Semi-supervised Concept Bottleneck Models

Lijie Hu*, Tianhao Huang*, Huanyi Xie, Xilin Gong
Chenyang Ren, Zhengyu Hu, Lu Yu, Ping Ma, and Di Wang†
Corresponding to di.wang@kaust.edu.sa

PRADA Lab

ICCV Oct 19-23, 2025 Honolulu Hawaii

Introduction

Concept Bottleneck Models provide human-interpretable explanations but rely heavily on costly expert annotations and often suffer from misalignment between concept predictions and input features. To address these challenges, we propose the SSCBM, which leverages both labeled and unlabeled data. Our framework introduces pseudo-concept labels via a KNN-based method and incorporates an alignment loss to reduce misalignment between concept saliency maps and input saliency maps. Experiments across multiple datasets demonstrate that SSCBM achieves high concept accuracy and strong interpretability, even with as little as 10% labeled data, closing the gap with fully supervised baselines.

Preliminaries

Concept Bottleneck Models We consider a classification task with a concept set $\mathcal{C} = \{p_1, \ldots, p_k\}$ and dataset $\mathcal{D} = \{(x^{(i)}, y^{(i)}, c^{(i)})\}_{i=1}^{N}$, where $x^{(i)} \in \mathbb{R}^d$ is the input, $y^{(i)} \in \mathbb{R}^l$ is the label, and $c^{(i)} \in \{0,1\}^k$ is the concept vector. A CBM learns (i) a *concept encoder* $g : \mathbb{R}^d \to \mathbb{R}^k$ and (ii) a *label predictor* $f : \mathbb{R}^k \to \mathbb{R}^l$, usually linear. For input x, the model predicts concepts $\hat{c} = g(x)$ and label $\hat{y} = f(g(x))$.

Concept Embedding Models For input x, a latent representation $\boldsymbol{h} = \psi(x)$ is used to generate embeddings for each concept c_i:

$$\hat{\boldsymbol{c}}_i = \phi_i(\boldsymbol{h}) = a(W_i \boldsymbol{h} + \boldsymbol{b}_i), \quad i = 1, \ldots, k. \tag{1}$$

Each concept is represented by embeddings $\hat{\boldsymbol{c}}_i^+$ (True) and $\hat{\boldsymbol{c}}_i^-$ (False). A scoring function predicts activation probability

$$\hat{p}_i = \sigma\big(W_s[\hat{\boldsymbol{c}}_i^+, \hat{\boldsymbol{c}}_i^-]^\top + \boldsymbol{b}_s\big), \tag{2}$$

and the final embedding is a convex combination

$$\hat{\boldsymbol{c}}_i = \hat{p}_i \hat{\boldsymbol{c}}_i^+ + (1 - \hat{p}_i)\hat{\boldsymbol{c}}_i^-. \tag{3}$$

Semi-Supervised Setting In practice, only a small portion of concept labels is available. We split the feature set into $\mathcal{X} = \{\mathcal{X}_L, \mathcal{X}_U\}$, where $\mathcal{X}_L$ has both class and concept labels, while $\mathcal{X}_U$ only has class labels. The training dataset is $\mathcal{D} = \mathcal{D}_L \cup \mathcal{D}_U$. The goal is to train $g : \mathbb{R}^d \to \mathbb{R}^k$ and $f : \mathbb{R}^k \to \mathbb{R}^l$ using both labeled and unlabeled data, improving task accuracy while maintaining concept-based interpretability.

Semi-supervised Concept Bottleneck Models

Labeled Data. Given labeled samples $\mathcal{D}_L = \{(x^{(i)}, y^{(i)}, c^{(i)})\}_{i=1}^{|\mathcal{D}_L|}$, a backbone network ψ extracts features $\boldsymbol{h} = \psi(x)$, which are then passed through an embedding generator ϕ to obtain concept embeddings:

$$\hat{\boldsymbol{c}}_i^{(j)} = \sigma\Big(\phi(\boldsymbol{h}^{(j)})\Big), \quad i = 1, \ldots, k,\ j = 1, \ldots, |\mathcal{D}_L|,$$

where σ is the activation function. The predicted concept vector $\hat{\boldsymbol{c}}$ is compared with ground-truth concepts $\boldsymbol{c}$ via binary cross-entropy: $\mathcal{L}_c = BCE(\hat{\boldsymbol{c}}, \boldsymbol{c})$. Label predictor maps $\hat{\boldsymbol{c}}$ to class labels $\hat{\boldsymbol{y}}$, and we optimize with categorical cross-entropy: $\mathcal{L}_{task} = CE(\hat{\boldsymbol{y}}, \boldsymbol{y})$.

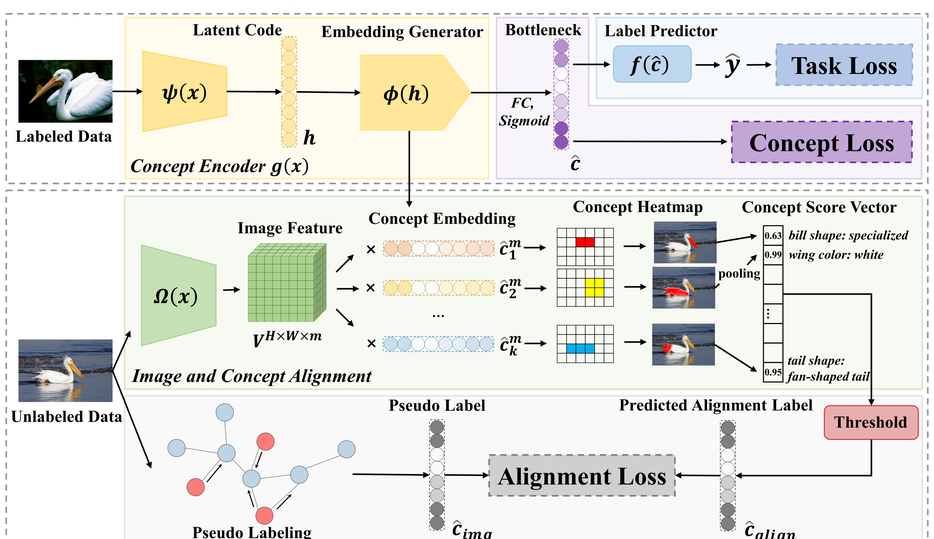
Unlabeled Data. For unlabeled samples $\mathcal{D}_U = \{(x^{(i)}, y^{(i)})\}_{i=1}^{|\mathcal{D}_U|}$, we first generate pseudo concept labels $\hat{\boldsymbol{c}}_{img}$ using a k-nearest neighbor (KNN) search: $\mathrm{dist}(x, x^{(j)}) = 1 - \frac{\Omega(x)\cdot\Omega(x^{(j)})}{\|\Omega(x)\|_2 \cdot \|\Omega(x^{(j)})\|_2}$, where Ω is a visual encoder. Concept labels of nearest neighbors are aggregated into $\hat{\boldsymbol{c}}_{img}$.

To capture feature–concept relations, we also compute heatmaps between image features $V = \Omega(x) \in \mathbb{R}^{H \times W \times m}$ and concept embeddings $\hat{\boldsymbol{c}}_i$: $\mathcal{H}_{p,q,i} = \frac{\hat{\boldsymbol{c}}_i^m \cdot V_{p,q}}{\|\hat{\boldsymbol{c}}_i^m\| \cdot \|V_{p,q}\|}, \quad p = 1, \ldots, H,\ q = 1, \ldots, W.$ Averaging over p, q yields similarity scores s_i, which are thresholded to form alignment labels $\hat{\boldsymbol{c}}_{align}$. We enforce consistency via alignment loss:$\mathcal{L}_{align} = BCE(\hat{\boldsymbol{c}}_{img}, \hat{\boldsymbol{c}}_{align})$.

Final Objective. The full optimization objective combines task accuracy, concept prediction, and alignment:

$$\mathcal{L} = \mathcal{L}_{task} + \lambda_1 \cdot \mathcal{L}_c + \lambda_2 \cdot \mathcal{L}_{align},$$

where λ_1, λ_2 control the trade-off between interpretability and accuracy.

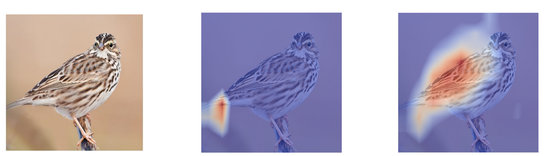


Evaluation Results

Method	CUB		AwA2		WBCatt		7-point	
	Concept	Task	Concept	Task	Concept	Task	Concept	Task
CBM	93.99%	67.33%	96.48%	88.71%	94.18%	99.71%	74.34%	75.44%
CEM	96.39%	79.82%	95.91%	87.00%	95.33%	99.71%	77.15%	75.85%
SSCBM	90.88%	67.67%	96.48%	89.77%	93.98%	99.68%	73.67%	70.09%

Interpretability Evaluation

We evaluate alignment by checking whether concept saliency maps match ground-truth semantics. Results confirm that SSCBM produces faithful and interpretable concept maps, demonstrating the effectiveness of alignment loss.



Original Image | Tail Pattern: Multi-colored | Wing Color: Brown

Test-time Intervention

We correct 10%–100% of concept labels during inference and observe steady accuracy gains, showing SSCBM learns faithful concept–label relations. Using COOP to adjust key uncertain concepts, intervention shifts predictions (e.g., from *Swainson Warbler* to *Great Crested Flycatcher*), improving both interpretability and performance.