ProbMED: A Probabilistic Framework for Medical Multimodal Binding

Supplementary Material

A. Datasets

We pretrained ProbMED exclusively on MIMIC datasets, specifically MIMIC-CXR [26], MIMIC-ECG [19], and MIMIC-ECHO [18]. Additionally, we constructed the **MIMIC-CONNECT** subset (following [16]), which contains patients with both CXR and ECG data (details in §A.4). **We maintained the same patient-level differences in all MIMIC datasets to prevent data contamination.** Furthermore, MIMIC-CONNECT enabled the evaluation of a multimodal approach, utilizing CXR and ECG for prediction. The details are described below and Tab. 7. We split up the datasets used by their respective modality.

A.1. Chest X-ray datasets

MIMIC-CXR [26]. This dataset is used for both pretraining and evaluation. It consists of CXR with their paired radiology reports. We preprocessed each case's CXR and corresponding text using methods outlined in [61]. We restricted the dataset to only the Anterior-Posterior (AP and PA) views. For pretraining, we used a predefined training split of the dataset for CXR-text binding.

Kaggle COVID [7]. This public dataset comprises CXR images annotated with binary COVID-19 labels and was used solely for ZS and FS task. For ZS testing, we generated prompts as suggested in [61].

RSNA Pneumonia [48]. The RSNA Pneumonia dataset contains CXR images of pneumonia cases and is publicly available through the National Institutes of Health database. This dataset was also only used for ZS and FS task. For ZS classification, we used a prompt, `"Chest X-ray findings consistent with lung infection"` to explain pneumonia for ZS evaluation. We followed [16] for splitting the dataset into train and test set.

Montgomery [4, 25]. This consists of CXR images collected from Tuberculosis Control program in Montgomery County, Maryland. It includes annotations for tuberculosis and other thoracic abnormalities, providing a challenging evaluation subset. For consistency with our methodology, we used the same prompt for the RSNA Pneumonia dataset to explain tuberculosis for ZS evaluation.

OpenI [13]. This dataset comprises CXR images, corresponding radiology reports, and clinical findings extracted from the Indiana University hospital database. Among all CXR images, we took the CXRs with the AP view. We used this dataset for TEXT-to-CXR retrieval evaluation.

Chexpert5x200 [24]. Following the formulation in previous works [16, 22, 61], CheXpert5x200 is a multi-class classification subset derived from CheXpert-1.0 [24]. It consists of five distinct classes (i.e., atelectasis, cardiomegaly, consolidation, edema, and pleural effusion), with 200 images per class. We followed [22] for the test split. This dataset's experimental setup (TEXT-to-CXR retrieval) and evaluation metrics mirrored those applied to the other datasets.

CheXchoNet [2]. This open-public CXR dataset has unique pairs of CXR with gold-standard ECHO labels. We used a label of composite of severe left ventricular hypertrophy and dilated left ventricle, which are both significant findings that can be found on ECHO and also in CXR [2]. In our experiments, CheXchoNet was utilized primarily for evaluation and acted as an emergent alignment dataset (i.e., detecting unseen diseases during the evaluation). ZS and FS settings assessed the model's performance in identifying ECHO-based pathologies.

A.2. ECG datasets

MIMIC-ECG [19]. The dataset comprises 10-second, 12-lead ECG recordings originally sampled at 500 Hz. These signals were down-sampled to 100 Hz using a low-pass filter to reduce noise and computational overhead [9]. Each ECG is accompanied by machine-generated reports and links `cart_id` to free-form textual data. When available, the free-form text was used to generate corresponding ECG descriptions; otherwise, the machine reports were employed. This dataset played a dual role in our study, as it was used for training and evaluation.

PTB-XL [55]. This ECG dataset is a large-scale, publicly available repository of 12-lead ECG recordings and contains annotated ECGs with comprehensive diagnostic labels and expert assessments. For our experiments, we applied similar preprocessing steps as with MIMIC-ECG, including downsampling to 100 Hz using an appropriate low-pass filter. The dataset was used solely for evaluation: retrieval, ZS, and FS settings.

ICBEB [33]. This dataset provides 12-lead ECG recordings, annotated with diagnostic information. We applied consistent pre-processing, that is, low-pass filtering and downsampling to 100 Hz, to ensure compatibility with our training protocols. The ICBEB dataset was used exclusively for evaluation with ZS and FS experiments.

MUSIC [36]. This ECG dataset comprises vectorcardiograms (VCGs) originally sampled at 1000 Hz. Each VCG recording is paired with ECHO labels indicating key clinical parameters: SLVH, DLV, and LVEF. Although the ECHO modality is not provided, the associated labels offer valuable diagnostic insights. To use this dataset, we ap-

Dataset	Pretrain?	Modalities	Task	#Cls	#train	#valid	#test
MIMIC-CXR [26]	✓	CXR + TXT	Pretrain/Retrieval/**Multimodal-Classification**	12	86,853	12,059	24,799
OpenI [13]	✗	CXR + TXT	Retrieval	-	-	-	2,864
CheXpert5x200 [24]	✗	CXR + TXT	Retrieval	5	-	-	1,000
RSNA [48]	✗	CXR	Classification	2	18,678	-	5,338
COVID [7]	✗	CXR	Classification	2	11,028	-	2,780
Montgomery [4, 25]	✗	CXR	Classification	2	32	-	106
CheXchoNet [2]	✗	CXR*	Classification	2	64,619	3,303	3,667
MIMIC-ECG [19]	✓	ECG + TXT	Pretrain/Retrieval/**Multimodal-Classification**	-	88,291	12,065	24,644
PTB-XL [55]	✗	ECG + TXT	Retrieval/Classification	71	17,415	2,183	2,198
ICBEB [33]	✗	ECG	Classification	9	5,501	-	1,376
MUSIC [36]	✗	ECG*	Classification	2	512	-	125
MIMIC-ECHO [18]	✓	ECHO + TXT	Pretrain/Retrieval	-	13,732	3,880	1,957
EchoNet-Dynamic [42]	✗	ECHO	Classification	2	7,394	1,273	1,264
MIMIC-CONNECT [19, 26]	✓	ECG + CXR	**Multimodal-Classification**	-	22,397	3,292	6,664

Table 7. **ALL** datasets for CXR, ECG, and ECHO modalities. Pretrain column represents data used to train ProbMED. Modalities highlight the modality types. *These datasets contain corresponding modality and **ECHO**-based labels derived from the ECHO-report.

plied the Kors regression transformation [54] to convert the VCGs into 12-lead ECGs, which were subsequently downsampled to 100 Hz. MUSIC was used solely for evaluation in our emergent ZS and FS experiments. Like CheXchoNet, we used a label for a composite of severe left ventricular hypertrophy and dilated left ventricle.

A.3. ECHO datasets

MIMIC-ECHO [18]. This dataset consists of ECHOs from various patients in the MIMIC dataset cohort. We matched patients using `hadm_id` to connect data to discharge notes in MIMIC-IV [27]. In this study, we used **all ECHOs connected to a discharge note containing ECHO-related text**. These texts were first processed by Llama3.1-Instruct 8B [14], followed by manual verification by human experts. DICOMs for each ECHO were processed into individual ECHOs. All frames were used as augmentations during training, while only the first frame was used for evaluation.
EchoNet-Dynamic [42]. We used this ECHO dataset comprising apical-4-chamber ECHO videos and corresponding left ventricular ejection fraction (LVEF) labels. Continuous LVEF values were constructed to be binary with threshold LVEF<40%, based on [53]. Each video has been preprocessed to a standardized ($3 \times 112 \times 112$). A single frame corresponding to the end-systolic (ES) phase of the left ventricle was extracted from the ECHO video, as labeled in the EchoNet-Dynamic dataset. We extrapolated the frame into ($3 \times 224 \times 224$) resolution using cubic interpolation to match the resolution with MIMIC-ECHO and used this dataset for evaluation in ZS and FS.

A.4. MIMIC-CONNECT

MIMIC-CONNECT. We derived this dataset by linking MIMIC-CXR and MIMIC-ECG to MIMIC-IV [27]. We matched `subject_id` and `hadm_id` ensuring that the modality recording times were within 7-day window. For cases with available visit identifiers, `hadm_id`, we directly paired MIMIC-CXR and MIMIC-ECG to form this dataset that we refer to as MIMIC-CONNECT; when `hadm_id` was unavailable, the pairing was based solely on `subject_id` and a 7-day window between the CXR and ECG recordings. This dataset was used for training and evaluation–multimodal ZS and FS classification tasks.

A.5. Data Representation

We used conventional representations for each modality. CXR images, initially single-channel ($1 \times 224 \times 224$), are duplicated across channels to match the standard 3-channel (RGB) inputs, making them into the size of ($3 \times 224 \times 224$). ECG signals are treated as 12 distinct leads over time, producing a (12×1000) tensor for a 10-second recording sampled at 100 Hz. Processing ECHO videos follows the method in ECHO-CLIP [8], utilizing separate frames with ($3 \times 224 \times 224$) resolution instead of the entire video as an input. Finally, text data is tokenized with a BERT-based tokenizer [28], with sequences padded or truncated to 100 tokens to fit typical medical report lengths.

B. Loss Function Implementation

B.1. Hellinger Loss Calculation

The Hellinger equation, from [44], is calculated in this paper based on the following simplifications–aligned with the propositions in the paper. First, we bring the Eq. (6) from the main manuscript here (as a reference):

$$H^2(q_n, k_t) = 1 - \frac{\det(\boldsymbol{\Sigma}_n)^{\frac{1}{4}} \det(\boldsymbol{\Sigma}_t)^{\frac{1}{4}}}{\det\left(\frac{\boldsymbol{\Sigma}_n + \boldsymbol{\Sigma}_t}{2}\right)^{\frac{1}{2}}} \times \exp\left(\left(-\frac{1}{8}(\mu_n - \mu_t)^\top \left(\frac{\boldsymbol{\Sigma}_n + \boldsymbol{\Sigma}_t}{2}\right)^{-1} (\mu_n - \mu_t)\right)\right).$$

Assuming that the covariance matrices are diagonal, i.e., our base assumption in § 3.1. The determinants can be writ-

ten as:

$$\det(\mathbf{\Sigma}_n) = \prod_{o=1}^{D} \sigma_{n,o}^2 \quad \text{and} \quad \det(\mathbf{\Sigma}_t) = \prod_{o=1}^{D} \sigma_{t,o}^2. \tag{12}$$

From Eq. (12), we see that the product term in Eq. (6) can be expressed as:

$$\begin{aligned} \det(\mathbf{\Sigma}_n)^{\frac{1}{4}} \det(\mathbf{\Sigma}_t)^{\frac{1}{4}} &= \prod_{o=1}^{D} (\sigma_{n,o}^2)^{\frac{1}{4}} (\sigma_{t,o}^2)^{\frac{1}{4}} \\ &= \prod_{o=1}^{D} (\sigma_{n,o}\sigma_{t,o})^{\frac{1}{2}}, \end{aligned} \tag{13}$$

Next, the determinant of the average term is:

$$\det\left(\frac{\mathbf{\Sigma}_n + \mathbf{\Sigma}_t}{2}\right)^{\frac{1}{2}} = \prod_{o=1}^{D} \left(\frac{\sigma_{n,o}^2 + \sigma_{t,o}^2}{2}\right)^{\frac{1}{2}}. \tag{14}$$

Furthermore, because the matrices are diagonal, the quadratic form in the exponent simplifies to a sum over dimensions:

$$\begin{aligned} &(\mu_n - \mu_t)^\top \left(\frac{\mathbf{\Sigma}_n + \mathbf{\Sigma}_t}{2}\right)^{-1} (\mu_n - \mu_t) \\ &= \sum_{o=1}^{D} \frac{(\mu_{n,o} - \mu_{t,o})^2}{\frac{\sigma_{n,o}^2 + \sigma_{t,o}^2}{2}} \\ &= 2 \sum_{o=1}^{D} \frac{(\mu_{n,o} - \mu_{t,o})^2}{\sigma_{n,o}^2 + \sigma_{t,o}^2}. \end{aligned} \tag{15}$$

Thus, given all of this, and exp properties. We arrived at the formulation:

$$H^2(q_n, k_t) = 1 - \prod_{o=1}^{D} \left[\frac{(\sigma_{n,o}\sigma_{t,o})^{\frac{1}{2}}}{(\frac{\sigma_{n,o}^2 + \sigma_{t,o}^2}{2})^{\frac{1}{2}}} \exp\left(-\frac{(\mu_{n,o} - \mu_{t,o})^2}{4(\sigma_{n,o}^2 + \sigma_{t,o}^2)}\right) \right]. \tag{16}$$

This can be simplied to in our formulation in Eq. (7):

$$H^2(q_n, k_t) = 1 - \prod_{o=1}^{D} \left[\left(\frac{2\sigma_{n,o}\sigma_{t,o}}{\sigma_{n,o}^2 + \sigma_{t,o}^2}\right)^{\frac{1}{2}} \exp\left(-\frac{(\mu_{n,o} - \mu_{t,o})^2}{4(\sigma_{n,o}^2 + \sigma_{t,o}^2)}\right) \right].$$

The pseudocode for calculating the Hellinger loss involves similar computations but uses the **logsumexp** trick (Algorithm 1). Hellinger distance is bounded between 0 and 1, where zero is the **same distribution**, and one is **far apart**. To use this in losses, we instead use $1 - \sqrt{H^2}$, where H^2 is the squared Hellinger distance. This lets us view our loss like **cosine similarity**. The implementation and pseudo-code are provided below.

Algorithm 1 Hellinger Distance

procedure COMPUTEHELLINGER(q_n, k_t)
  ▷ Get mean and log-variance:
    $\mu_n, \log(\sigma_n^2) \leftarrow q_n$
    $\mu_t, \log(\sigma_t^2) \leftarrow k_t$
  ▷ Convert log-variance to variance:
    $\sigma_n^2 \leftarrow \exp(\log \sigma_n^2))$
    $\sigma_t^2 \leftarrow \exp(\log \sigma_t^2))$
  ▷ Compute hellinger terms, logsumexp trick:
    $\sigma_{\text{prod}} = \sqrt{\sigma_n^2 \sigma_t^2}$
    $\sigma_{\text{sum}}^2 = \sigma_n^2 + \sigma_t^2$
    $T_1 = \frac{1}{2} \times \log\left(\frac{2 \times \sigma_{\text{prod}}}{\sigma_{\text{sum}}^2}\right)$
    $T_2 = \frac{1}{4} \times \frac{(\mu_n - \mu_t)^2}{\sigma_{\text{sum}}^2}$
  ▷ Compute the sum of logs across D dims:
    $T = \sum(T_1 + T_2)$
  ▷ Convert back with exp:
    $p = \exp(T)$
  ▷ Compute squared Hellinger distance:
    $H^2 = 1 - p$
  return $\sqrt{H^2}$
end procedure

Algorithm 2 SIS Loss Computation

procedure COMPUTESISLOSS(μ, $\log \sigma^2$, $N_s = 2$)
  ▷ Compute the standard deviation w/ log-variance:
  $\sigma \leftarrow \exp\left(0.5 \times \log \sigma^2\right)$
  $S \leftarrow$ **empty list**
  for $l \leftarrow 1$ **to** N_s **do**
    ▷ Sample ϵ^l from standard normal distribution:
    $\epsilon^l \leftarrow \text{RandomNormal}(\text{shape}(\mu))$
    ▷ Reparameterization is used to obtain samples:
    $s \leftarrow \mu + \text{diag}(\sigma)\epsilon^l$
    Append s to S
  end for
  ▷ Compute the 2N InfoNCE loss between samples:
  $L \leftarrow \text{InfoNCE}(S[0], S[1])$
  return L
end procedure

B.2. Intra-modality Loss

The pseudocode in Algorithm 2 outlines our SIS loss computation for within-modality learning in three main steps. First, we calculate the standard deviation from the given $\log(\sigma^2)$ by taking the exponential of half the $\log(\sigma^2)$. Next, we sample a normal distribution's noise vector ϵ. By applying the reparameterization trick, scaling this noise by the computed standard deviation, and shifting it by the mean, we generate a sample from the desired multivariate normal distribution with diagonal covariance [29]. This process is

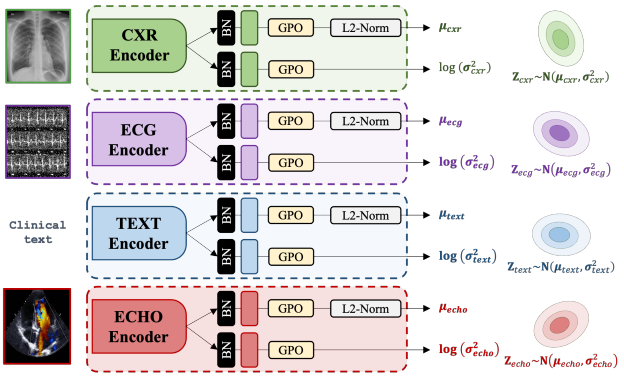

Figure 4. PROBMED model architecture. Encoders follow standard models. The proposed method of extracting the μ and $\log(\sigma^2)$ follows PCME++ [10]. BN represents BatchNorm1d and GPO is the Generalized Pooling Operator [5].

repeated to produce the required samples, N_s (here, $N_s = 2$ in our implementation–analagous to SimCLR [6]).

C. Model Architecture

PROBMED was built on PCME++ [10], which trains separate encoders for different data modalities (e.g., images and text) and represents each input as a normal distribution in a shared latent space. Specifically, each encoder outputs two D-dimensional vectors—one for μ and one for $\log(\sigma^2)$—that parameterize a Gaussian distribution. This setup allows the model to capture uncertainty and variability in the learned embeddings. Following PCME++, we found that using traditional encoders with two outputs effectively trained the probabilistic models. Extrapolated from PCME++, we produced μ and $\log(\sigma^2)$, using a duplicated final Transformer layer, i.e., the first branch is initialized with the same weights as the backbone (for μ). In contrast, the second branch (for $\log(\sigma^2)$) is initialized randomly. We adopt the GPO [5] for feature aggregation, improving training stability and performance.

Fig. 4 shows an overview of our model architecture. Our framework differs from the original PCME++ in two key ways. First, we introduce batch normalization layers following the encoders to normalize the input based on the mini-batch mean and variance. Second, we extend the architecture to simultaneously handle multiple medical modalities—such as CXR, ECG, text, and ECHO—by adding separate encoder branches for each modality. These multimodal embeddings are learned in a unified latent space, facilitating cross-modal alignment and downstream clinical tasks.

D. Pretraining details

In PROBMED, each modality is processed by a dedicated encoder chosen for its domain-specific strengths. Inspired by PCME++—highlighting the importance of modality-specific representations when transitioning to a probabilistic embedding space—we adopt state-of-the-art pre-trained models where they are most effective [10]. Our text data is encoded using BioBERT [31] to capture rich, domain-aware linguistic nuances, while the CXR modality benefits from the robust feature extraction of the Swin-tiny [34, 38] model. For ECHO, we employ ConvNeXt [35] CLIP, with ECHO-CLIP weights [8] to effectively model its complex visual patterns. In contrast, our ECG encoder is built on a streamlined ResNet1D architecture and trained from scratch, as our experiments did not reveal any advantages from pretraining for this modality. This modular design enables PROBMED to leverage the strengths of specialized encoders within a unified framework for cross-modal probabilistic learning.

When we pretrain PROBMED, we utilized data augmentations for the input modalities. For CXR, we applied the random cropping to 224 $\times$ 224 from 256 $\times$ 256, horizontal flipping, color jittering, and random affine transformations following [56]. For ECG data augmentation, we applied adding random Gaussian noise. For the ECHO data, we applied randomized color jittering, gray scaling, and adding random Gaussian noise.

We trained PROBMED with the following hyperparameters detailed in Tab. 8. The hyperparameters for **the final loss** (Eq. (11)) are $\alpha = 1.0$, $\beta = 0.5$, $\gamma = 0.0001$. We set the temperature scale with $\tau = 0.07$ for all related losses; we explore more τ parameters in §H. All our experiments (including pretraining) were conducted on a single 48GB L40S or a 40GB A100 GPU.

E. Evaluation Details

In this section, we expand on the results presented in the main paper.

E.1. Zero-Shot Prompts

We generated dataset-specific prompts per label with varying descriptions of medical findings. Especially for the emergent ZS tasks (e.g., predicting the labels observable in ECHO using CXR), we generated 10 different prompts for the positive label since the emergent ability of the model often lacks capturing the meaning of the disease with using a single prompt. For non-emergent datasets, we chose using a simple prompt for the ZS task for simplicity. To keep the prompts concise and clinically relevant, we utilized short sentences focusing on the presence or absence of disease. We observed that succinct prompts improved interpretability and maintained performance in our ZS evaluations. For generating 80 different prompts used in Tab. 5, we used GPT-4o to paraphrase the baseline prompt making the descriptions of all prompts consistent. We compute the cosine distance between the text embeddings for each la-

Config	TEXT	CXR	ECG	ECHO
Pretrained Models	BioBERT [31]	Swin-tiny [34]	XResNet-1d [21]	ECHO-CLIP [8]
Final output dim.	μ-output=512, $\log(\sigma^2)$-output=512			
Optimizer	AdamW			
Optimizer Momentum	$\beta_1 = 0.9, \beta_2 = 0.95$			
Learning Rate (LR)	1.00e-04			
LR Scheduler	CosineAnnealingLR			
Gradient clipping	1.0			
Weight Decay	1.00e-05			
Batch size	192			
Total Epoches	120			

Table 8. PROBMED hyperparameters.

bel (i.e., positive and negative label) and image embedding for ZS classification. For multiple prompts, we averaged the text embeddings for each label to generate a prototype, representing each label.

E.2. Few-Shot Evaluation

The FS results in Tab. 10, Tab. 11a, and Tab. 11b used a traditional linear probing set-up [46]. We sampled k training samples per class, where $k \in \{4, 16\}$. These were chosen to highlight the use cases of our model in FS learning. Extended FS results (i.e., $k \in \{2, 4, 8, 16\}$) are presented in the following subsection.

F. Extended Results

This section highlights the full results in many of the tables presented in the main text.

F.1. Multimodal classification using MIMIC

We showed that Top-K retrieval analysis identifies the most effective distance metric for retrieval in the main manuscript under Tab. 1a, Tab. 1b, and Tab. 1c. Deterministic methods leverage cosine similarity distance, whereas probabilistic methods employ the distance measure used during model training (e.g., Hellinger distance for PROBMED). While this approach yielded strong results, we also emphasized evaluating all models under a **consistent** distance metric. Specifically, we adopted **cosine similarity** as the standard measure, which, in the case of probabilistic models, involves computing distances using **only the μ embedding**, as initially proposed in [10]. The corresponding retrieval performance for various modality-text pairs is presented in Tab. 9a, Tab. 9b, Tab. 9c.

F.2. Few-Shot Classification

In Tab. 3a, Tab. 3b, Tab. 3c, we highlighted performance under a limited range of k-shot conditions to demonstrate our approach's ability to learn effectively from scarce labeled examples. Here, we present an expanded set of few-shot experiments (including 2-, 4-, 8-, and 16-shot scenarios) for completeness and transparency. These additional results, shown in Tab. 10, Tab. 11a, Tab. 11b, show how each model scales with varying amounts of labeled data in FS scenarioes.

(a) **Cosine-based TEXT-to-CXR retrieval**

	MIMIC-CXR		OpenI		Chexpert5x200		RSUM
	R@1	R@5	R@1	R@5	R@1	R@5	
MedCLIP [56]	1.0	4.3	0.6	2.8	2.6	3.0	14.3
CXR-CLIP [61]	**47.3**	70.4	12.7	25.2	8.5	23.0	187.1
BiomedCLIP [62]	36.2	59.9	9.0	19.9	6.4	19.8	151.2
CheXzero [50]	26.7	50.0	5.8	15.1	3.5	17.8	118.9
MEDBind [16]	40.8	67.5	11.6	25.5	7.9	21.4	174.7
BioVil-T [1]	28.4	58.2	8.1	18.9	4.9	17.1	135.6
SAT [32]	40.3	69.2	6.7	14.7	**9.1**	**26.7**	166.7
PCME++ [10]	4.5	21.9	9.5	20.6	1.3	4.6	62.4
PROBMED (Ours)	47.0	**70.8**	**13.2**	**28.1**	8.8	23.9	**191.8**

(b) **Cosine-based TEXT-to-ECG retrieval**

	MIMIC-ECG		PTB-XL		RSUM
	R@1	R@5	R@1	R@5	
ECG-CLIP [16]	40.8	76.7	2.3	9.8	129.6
MEDBind [16]	44.1	78.2	**3.1**	12.1	137.5
PCME++ [10]	7.1	14.1	1.5	11.2	33.9
PROBMED (Ours)	**51.3**	**86.9**	2.4	**12.7**	**153.3**

(c) **Cosine-based TEXT-to-ECHO retrieval**

	MIMIC-ECHO		RSUM
	R@1	R@5	
EchoCLIP [8]	1.1	6.4	7.5
PCME++ [10]	1.0	4.0	5.0
PROBMED (Ours)	**1.7**	**7.8**	**9.5**

Table 9. Cross-modal retrieval performance (Recall@K) for TEXT-to-CXR, TEXT-to-ECG, and TEXT-to-ECHO retrieval tasks, **the similarity metric for all models is cosine similarity**.

F.3. CXR and ECG Combination

Here, we provide the complete results for our CKD and CHD classification experiments, including additional performance metrics and comparisons across all evaluated methods. These extended results further validate the robustness of our findings, showing consistent gains from integrating CXR and ECG and demonstrating PROBMED's advantages over competing approaches. We also include detailed ablations and per-class breakdowns to highlight the nuanced benefits of our probabilistic modeling framework

	Kaggle COVID				RSNA Pneumonia				Montgomery				CheXchoNet ⋆			
	2S	4S	8S	16S	2S	4S	8S	16S	2S	4S	8S	16S	2S	4S	8S	16S
MedCLIP [56]	80.5	85.5	88.8	90.8	55.7	58.0	61.8	65.4	87.0	87.3	86.4	88.5	52.4	55.8	59.3	63.9
CXR-CLIP [61]	81.5	**86.7**	89.9	91.6	60.2	64.1	66.5	70.9	80.1	85.8	89.1	91.6	51.1	53.2	55.9	59.7
BiomedCLIP [62]	**82.8**	86.0	88.8	89.4	75.6	80.3	83.1	84.0	86.6	87.0	90.9	92.2	**58.8**	59.8	60.9	61.8
CheXzero [50]	82.2	82.8	84.7	88.4	75.3	75.0	78.1	82.7	85.8	88.5	90.4	92.9	52.5	60.3	60.8	66.1
MEDBind [16]	82.0	86.2	89.6	**92.0**	62.5	67.3	70.6	73.4	87.5	89.9	91.0	91.8	56.0	57.6	62.2	65.4
PCME++ [10]	81.8	79.6	81.1	85.9	72.6	73.2	77.0	79.2	76.2	75.7	80.6	81.8	51.1	56.8	61.8	62.7
ProbMED (Ours)	81.7	86.5	**90.4**	91.8	**77.5**	**82.2**	**84.0**	**84.7**	**91.2**	**93.1**	**93.2**	**93.7**	58.0	**63.3**	**65.3**	**68.5**

Table 10. **CXR-based few-shot extended results** (2, 4, 8, 16 shots denoted as #S). Model performance is reported as AUROC (%). ⋆ CheXchoNet is an *emergent* dataset using CXR-to-ECHO labels.

(a) **ECG-based few-shot classification.**

	PTB-XL				ICBEB				MUSIC ⋆			
	2S	4S	8S	16S	2S	4S	8S	16S	2S	4S	8S	16S
ECG-CLIP [16]	62.7	67.1	70.0	71.2	62.7	69.1	72.0	74.1	44.7	48.6	51.1	51.4
MEDBind [16]	65.1	71.1	75.9	81.8	76.2	81.2	84.5	87.8	50.0	51.5	52.5	54.6
ECG-FM [37]	64.6	69.1	70.9	71.6	65.6	69.3	71.0	71.8	47.0	50.0	51.0	53.1
PCME++ [10]	72.6	75.4	77.9	79.9	65.2	74.1	79.5	80.5	49.1	46.7	50.3	48.6
ProbMED (Ours)	**80.7**	**82.6**	**85.6**	**87.6**	**81.0**	**84.8**	**87.3**	**90.1**	**51.6**	**53.8**	**57.1**	**59.1**

(b) **ECHO-based few-shot results.**

	EchoNet-Dynamic			
	2S	4S	8S	16S
ECHO-CLIP [8]	**88.2**	**88.3**	93.7	95.0
PCME++ [10]	87.0	87.5	92.5	94.1
ProbMED (Ours)	86.7	87.7	**95.9**	**96.2**

Table 11. Comparison of ECG-based and ECHO-based few-shot classification results. AUROC (%) for 2-, 4-, 8-, 16-shot (#S).⋆ MUSIC is an *emergent* dataset using ECG-to-ECHO labels.

Method	**CKD**				
	ZS	2S	4S	8S	16S
CXR-only model performance					
MedCLIP [56]	61.8	56.7	59.6	61.6	62.3
CXR-CLIP [61]	73.5	56.5	59.4	61.5	62.0
MEDBind [16]	71.9	54.0	54.8	55.5	57.6
CheXzero [50]	73.6	68.7	66.9	69.4	70.4
BiomedCLIP [62]	65.6	61.1	66.5	68.4	71.2
PCME++ [10] (CXR)	51.0	55.9	68.6	69.8	68.8
ProbMED (CXR)	75.0	**69.8**	70.6	70.8	76.5
ECG-only model performance					
ECG-CLIP [16]	54.1	50.9	58.9	61.0	66.4
MEDBind [16]	61.2	54.9	66.4	66.9	67.8
ECG-FM [37]	-	48.9	55.2	59.7	62.1
PCME++ [10] (ECG)	31.7	55.8	66.7	69.0	70.3
ProbMED (ECG)	68.5	56.7	67.4	65.1	71.1
Using both ECG + CXR Models					
MEDBind [16]	71.5	54.3	68.4	69.8	69.8
PCME++ [10]	46.8	56.5	69.4	70.6	71.6
ProbMED	**78.1**	67.5	**71.5**	**73.4**	**76.8**

Table 12. **MIMIC-CKD Results**

Method	**CHD**				
	ZS	2S	4S	8S	16S
CXR-only model performance					
MedCLIP [56]	65.4	64.4	68.2	73.8	74.7
CXR-CLIP [61]	73.1	65.1	68.4	73.9	75.2
MEDBind [16]	76.6	56.7	63.4	65.1	69.1
CheXzero [50]	77.1	62.1	67.4	73.8	74.6
BiomedCLIP [62]	61.8	61.2	67.0	69.4	73.7
PCME++ [10] (CXR)	51.1	70.1	72.7	75.6	76.7
ProbMED (CXR)	77.0	71.0	71.9	77.7	79.8
ECG-only model performance					
ECG-CLIP [16]	65.7	60.5	71.1	75.0	74.1
MEDBind [16]	65.6	61.7	73.5	73.9	73.5
ECG-FM [37]	-	69.4	71.6	72.1	73.9
PCME++ [10] (ECG)	40.3	64.7	74.7	75.9	76.7
ProbMED (ECG)	70.3	64.4	72.2	73.8	76.7
Using both ECG + CXR Models					
MEDBind [16]	75.3	68.3	71.7	75.9	78.6
PCME++ [10]	52.4	**73.7**	**76.9**	77.7	78.7
ProbMED	**78.4**	72.4	73.0	**79.2**	**80.8**

Table 13. **MIMIC-CHD Results**

in zero-shot and few-shot scenarios.

G. Visualizations

G.1. Qualitative Image-Text Visualizations

Consider a patient's CXR showing signs of respiratory distress. Even though we can describe it with different phrases—e.g., `"CXR shows a cloudy patch in the lower lung"` or `"CXR has pneumonia"`—both statements reflect the same underlying finding. Fig. 5 illustrates this by plotting ProbMED embeddings for the pneumonia CXR and two text descriptions. While a purely deterministic approach (using only μ) does not reveal the full similarity structure, the probabilistic embeddings (incorporating μ and σ) cluster the image and text descriptions together, highlighting the importance of modeling uncertainty in medical image–text alignment.

G.2. Increasing Uncertainty with Noise

Fig. 6 shows how ProbMED responds to increasing levels of Gaussian noise injected into a CXR image. Specifically, we examine the average of the $\log(\sigma^2)$ vector across em-

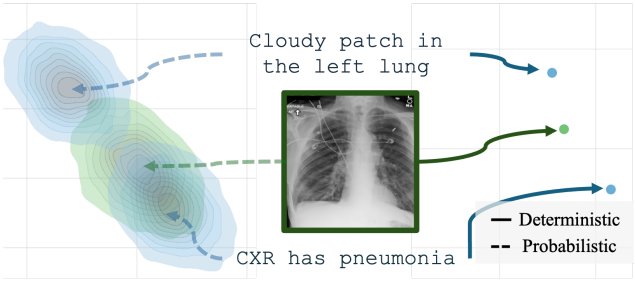

Figure 5. Qualitative visualization of PROBMED embeddings using PCA for dimensionality reduction. We plot a **CXR embedding** (depicting pneumonia, a CXR sampled from RSNA Pneumonia dataset) alongside two distinct **TXT embeddings** of "Cloudy patch in the left lung" and "CXR has pneumonia". In the diagram, probabilistic embeddings provide the distributions of each embedding, while the deterministic embeddings provide the limited interpretation of the ambiguity. Best viewed in color.

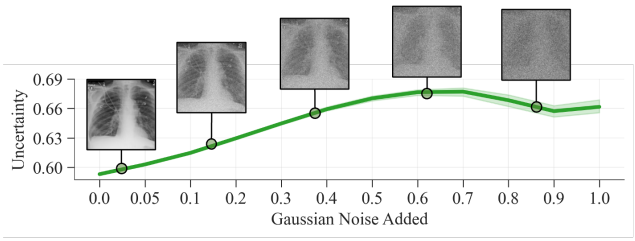

Figure 6. Qualitative visualization of PROBMED embeddings of a CXR with increasing Gaussian noise added. **Uncertainty** was defined and represented as the $\frac{1}{2}\exp(\cdot)$ average of the $\log(\sigma^2)$ vector across the dimension axis.

bedding dimensions—interpreted as the model's estimated uncertainty. As noise increases, $\log(\sigma^2)$ also increases, indicating that PROBMED becomes more "uncertain" when the input is corrupted. This behavior makes intuitive sense as we imagine that a more pixelated CXR may encapsulate many possibilities. Ultimately, PROBMED's ability to capture variability in its latent space as it dynamically adjusts the mean and variance of its probabilistic embeddings in response to noisy inputs. Note, empirically, similar observations were seen with ECG and ECHO; we present the CXR for simplicity of visualization.

H. Additional Ablations

H.1. Effect of Temperature on Loss

We also varied the temperature hyperparameter within the different contrastive losses used in the study. We tried different τ parameters for consistency but kept them the same throughout the model. This was due to the associated computational cost. Herein, we show its impact on multimodal alignment. As shown in Tab. 14, smaller fixed temperatures

τ	MIMIC-CXR		MIMIC-ECG		MIMIC-ECHO		RSUM
	R@1	R@5	R@1	R@5	R@1	R@5	
Trainable	21.8	59.0	21.3	53.2	**2.4**	**11.4**	169.1
0.05	**49.3**	**72.0**	46.1	86.2	2.2	6.8	262.6
0.07	47.9	71.4	**48.3**	**87.0**	**2.4**	7.8	**264.8**
0.2	47.6	71.3	48.2	84.9	2.3	7.6	261.9
1	47.3	70.9	48.2	83.6	**2.4**	8.0	260.4

Table 14. τ examination for PROBMED contrastive losses.

(e.g., 0.05) increase retrieval performance on MIMIC-CXR and MIMIC-ECG relative to a trainable temperature, while a moderate temperature (0.07) achieves the highest overall RSUM. These results suggest appropriately tuning the temperature can significantly influence alignment effectiveness in contrastive learning.

We aim to explore this phenomenon more in future studies, particularly by optimizing each temperature. However, this was not feasible at this time due to computational costs with hyperparameter tuning.

I. Ethical Considerations

PROBMED employs a probabilistic joint embedding framework to integrate multiple medical modalities and capture clinically relevant associations. However, it is designed to uncover meaningful relationships within heterogeneous medical data. It is essential to rigorously evaluate the embeddings and their potential implications for clinical decision-making. Our framework builds upon embeddings derived from various sources, including curated clinical datasets and publicly accessible medical repositories. We advocate for continuously scrutinizing probabilistic embedding methods in medical contexts to identify and mitigate unintended associations.