard deviation (Fig. 9). Stable features have large posterior importance with narrow intervals, while uncertain features have wider intervals and should be interpreted cautiously. Additional stable and unstable coefficient examples are provided in Appendix Fig. 12. This coefficient-level uncertainty supports feature-level inspection while making explanation stability explicit.

3.8 Summary and Limitations

Overall, RAIL is most useful in the low-data tail of clinical procedure prediction, where task-specific supervised models degrade and zero-shot synthesis becomes necessary. Retrieval and embedding diagnostics show that its gains depend on meaningful task representations and semantically aligned source-task retrieval, while ablations show that learned coefficient synthesis improves over direct transfer or naive averaging. Uncertainty analyses further show that RAIL can flag unreliable predictions and unstable explanations for review. Limitations are that RAIL assumes a shared diagnostic-feature space and depends on task-memory coverage; when no related source tasks exist, retrieval-conditioned synthesis may be less reliable. Coefficient explanations are also associational rather than causal.

4 Conclusions

We introduced **RAIL**, a retrieval-augmented probabilistic meta-learning framework for generating task-specific interpretable clinical prediction models from natural-language task descriptions. By retrieving prior task predictors and synthesizing structure directly in coefficient space, RAIL produces models in the original diagnostic-feature space, preserving feature-level interpretability while supporting uncertainty-aware prediction. Across long-tailed clinical procedure tasks, RAIL is most effective when target-task supervision is scarce or unavailable, and our ablations show that its gains rely on clinically informed task representations, meaningful retrieval, and learned coefficient synthesis. Together, these results suggest a practical path toward scalable clinical prediction systems that can adapt to newly emerging or low-resource tasks without sacrificing inspectability, uncertainty awareness, or compatibility with human oversight.

Acknowledgments and Disclosure of Funding

This research has been graciously funded by the National Science Foundation (NSF) awards BCS2040381, CNS2414087 and IIS2123952 (to S.M. and E.X.); the Defense Advanced Research Projects Agency (DARPA) award HR00112390063 (to S.M. and E.X.); the Semiconductor Research Corporation (SRC) AIHW award 2024AH3210 (to S.M. and E.X.); and, the National Institutes of Health (NIH) award R01GM140467 (to C.E. and E.X.). Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the author(s) and do not necessarily reflect the views of the National Science Foundation, the Defense Advanced Research Projects Agency, the Semiconductor Research Corporation, and the National Institutes of Health.

References

[1] Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. Meta-learning in neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(9):5149–5169, 2021.

[2] Benjamin Lengerich, Caleb N Ellington, Andrea Rubbi, Manolis Kellis, and Eric P Xing. Contextualized machine learning. *arXiv preprint arXiv:2310.11340*, 2023.

[3] Caleb N Ellington, Benjamin J Lengerich, Thomas BK Watkins, Jiekun Yang, Abhinav K Adduri, Sazan Mahbub, Hanxi Xiao, Manolis Kellis, and Eric P Xing. Learning to estimate sample-specific transcriptional networks for 7,000 tumors. *Proceedings of the National Academy of Sciences*, 122(21):e2411930122, 2025.

[4] Jannik Deuschel, Caleb Ellington, Yingtao Luo, Ben Lengerich, Pascal Friederich, and Eric P Xing. Contextualized policy recovery: Modeling and interpreting medical decisions with adaptive imitation learning. In *Forty-first International Conference on Machine Learning*, 2024.

[5] Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. Large language models in medicine. *Nature medicine*, 29(8):1930–1940, 2023.

[6] Yuqing Du, Olivia Watkins, Zihan Wang, Cédric Colas, Trevor Darrell, Pieter Abbeel, Abhishek Gupta, and Jacob Andreas. Guiding pretraining in reinforcement learning with large language models, 2023.

[7] Thommen George Karimpanal, Laknath Buddhika Semage, Santu Rana, Hung Le, Truyen Tran, Sunil Gupta, and Svetha Venkatesh. Lagr-seq: Language-guided reinforcement learning with sample-efficient querying, 2023.

[8] Wanpeng Zhang and Zongqing Lu. Adarefiner: Refining decisions of language models with adaptive feedback, 2024.

[9] Dyah Adila, Changho Shin, Linrong Cai, and Frederic Sala. Zero-shot robustification of zero-shot models, 2024.

[10] Stephanie Long, Alexandre Piché, Valentina Zantedeschi, Tibor Schuster, and Alexandre Drouin. Causal discovery with language models as imperfect experts, 2023.

[11] Chenxi Liu, Yongqiang Chen, Tongliang Liu, Mingming Gong, James Cheng, Bo Han, and Kun Zhang. Discovery of the hidden world with large language models, 2024.

[12] Mingyu Jin, Qinkai Yu, Dong Shu, Chong Zhang, Lizhou Fan, Wenyue Hua, Suiyuan Zhu, Yanda Meng, Zhenting Wang, Mengnan Du, et al. Health-llm: Personalized retrieval-augmented disease prediction system. *arXiv preprint arXiv:2402.00746*, 2024.

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[14] Alistair EW Johnson, Lucas Bulgarelli, Lu Shen, Alvin Gayles, Ayad Shammout, Steven Horng, Tom J Pollard, Sicheng Hao, Benjamin Moody, Brian Gow, et al. Mimic-iv, a freely accessible electronic health record dataset. *Scientific data*, 10(1):1, 2023.

[15] Sana Tonekaboni, Shalmali Joshi, Melissa D McCradden, and Anna Goldenberg. What clinicians want: contextualizing explainable machine learning for clinical end use. In *Machine learning for healthcare conference*, pages 359–380. PMLR, 2019.

[16] Ben Lengerich, Bryon Aragam, and Eric P Xing. Learning sample-specific models with low-rank personalized regression. *Advances in Neural Information Processing Systems*, 32, 2019.

[17] Hayder Abbood MD. Meta-learning approaches for causal discovery in dynamic healthcare and robotics environments. *Mesopotamian Journal of Artificial Intelligence in Healthcare*, 2025:136–153, 2025.

[18] Xi Sheryl Zhang, Fengyi Tang, Hiroko H Dodge, Jiayu Zhou, and Fei Wang. Metapred: Meta-learning for clinical risk prediction with limited patient electronic health records. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2487–2495, 2019.

[19] Yanchao Tan, Carl Yang, Xiangyu Wei, Chaochao Chen, Weiming Liu, Longfei Li, Jun Zhou, and Xiaolin Zheng. Metacare++: Meta-learning with hierarchical subtyping for cold-start diagnosis prediction in healthcare data. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 449–459, 2022.

[20] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

[21] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2023.

[22] Sazan Mahbub, Souvik Kundu, and Eric P Xing. Prism: Enhancing protein inverse folding through fine-grained retrieval on structure-sequence multimodal representations. *arXiv preprint arXiv:2510.11750*, 2025.

[23] Maria Mahbub, Sudarshan Srinivasan, Edmon Begoli, and Gregory D Peterson. Bioadapt-mrc: adversarial learning-based domain adaptation improves biomedical machine reading comprehension task. *Bioinformatics*, 38(18):4369–4379, 2022.

[24] Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Michael Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling, 2021.

[25] Emre Kıcıman, Robert Ness, Amit Sharma, and Chenhao Tan. Causal reasoning and large language models: Opening a new frontier for causality, 2024.

[26] Kristy Choi, Chris Cundy, Sanjari Srivastava, and Stefano Ermon. Lmpriors: Pre-trained language models as task-specific priors, 2022.

[27] Shuang Li, Xavier Puig, Chris Paxton, Yilun Du, Clinton Wang, Linxi Fan, Tao Chen, De-An Huang, Ekin Akyürek, Anima Anandkumar, et al. Pre-trained language models for interactive decision-making. *Advances in Neural Information Processing Systems*, 35:31199–31212, 2022.

[28] Xue Yan, Yan Song, Xidong Feng, Mengyue Yang, Haifeng Zhang, Haitham Bou Ammar, and Jun Wang. Efficient reinforcement learning with large language model priors. In *13th International Conference on Learning Representations Iclr 2025*, pages 30818–30842. ICLR, 2025.

[29] Sebastian Bordt, Ben Lengerich, Harsha Nori, and Rich Caruana. Data science with llms and interpretable models, 2024.

A Related Works

Clinical predictive modeling often faces a data-scarcity problem: many clinically relevant tasks have limited supervision, yet the resulting models must remain accurate, interpretable, and reliable enough to support expert review. Several research directions address different parts of this challenge. Contextualized machine learning studies prediction settings where the input–output relationship varies across contexts, individuals, or tasks, which is common in clinical and decision-making domains [2, 4, 15, 16, 3]. Meta-learning similarly seeks to use experience across tasks to support rapid adaptation to new tasks with limited data [17, 18, 19, 1]. Recent meta-models and task-conditioned predictors extend this idea by dynamically generating or adapting predictors for particular contexts or tasks [1, 2, 3, 4]. These methods provide a foundation for scalable low-data prediction, but many rely on latent representations or flexible neural predictors rather than directly inspectable models in the original feature space.

Another approach is to use retrieval to provide task-relevant external information. Retrieval-augmented generation conditions a model on retrieved content, allowing it to use information beyond what is stored in its parameters [20, 21]. In biomedical domain, retrieval-augmented pipelines have been used to incorporate health reports and medical knowledge into large models for improved feature extraction and prediction [12, 22]. Retrieval is therefore a natural mechanism for grounding predictions in relevant prior information. However, standard RAG systems typically retrieve documents or knowledge snippets for language generation or latent prediction, rather than using retrieved prior predictors as reusable task-level structure.

A complementary direction uses pretrained language models as sources of prior knowledge when labeled data are scarce [5, 23]. Language-model priors have been used in reinforcement learning to guide exploration and decision refinement [24, 6, 7, 8], in feature selection to identify useful variables [9], and in causal discovery or reasoning to propose or evaluate plausible causal structure [10, 25, 11]. Related work has also studied language-model priors across multiple decision-making settings [26, 27, 28]. These studies show that pretrained representations can encode useful task-level information, but this information is often used to guide black-box decision processes, construct latent features, or generate natural-language rationales rather than directly producing task-specific interpretable predictors.

Finally, work combining language models with interpretable models aims to leverage language-derived knowledge while preserving transparency in the final decision process [29]. This direction is especially relevant for healthcare, where users may need to inspect feature-level evidence, assess reliability, and decide when additional review is needed. Overall, prior work offers important tools for low-data prediction—task adaptation, retrieval, language-derived priors, and interpretability—but leaves open the challenge of combining these ingredients into a framework that produces task-specific predictors that are simultaneously data-efficient, directly inspectable, and uncertainty-aware.

B Dataset and Preprocessing

We use the MIMIC-IV dataset [14] to construct clinical procedure prediction tasks. The preprocessed data consist of two main tables: laboratory events and procedure records. Both tables are grouped by hospital admission and patient identifiers, (`hadm_id`, `subject_id`). Laboratory events provide diagnostic measurements, which form the input features, while procedure records define task-specific binary labels indicating whether a procedure was administered during a hospital stay. We retain only admissions with corresponding laboratory measurements so that every labeled example has an associated diagnostic feature vector.

Each input vector contains 217 diagnostic features derived from laboratory measurements. Since laboratory items have different clinical reference ranges, we normalize each raw measurement using its item-specific lower and upper reference values:

$$\text{normalized\_feature} = \frac{\text{raw\_value} - \text{lower\_range}}{\text{upper\_range} - \text{lower\_range}}. \tag{16}$$

Values outside the reference interval are not clipped; this preserves clinically meaningful deviations while placing heterogeneous diagnostics on a comparable scale.

Each clinical procedure defines a separate binary prediction task. The dataset contains $7{,}487$ procedure tasks in total. Among these, $4{,}412$ procedures have at least two positive examples and are used for high-resource, low-resource, and few-shot evaluation. For each of these tasks, we construct balanced task-specific datasets with equal numbers of positive and negative examples and use a $50/50$ train–test split within the task. The remaining $3{,}075$ procedure tasks are used for zero-shot evaluation, where no target-task training examples or task-specific oracle models are available during inference. Tab. 2 summarizes the resulting task regimes.

We use natural-language procedure definitions as task descriptions. Example descriptions include "Respiratory Ventilation, 24–96 Consecutive Hours," "Inspection of Upper Intestinal Tract, Via Natural or Artificial Opening Endoscopic," "Fluoroscopy of Multiple Coronary Arteries using Other Contrast," and "Introduction of Other Antineoplastic into Central Vein, Percutaneous Approach." These descriptions provide the task-text signal used by RAIL for retrieval-augmented model generation.

Table 2: Summary of task regimes used in RAIL evaluation. Each clinical procedure defines one binary prediction task. The zero-shot regime contains held-out target procedures whose labels and task-specific oracle models are excluded from the retrieval memory during inference.

Regime	Task ID range	Examples per task
High-resource	0–341	50+
Moderate-resource	342–1521	10–49
Low-resource	1522–2306	5–9
Few-shot	2307–4411	2–4
Zero-shot	4412–7486	No target-task labels used

C Hardware Specifications

All experiments were run on a single workstation equipped with 12 CPU cores of AMD EPYC 9354 processor at 3.25 GHz, 100 GB of RAM, and one NVIDIA RTX A6000 GPU with 48 GB of VRAM.

D Additional Experimental Details

This section provides supplementary tables for the experimental setup and analyses in the main paper. Tab. 3 reports the full performance breakdown across sample-scarcity regimes. Tab. 4 summarizes the ablation and diagnostic suite. Tab. 5 defines the uncertainty signals used in reliability analyses, and Tab. 6 summarizes the interpretable outputs exposed by RAIL.

Table 3: Performance across sample-scarcity regimes. The task-specific LR oracle uses target-task labels and remains strongest when sufficient supervision is available, while RAIL is most beneficial in low-data regimes.

Sample regime	**RAIL**				**LR oracle**			
	Precision	Recall	F1	Accuracy	Precision	Recall	F1	Accuracy
50+	0.7033	0.7883	0.7396	0.7286	0.7639	0.7641	0.7610	0.7615
10–49	0.7258	0.7788	0.7396	0.7343	0.6953	0.6689	0.6597	0.6715
5–9	0.7455	0.8002	0.7503	0.7441	0.5975	0.5745	0.5437	0.6016
2–4 (few-shot)	0.6878	0.7876	0.7129	0.7293	0.3860	0.5285	0.4240	0.5500
Zero-shot	0.6621	0.7837	0.7027	0.7348	N/A	N/A	N/A	N/A

E Additional Retrieval Robustness, Uncertainty, and Interpretability Diagnostics

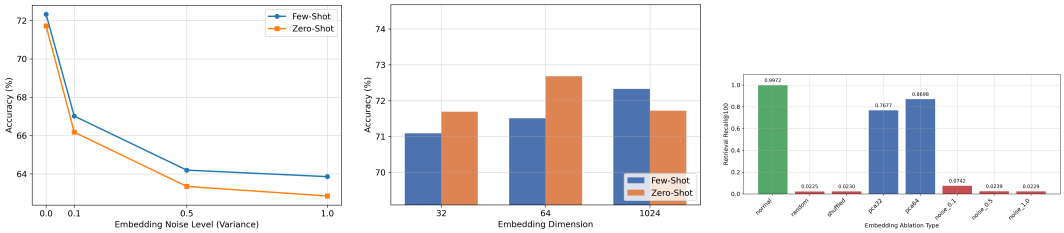
This section provides additional diagnostic figures supporting the retrieval robustness, uncertainty, and coefficient-level interpretability analyses in the main paper. Fig. 10 reports controlled perturbations of the task-representation space. Fig. 11 reports ROC and precision–recall curves for uncertainty-based failure detection. Fig. 12 shows examples of stable and unstable coefficient-level explanations.

Table 4: Summary of RAIL ablation and diagnostic analyses.

Analysis group	Variants tested	Purpose
Text encoder	DistilBERT, MedEmbed	Tests effect of clinical task representations.
Retrieval perturbation	Normal, random, distant	Tests causal role of meaningful retrieval.
Transfer baseline	Top-1 LR, Top-k coefficient average	Tests learned synthesis vs direct transfer.
Embedding corruption	Gaussian noise	Tests sensitivity to semantic corruption.
Embedding compression	PCA-32, PCA-64	Tests whether retrieval structure is compressible.
Embedding randomization	Random, shuffled	Tests whether structure exceeds chance.
Retrieval diagnostics	Query-neighbor, neighbor-neighbor, Oracle Recall@k	Tests semantic/model alignment.
Uncertainty diagnostics	Retrieval, parameter, predictive uncertainty	Tests reliability and failure detection.

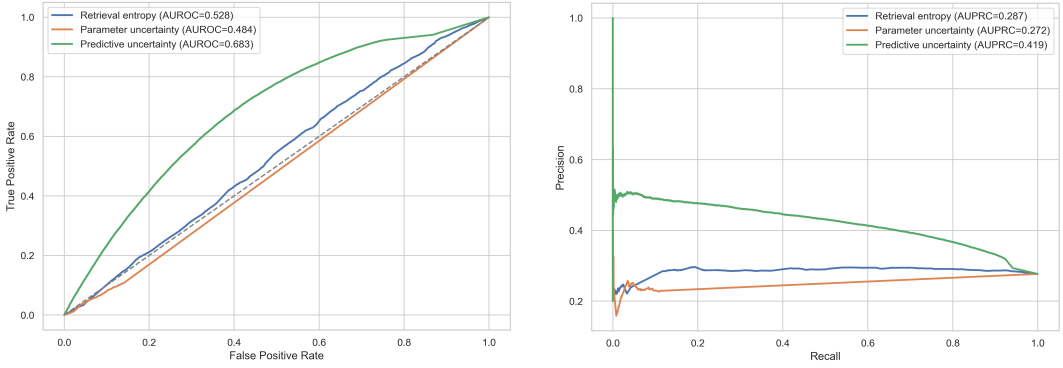
Table 5: Uncertainty signals used in RAIL diagnostics.

Signal	Definition	Level
Retrieval uncertainty	$-\sum_j q_\psi(I_t = j) \log q_\psi(I_t = j)$	Task
Parameter uncertainty	$\sum_r \sigma^2_{q,t,r}$	Task/model
Predictive uncertainty	$-\bar{p} \log \bar{p} - (1-\bar{p}) \log(1-\bar{p})$	Sample
Confidence	$\max(\bar{p}, 1-\bar{p})$	Sample



(a) Noise perturbation. (b) Embedding dimension. (c) Retrieval ablations.

Figure 10: Additional representation ablations. Noise degrades RAIL performance, compact embeddings preserve useful retrieval structure, and random or shuffled embeddings collapse retrieval quality.



(a) Failure-detection ROC. (b) Failure-detection PR.

Figure 11: Failure detection using uncertainty. Predictive uncertainty is the strongest signal for identifying likely incorrect predictions, outperforming retrieval entropy and parameter uncertainty in both ROC and precision–recall analyses.

F Additional Theoretical Analysis

F.1 ELBO derivation for learned retrieval

We derive the objective used in Eq. (15). Recall the conditional generative model

$$p_\phi(\mathcal{D}_t, \theta_t, I_t \mid c_t, \mathcal{V}) = p(\mathcal{D}_t \mid \theta_t)\; p_\phi(\theta_t \mid I_t, c_t)\; p(I_t \mid c_t, \mathcal{V}). \tag{17}$$

Table 6: Interpretability outputs produced by RAIL.

Output	Level	Interpretation
Generated coefficients θ_t	Task	Interpretable diagnostic-feature weights.
Posterior coefficient variance	Feature	Uncertainty in each feature's contribution.
Top-k feature frequency	Feature	Stability of feature importance across posterior samples.
Retrieval posterior weights	Task memory	Attribution to retrieved source tasks.
Prediction confidence	Sample	Reliability of individual predictions.

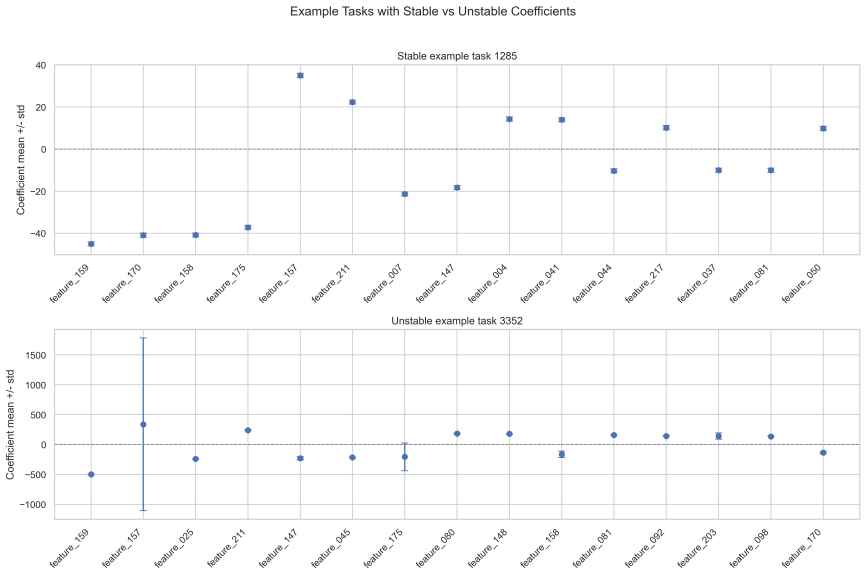


Figure 12: Stable and unstable coefficient examples. Stable tasks exhibit top coefficients with relatively narrow uncertainty intervals, while unstable tasks show wider intervals for some features, indicating explanations that should be interpreted with caution.

Here, I_t denotes the retrieved memory object induced by the retriever. In the exact set-valued view, $I_t \subseteq \mathcal{V}$. In implementation, we use a tractable top-S categorical relaxation over the retrieved candidate entries; the closed-form categorical KL below corresponds to this implemented relaxation.

The marginal evidence for task t is

$$p_\phi(\mathcal{D}_t \mid c_t, \mathcal{V}) = \sum_{I_t} \int p_\phi(\mathcal{D}_t, \theta_t, I_t \mid c_t, \mathcal{V})\, d\theta_t, \tag{18}$$

where the summation is over the support of the retrieval distribution.

For any variational distribution $q_\psi(\theta_t, I_t \mid \mathcal{D}_t, c_t, \mathcal{V})$, we can write

$$\log p_\phi(\mathcal{D}_t \mid c_t, \mathcal{V}) = \log \sum_{I_t} \int q_\psi(\theta_t, I_t \mid \mathcal{D}_t, c_t, \mathcal{V}) \frac{p_\phi(\mathcal{D}_t, \theta_t, I_t \mid c_t, \mathcal{V})}{q_\psi(\theta_t, I_t \mid \mathcal{D}_t, c_t, \mathcal{V})}\, d\theta_t \tag{19}$$

$$\geq \mathbb{E}_{q_\psi(\theta_t, I_t \mid \mathcal{D}_t, c_t, \mathcal{V})} \left[\log \frac{p_\phi(\mathcal{D}_t, \theta_t, I_t \mid c_t, \mathcal{V})}{q_\psi(\theta_t, I_t \mid \mathcal{D}_t, c_t, \mathcal{V})} \right], \tag{20}$$

where the inequality follows from Jensen's inequality. Thus,

$$\mathcal{L}^{(t)}_{\mathrm{ELBO}} = \mathbb{E}_{q_\psi} \left[\log p_\phi(\mathcal{D}_t, \theta_t, I_t \mid c_t, \mathcal{V}) - \log q_\psi(\theta_t, I_t \mid \mathcal{D}_t, c_t, \mathcal{V}) \right]. \tag{21}$$

We use the restricted amortized factorization

$$q_\psi(\theta_t, I_t \mid \mathcal{D}_t, c_t, \mathcal{V}) = q_\psi(I_t \mid c_t, \mathcal{V})\; q_\psi(\theta_t \mid \mathcal{D}_t, I_t, c_t). \tag{22}$$

The retrieval factor is intentionally label-free. This restriction preserves zero-shot retrieval at inference time, where $\mathcal{D}_t$ is unavailable. It is still a valid variational family, but it may yield a looser bound than a fully label-conditioned retrieval posterior.

Substituting Eq. (17) and Eq. (22) into Eq. (21) gives

$$\begin{aligned}\mathcal{L}_{\mathrm{ELBO}}^{(t)} = \mathbb{E}_{q_\psi}\Big[&\log p(\mathcal{D}_t \mid \theta_t) + \log p_\phi(\theta_t \mid I_t, c_t) + \log p(I_t \mid c_t, \mathcal{V}) \\ &- \log q_\psi(\theta_t \mid \mathcal{D}_t, I_t, c_t) - \log q_\psi(I_t \mid c_t, \mathcal{V})\Big].\end{aligned} \tag{23}$$

Grouping the likelihood, coefficient, and retrieval terms yields

$$\begin{aligned}\mathcal{L}_{\mathrm{ELBO}}^{(t)} = &\mathbb{E}_{q_\psi(I_t|c_t,\mathcal{V})}\mathbb{E}_{q_\psi(\theta_t|\mathcal{D}_t,I_t,c_t)}\big[\log p(\mathcal{D}_t \mid \theta_t)\big] \\ &+ \mathbb{E}_{q_\psi(I_t|c_t,\mathcal{V})}\mathbb{E}_{q_\psi(\theta_t|\mathcal{D}_t,I_t,c_t)}\left[\log \frac{p_\phi(\theta_t \mid I_t, c_t)}{q_\psi(\theta_t \mid \mathcal{D}_t, I_t, c_t)}\right] \\ &+ \mathbb{E}_{q_\psi(I_t|c_t,\mathcal{V})}\left[\log \frac{p(I_t \mid c_t, \mathcal{V})}{q_\psi(I_t \mid c_t, \mathcal{V})}\right].\end{aligned} \tag{24}$$

Recognizing the last two terms as negative KL divergences gives

$$\begin{aligned}\mathcal{L}_{\mathrm{ELBO}}^{(t)} = &\mathbb{E}_{q_\psi(I_t|c_t,\mathcal{V})}\mathbb{E}_{q_\psi(\theta_t|\mathcal{D}_t,I_t,c_t)}\big[\log p(\mathcal{D}_t \mid \theta_t)\big] \\ &- \mathbb{E}_{q_\psi(I_t|c_t,\mathcal{V})}\Big[\mathrm{KL}(q_\psi(\theta_t \mid \mathcal{D}_t, I_t, c_t) \,\|\, p_\phi(\theta_t \mid I_t, c_t))\Big] \\ &- \mathrm{KL}(q_\psi(I_t \mid c_t, \mathcal{V}) \,\|\, p(I_t \mid c_t, \mathcal{V}))\,.\end{aligned} \tag{25}$$

This leads to the negative ELBO objective in Eq. (15). The first term trains the generated coefficients to explain the target-task labels. The second term anchors the coefficient posterior to the retrieval-conditioned coefficient prior. The third term trains the learned retriever relative to the cosine-similarity retrieval prior.

Closed-form KL terms. When both the coefficient posterior and prior are diagonal Gaussians,

$$q_\psi(\theta_t \mid \cdot) = \mathcal{N}(\theta_t; \mu_{q,t}, \mathrm{diag}(\sigma_{q,t}^2)), \qquad p_\phi(\theta_t \mid \cdot) = \mathcal{N}(\theta_t; \mu_{\phi,t}, \mathrm{diag}(\sigma_{\phi,t}^2)), \tag{26}$$

the general diagonal Gaussian KL is

$$\mathrm{KL}\big(q_\psi(\theta_t \mid \cdot) \,\|\, p_\phi(\theta_t \mid \cdot)\big) = \frac{1}{2}\sum_{r=1}^{f}\left[\log \frac{\sigma_{\phi,t}^{2(r)}}{\sigma_{q,t}^{2(r)}} + \frac{\sigma_{q,t}^{2(r)} + (\mu_{q,t}^{(r)} - \mu_{\phi,t}^{(r)})^2}{\sigma_{\phi,t}^{2(r)}} - 1\right]. \tag{27}$$

In our implementation, we tie the posterior and prior variances, $\sigma_{q,t}^2 \equiv \sigma_{\phi,t}^2$, so Eq. (27) simplifies to

$$\mathrm{KL}\big(q_\psi(\theta_t \mid \cdot) \,\|\, p_\phi(\theta_t \mid \cdot)\big) = \frac{1}{2}\sum_{r=1}^{f}\frac{(\mu_{q,t}^{(r)} - \mu_{\phi,t}^{(r)})^2}{\sigma_{\phi,t}^{2(r)}}. \tag{28}$$

For the implemented categorical retrieval relaxation, $q_\psi(I_t \mid \cdot) = \mathrm{Cat}(\pi_t^{\mathrm{post}})$ and $p(I_t \mid \cdot) = \mathrm{Cat}(\pi_t^{\mathrm{prior}})$, the retrieval KL is

$$\mathrm{KL}\big(q_\psi(I_t \mid \cdot) \,\|\, p(I_t \mid \cdot)\big) = \sum_{j=1}^{S}\pi_{t,j}^{\mathrm{post}}\log\frac{\pi_{t,j}^{\mathrm{post}}}{\pi_{t,j}^{\mathrm{prior}}}. \tag{29}$$

Practical estimator. We estimate the expected log-likelihood term with one reparameterized coefficient sample θ_t per minibatch and minimize a weighted negative-ELBO objective:

$$\widehat{\mathcal{J}}_t = \underbrace{\mathrm{BCE}\big(y, \sigma(\theta_t^\top x)\big)}_{\text{data term}} + \beta_\theta \underbrace{\mathrm{KL}_\theta}_{(28)} + \beta_I \underbrace{\mathrm{KL}_I}_{(29)}. \tag{30}$$

The weights β_θ and β_I control the strength of coefficient-prior anchoring and retrieval-prior anchoring, respectively. When $\beta_\theta = \beta_I = 1$, this corresponds to the negative ELBO above; otherwise it is a weighted negative-ELBO objective.

F.2 Prior anchoring and generalization

We give a simple Rademacher-complexity argument showing why anchoring the coefficient posterior near a retrieval-conditioned prior can reduce the effective hypothesis class in low-data regimes. This result is not intended to characterize the full neural retrieval generator. Rather, it formalizes the intuition that, once the retrieved prior mean is fixed, constraining coefficients to remain near that mean controls the complexity of the induced linear predictors.

Assumption 1 (Bounded features and bounded Lipschitz loss)**.** *The input features satisfy $\|x\|_2 \le R$ almost surely. The loss $\ell(\hat{y}, y)$ is L_ℓ-Lipschitz in the prediction $\hat{y}$ and takes values in an interval of length at most C_ℓ over the hypothesis class considered below.*

Proposition 1 (Generalization benefit of prior anchoring)**.** *Fix a retrieval-conditioned prior mean $\mu_{\phi,t}$ independently of the target-task sample used in the bound, and consider the hypothesis class*

$$\mathcal{H}_B = \{x \mapsto \theta^\top x : \|\theta - \mu_{\phi,t}\|_2 \le B\}. \tag{31}$$

Then the empirical Rademacher complexity satisfies

$$\mathfrak{R}_m(\mathcal{H}_B) \le \frac{RB}{\sqrt{m}}. \tag{32}$$

Moreover, under Assumption 1, with probability at least $1-\delta$ over a sample of size m, for all θ such that $\|\theta - \mu_{\phi,t}\|_2 \le B$,

$$\mathcal{L}(\theta) \le \widehat{\mathcal{L}}(\theta) + 2L_\ell\, \mathfrak{R}_m(\mathcal{H}_B) + 3C_\ell\sqrt{\frac{\log(2/\delta)}{2m}} \le \widehat{\mathcal{L}}(\theta) + \frac{2L_\ell RB}{\sqrt{m}} + 3C_\ell\sqrt{\frac{\log(2/\delta)}{2m}}. \tag{33}$$

Proof. Let $S_m = \{x_1, \dots, x_m\}$ be a fixed sample and let $\sigma_1, \dots, \sigma_m$ be independent Rademacher variables. The empirical Rademacher complexity of $\mathcal{H}_B$ is

$$\mathfrak{R}_m(\mathcal{H}_B) = \mathbb{E}_\sigma\left[\sup_{\|\theta-\mu_{\phi,t}\|_2\le B} \frac{1}{m}\sum_{i=1}^m \sigma_i \theta^\top x_i\right]. \tag{34}$$

Write $\theta = \mu_{\phi,t} + u$, where $\|u\|_2 \le B$. Then

$$\mathfrak{R}_m(\mathcal{H}_B) = \mathbb{E}_\sigma\left[\sup_{\|u\|_2\le B} \frac{1}{m}\sum_{i=1}^m \sigma_i (\mu_{\phi,t} + u)^\top x_i\right] \tag{35}$$

$$= \mathbb{E}_\sigma\left[\frac{1}{m}\sum_{i=1}^m \sigma_i \mu_{\phi,t}^\top x_i + \sup_{\|u\|_2\le B} u^\top\left(\frac{1}{m}\sum_{i=1}^m \sigma_i x_i\right)\right]. \tag{36}$$

The first term vanishes in expectation over the Rademacher variables because $\mathbb{E}_\sigma[\sigma_i] = 0$. Therefore,

$$\mathfrak{R}_m(\mathcal{H}_B) = \mathbb{E}_\sigma\left[\sup_{\|u\|_2\le B} u^\top\left(\frac{1}{m}\sum_{i=1}^m \sigma_i x_i\right)\right]. \tag{37}$$

By Cauchy–Schwarz,

$$\sup_{\|u\|_2\le B} u^\top\left(\frac{1}{m}\sum_{i=1}^m \sigma_i x_i\right) \le \frac{B}{m}\left\|\sum_{i=1}^m \sigma_i x_i\right\|_2. \tag{38}$$

Thus,

$$\mathfrak{R}_m(\mathcal{H}_B) \le \frac{B}{m}\mathbb{E}_\sigma\left\|\sum_{i=1}^m \sigma_i x_i\right\|_2. \tag{39}$$

Applying Jensen's inequality,

$$\mathbb{E}_\sigma\left\|\sum_{i=1}^m \sigma_i x_i\right\|_2 \le \left(\mathbb{E}_\sigma\left\|\sum_{i=1}^m \sigma_i x_i\right\|_2^2\right)^{1/2}. \tag{40}$$

Expanding the squared norm gives

$$\mathbb{E}_\sigma \left\| \sum_{i=1}^{m} \sigma_i x_i \right\|_2^2 = \mathbb{E}_\sigma \left[\sum_{i=1}^{m} \sum_{j=1}^{m} \sigma_i \sigma_j x_i^\top x_j \right] \tag{41}$$

$$= \sum_{i=1}^{m} \|x_i\|_2^2, \tag{42}$$

because $\mathbb{E}[\sigma_i \sigma_j] = 0$ for $i \neq j$ and $\mathbb{E}[\sigma_i^2] = 1$. Since $\|x_i\|_2 \leq R$,

$$\sum_{i=1}^{m} \|x_i\|_2^2 \leq mR^2. \tag{43}$$

Therefore,

$$\mathfrak{R}_m(\mathcal{H}_B) \leq \frac{B}{m} \sqrt{mR^2} = \frac{RB}{\sqrt{m}}. \tag{44}$$

It remains to connect this complexity bound to generalization. By the standard Rademacher generalization theorem for a loss class with range at most C_ℓ, with probability at least $1 - \delta$, uniformly for all $h \in \mathcal{H}_B$,

$$\mathcal{L}(h) \leq \widehat{\mathcal{L}}(h) + 2\mathfrak{R}_m(\ell \circ \mathcal{H}_B) + 3C_\ell \sqrt{\frac{\log(2/\delta)}{2m}}. \tag{45}$$

Since ℓ is L_ℓ-Lipschitz in the prediction, the contraction inequality gives

$$\mathfrak{R}_m(\ell \circ \mathcal{H}_B) \leq L_\ell \mathfrak{R}_m(\mathcal{H}_B). \tag{46}$$

Substituting the Rademacher-complexity bound derived above yields

$$\mathcal{L}(\theta) \leq \widehat{\mathcal{L}}(\theta) + 2L_\ell \mathfrak{R}_m(\mathcal{H}_B) + 3C_\ell \sqrt{\frac{\log(2/\delta)}{2m}} \leq \widehat{\mathcal{L}}(\theta) + \frac{2L_\ell RB}{\sqrt{m}} + 3C_\ell \sqrt{\frac{\log(2/\delta)}{2m}}. \tag{47}$$

This completes the proof. □

G Broader Impacts

The paper discusses potential benefits for scalable, interpretable clinical prediction in low-data settings and emphasizes responsible use through uncertainty awareness, explanation stability, and clinician oversight.

H Declaration of LLM usage

The core method development in this research does not involve LLMs as any important, original, or non-standard components. LLMs were used **solely for writing refinement** and **not for** retrieval, discovery, or research ideation.