APSIPA Transactions on Signal and Information Processing, 2025, 14, e301

Original Paper

ViP-CBM: A Low-parameter and Interpretable Concept Bottleneck Model Using Visual-projected Embeddings

Ji Qi*, Huisheng Wang and H. Vicky Zhao

Department of Automation, Tsinghua University, Beijing, China

ABSTRACT

With the increasing application of deep neural networks (DNN) in personal and property security-related scenarios, ensuring the interpretability and trustworthiness of DNN models is crucial. Concept Bottleneck Models (CBMs) improve interoperability by predicting human-understandable concepts in the hidden layer for the final task, but they face challenges in efficiency and interpretability in multi-label classification (MLC) of concepts, such as ignoring concept correlations or relying on complex models with limited performance gain. To address the challenge of massive parameters and limited interpretability in the concept MLC problem, we propose a novel **Visual-Projecting CBM (ViP-CBM)**, which reformulates the MLC of concepts as an input-dependent binary classification problem of concept embeddings using visual features for projection. Our ViP-CBM model reduces the training parameter set by more than 50% compared to other *embedding-based* CBMs while achieving comparable or even better performance in concept and class prediction. Our ViP-CBM also provides a more intuitive explanation by visualizing the projected embedding space. Additionally, we propose an intervention method for our ViP-CBM,

*Corresponding author: qij21@mails.tsinghua.edu.cn

Received 13 March 2025; revised 03 June 2025; accepted 09 June 2025
ISSN 2048-7703; DOI 10.1561/116.20250015

which is shown to be more efficient than other embedding-based CBMs under joint training by experiments.

Keywords: Concept bottleneck models, multi-label classification, visual projection

1 Introduction

As deep neural networks (DNNs) are increasingly used in scenarios concerning personal and property security, such as healthcare, automatic driving, and financial services, the trustworthiness and interpretability of DNNs have been increasingly significant, bringing explainable AI (XAI) to the forefront.

Recent advances in XAI have increasingly focused on concept-based approaches, which aim to enhance the interpretability of DNNs by leveraging human-understandable concepts. These researches follow two routines: post-hoc explanation techniques and human-interpretable modeling strategies. Post-hoc methods, such as LIME [33] and SHAP [25], focus on interpreting already-trained black-box models by identifying the most crucial features of the input data that support the model's decision. Network Dissection [2] went further in post-hoc methods, attempting to interpret image classification by examining features in all intermediate layers of convolutional neural networks (CNNs). Testing with Concept Activation Vectors (TCAV) [16] generates activation vectors for concepts of interest using a Support Vector Machine (SVM) to classify the feature vectors in the hidden layer of a pretrained DNN and provide concept-based explanations by quantifying the sensitivity of model predictions to the concepts according to the CAVs. However, there are fundamental limitations in fully explaining end-to-end DNNs through post-hoc explanations alone [34]. These methods cannot align these features extracted by end-to-end trained black-box models perfectly with human cognition [16], leaving a gap in achieving comprehensive interpretability.

In contrast, human-interpretable modeling seeks to construct DNNs with inherent interpretable inference processes based on concepts. These models are trained to provide supportive information in the intermediate layers with additional supervision. Concept-based models have emerged as a popular approach within this category, providing explanations of the model's decisions through high-level concepts. Concept Whitening [6] performs affine transformations in the latent space to align axes with concepts of interest. Pan and Zhang [29] propose assigning specific concepts to different layers of a CNN, progressing from low-level features (*e.g.*, colors, textures, *etc*) to high-level semantics (*e.g.*, objects). While these methods provide layer-wise concept explanations, they often require massive annotation, incur significant training

costs, and suffer from scalability issues due to quadratic parameter growth with concept numbers [51].

In this work we focus on a pivot concept-based framework interpretable image classification, the Concept Bottleneck Model (CBM) [19]. CBMs decouple the inference process into two phases: (1) predicting human-understandable concepts from the input images and (2) utilizing these concepts exclusively for final class prediction. CBM is a simple and useful interpretable deep model since it enables humans to understand the decisions of the model with concept predictions and allows test-time *intervention* to improve accuracy in downstream tasks by correcting false concept predictions.

However, conventional CBMs face critical limitations when scaling to complex multi-concept scenarios. As the bottleneck of CBM, concept prediction in conventional CBMs typically employs independent binary classifiers for each concept and neglects the correlations in concept semantics, leading to suboptimal accuracy and compromised interpretability when dealing with large concept sets. To address this issue, recent researches employ sophisticated architectures to capture inter-concept dependencies. Havasi *et al.* [9] propose an autoregressive architecture inspired by the classifier chain [32] in multi-label classification (MLC) to capture correlations among concepts, which improves concept accuracy and task accuracy significantly. Xu *et al.* [46] employs an energy-based probabilistic graph to learn the relevance of concepts through inferences with gradient descent. However, the above methods involve more parameters and more complex model structures and require extensive calculations in gradients and samplings but achieve only limited performance gain.

To improve the efficiency and interpretability of the MLC problem in concept prediction, we propose **Visual-Projecting Concept Bottleneck Models (ViP-CBM)**. Inspired by [47], our work uniquely employs visual features extracted from input images as projecting matrices on the concept embedding space. The projected embedding vectors of the concepts are binary classified as activated or not by a unified linear classifier, as is shown in Figure 1. Through the visual-projecting (ViP) module in ViP-CBM, we convert the MLC problem in concept prediction into an input-dependent binary classification problem, which intuitively explains the classification with the projection mechanism and attains better interpretability. Our ViP-CBM reduces the parameters in concept prediction to that of a scalar-CBM where the hidden layers directly represent the concept's activation probability.

The contributions of our work are as follows:

- We propose an interpretable ViP module for image multi-label classification to convert MLC problems into binary classification problems in the embedding space of labels.
- We propose ViP-CBM, which reduces the number of training parameters to that of the minimal scalar CBM, which is less than 50% of that of

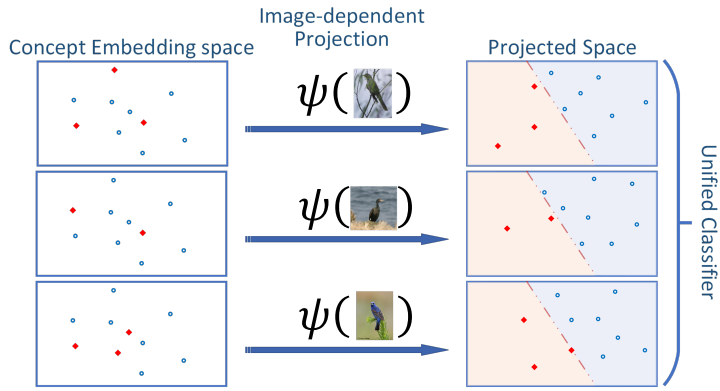


Figure 1: Our ViP-CBM converts multi-label classification of concepts to unified binary classification of concept embeddings projected by visual features of input images. $\psi(\cdot)$ denotes the visual feature extractor.

other embedding-based CBMs. Our ViP-CBM achieves similar performance in both concept prediction and class prediction when compared to other CBMs.

- Experiments on interpretability and intervention show that compared to other embedding-based CBMs, our ViP-CBM can be better explained by visualizing the projected concept-embedding space, and is more efficient for intervention under joint training.

2 Related Works

In this section, we first briefly introduce the three paradigms in research in Explainable Artificial Intelligence (XAI). Then we review related works in Concept Bottleneck Models (CBMs) and visual-Semantic embedding that inspire our work.

2.1 Explainable Artificial Intelligence (XAI)

Research on XAI can be categorized into three main paradigms: concept-based, model-based, and causal-and-reasoning-based. Concept-based paradigm focuses on defining and evaluating interpretability from a human cognition perspective by linking model outputs or intermediate variables to human-understandable concepts, providing explanations in decision rules [52], feature importance [33, 25] and hidden layer semantics [16, 19, 6, 29]. Model-based

paradigm focuses on designing models with inherent interpretability, emphasizing transparency in the structures and functions of its components, including creating sparse connections in neurons [44, 11], encouraging disentanglement upon inputs [12], and inventing modules based on conventional signal processing approaches [27]. However, this type of research brings difficulty in building models and may struggle to achieve performance comparable to black-box models in complex tasks. Causal-Reasoning-based paradigm focuses on the causal relationships between variables and the reasoning mechanisms. Following the pioneer thoughts by Pearl and Mackenzie [31], causal inference has been developed to analyze and learn causal effects between variables through causal graphs [35]. Other efforts in revealing the reasoning process lie in prototype learning [3, 22] which extracts typical instances to explain the model's decision, the self-explaining method [1, 20, 21, 39] which produces explanations simultaneously with predictions. With the development of large language models (LLMs), prompt learning can also be applied to causal inference [53, 23] besides providing explanations for reasoning. These methods provide more profound and essential explanations but suffer from high complexity and computational costs.

2.2 Concept Bottleneck Models

Concept Bottleneck Models (CBMs) [19] consist of two parts: the *concept predictor* and the *class predictor*. The concept predictor generates human-specified concepts from input images, while the class predictor uses these concepts for final classification. Earlier studies in CBMs directly employ each variable in the concept learning layer as probabilities for each concept's existence, which we refer to as scalar CBMs in the rest of this paper. Assuming that the incompleteness of the concept set prevents CBMs from achieving higher task accuracy, Havasi *et al.* [9] introduced side channels to represent undiscovered binary concepts to enhance model performance and mitigate information leakage. Subsequent works like Coop-CBM [38] add a side branch before the concept prediction layer for immediate task prediction to improve accuracy in concept prediction. Post hoc CBM [49] suggests a new framework to convert any pretrained black-box models into CBMs while maintaining task accuracies by predicting concepts by projecting extracted features on Concept Activation Vectors trained from other supporting datasets by SVM or multimodal models. Building upon this model, Label-free CBM [28] further employs GPT-3 [4] for concept annotation to eliminate the need for densely annotated data. However, these models fail to provide exact predictions of concepts.

Recent studies have focused on improving CBM efficiency and flexibility while preserving interpretability. Editable CBM [14] addresses scalability by three levels of data removal, including labeling level, concept level, and data

level, to eliminate the need for full retraining. Similarly, Incremental Residual CBM [37] tackles concept incompletement by complementing missing concepts with a set of optimizable vectors and incrementally discovering new ones from the candidate concept bank. To address annotation scarcity, Semi-supervised CBM [13] leverage pseudo-labeling, jointly train on both labeled and unlabeled data, and then align the unlabeled data at the concept level. These approaches reduce reliance on costly annotations and retraining overhead and align with CBM's lightweight interpretability goals.

To improve the expressivity of concepts in the model, embeddings are introduced in the concept-bottleneck frameworks. Concept Embedding Model (CEM) [50] learns a pair of positive and negative embeddings for each concept from the input to extend feature representations to higher dimensions. ProbCBM [17] and Energy-based CBM [46] use individually trained concept embeddings for concept prediction through their relationship with features extracted from the original input, the former using Euclidean distance in space and the latter constructing probabilistic graphs via Boltzmann energy models. Stochastic CBM [41] explicitly models concept correlations and allows single interventions to propagate corrections across related concepts, which improves effectiveness in CLIP-inferred settings. Additionally, Coarse-to-Fine CBM [30] introduces the notion of concept hierarchy to uncover and exploit more granular concept information in patch-specific regions of the image scene, which outperforms classical CBM architectures. However, the above methods introduce more parameters or even other large models and additional data to attain higher concept and task accuracy, which makes model structures and training complicated increases training costs, and deviates from the original intent of achieving interpretability on basic small models by feature supervision.

2.3 Visual-semantic Embeddings Models

Visual-semantic embedding is a powerful paradigm for image-text matching problems, which bridges vision and language modalities by mapping visual features into an embedding space of labels based on their textual semantics. The pioneering work Deep Visual Semantic Embedding (DeViSE) [8] is typically proposed for image classification with extreme labels and outperforms conventional models that treat labels as mutually independent entities by exploiting correlations in label semantics. The DeVisE model leverages a pre-trained word2vec [26] model to embed words into vectors with semantic information preserved such as synonymy and use a pre-trained convolutional neural network (CNN) to extract feature vectors with the same dimensionality of the label-text embedding space from input images. In this paradigm, classifying an input image is to assign the most relevant label based on the similarities between the image and labels, which is measured by a generalized dot product of the visual features and concept embeddings in the embedding space

with a trainable metric matrix. The DeViSE framework is also extended to sentence-level problem such as image description generation in [18, 15], and are generalized for zero- and few-shot learning due to the continuity of visual space and the use of unannotated text in [40, 5].

For MLC of images with extreme labels, which is most relevant to the challenges in CBM research, Yeh and Li [47] adapt DeViSE's scoring mechanism and reformulate the compatibility between images and labels as the norms of the projected label embedding vectors with the extracted visual features working as the projecting matrices. This framework suggests a classification rule that the visual features always project embeddings of the positive labels closer to $\mathbf{0}$, and is trained with Triplet Loss [36]. However, this model can only predict the k most possible labels from the highest k matching scores and does not provide predictions on the entire set of labels.

In summary, all the above visual-semantic embedding models depend on extensive textual corpora for learning label embeddings from semantic relations and syntactic components. Furthermore, since these semantic embeddings are generated by the pretrained language model, they are fixed for the final class predicting task. Thus, the performance of the model relies entirely on the training of the visual part, while the concept part is not involved in improving performance.

There are also other attempts to include trainable concept representations when bridging visual-textual information in the concept-bottleneck paradigm. For example, BotCL [43] learns a visual-semantic concept bottleneck in SENN [1] with nonsemantic embeddings of implicit concepts where concept embeddings are learnable during training. However, due to self-supervised implicit concepts in SENN that are not understandable to humans, BotCL lacks credibility compared to CBM and does not support intervention. In comparison, our ViP-CBM represents concepts as trainable vectors without restrictions on semantics and assigns definite labels to each concept for the class predicting step, which addresses the above weaknesses.

3 The Proposed ViP-CBM Framework

3.1 Notations

As a variant of CBM, our ViP-CBM requires a fully supervised dataset denoted as $\mathcal{D} = \left\{\mathbf{x}^{(i)}, \mathbf{c}^{(i)}, y^{(i)}\right\}_{j=1}^{N}$ with N data points, K binary concepts and M classes, where the i-th data point consists of the input $\mathbf{x}^{(i)} \in \mathcal{X}$, the concepts $\mathbf{c}^{(j)} \in \{0,1\}^K$ and the label $y^{(j)} \in \{1,\ldots,M\}$.

3.2 Motivation

Traditional approaches to MLC in concept prediction in CBMs [19, 50, 17] treat concept classification as isolated binary tasks, neglecting the correlations between concepts and resulting in shortcomings in interpretability. Introducing embeddings in CBM yields limited performance benefits while significantly increasing the number of parameters. To address this challenge, we propose a visual projecting (ViP) method to reduce the number of parameters and enhance interpretability while leveraging the rich semantics of embeddings.

Our key thoughts and innovation lie in rethinking MLC of concepts in CBMs as a *bipartition problem* in the concept embedding space, where a partition surface separates the embeddings of positive and negative concepts for each input image. To unify this input-dependent partition problem, we suppose an input-dependent projection operator $\psi_{\mathbf{x}}(\cdot)$ (where $\mathbf{x}$ denotes the input image) on the concept embedding space $\mathbb{R}^d$ that project the partition surface into a unified hyperplane S for every image $\mathbf{x}$ in the projected space $\mathbb{R}^m$, and projected embeddings of positive and negative concept are also separated by the hyperplane S. In conclusion, we propose converting the learning of a multi-label concept classifier in CBMs into training a linear binary classifier on the input-conditioned projected embeddings for each concept.

We construct the projection operator utilizing the powerful fitting capacity of CNN by employing visual features extracted from images by the same CNN backbone as projecting matrices for original concept embeddings, which simplifies our model structure to be similar to scalar CBMs [19], and ensure low-parameter properties in our model. Our design extends beyond the multi-label DeViSE method [47] by implementing clear classification distinctions and removing the dependency on word embedding models and backup corpus for concept embedding generation, which address the problems discussed in Section 2.3. We refer to this method as the visual-projecting (ViP) method, which will be detailed in the following section.

3.3 Model Structure

Figure 2 shows the general prediction flow of our ViP-CBM. Our ViP-CBM includes concept embeddings the visual projecting (ViP) module, the concept predictor, and the task predictor, which we will introduce one by one.

3.3.1 Concept Embeddings and the ViP Module

To exploit the rich semantics and ensure the effective utilization during model training of concept embeddings, we represent all K concepts $\{c_1, \ldots, c_K\}$ as trainable vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_K\} \subset \mathbb{R}^d$. Given an input image $\mathbf{x}$, a backbone CNN $\phi(\cdot)$ extracts visual features as a matrix $\mathbf{Z} = \phi(\mathbf{x}) \in \mathbb{R}^{m \times d}$. We introduce

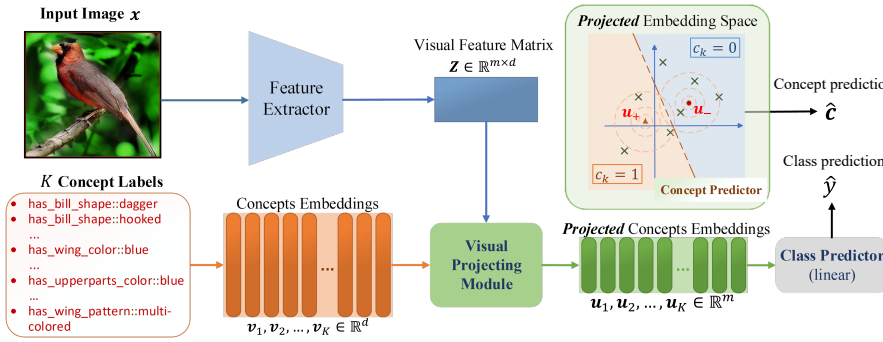


Figure 2: Model structure of our ViP-CBM.

a nonlinear projection $\mathbf{u}_k = \theta(\mathbf{Z}\mathbf{v}_k) \in \mathbb{R}^m, k = 1, \ldots, K$, by adding a *non-parametric* nonlinear function $\theta(\cdot)$ on a simple linear projection. In this work, we propose the nonlinear projection function as

$$u_{k,j} = ReLU\left(\mathbf{z}_j^\top \mathbf{v}_k + \frac{\mathbf{z}_j^\top}{\|\mathbf{z}_{j,\cdot}\|_2}\mathbf{v}_k\right), j = 1, \ldots, m, \tag{1}$$

where $\mathbf{z}_j \in \mathbb{R}^d, j = 1, \ldots, m$ denotes the j-th row of the matrix $\mathbf{Z}$ and $u_{k,j}$ denotes the j-th element of $\mathbf{u}_k$. This projection function uses ReLU as the activation function and introduces "normalized linear projections", which is the latter term of the inputs of the ReLU function, to increase the nonlinearity.

3.3.2 *Concept Predictor*

As is discussed in Section 3.2, our ViP-CBM proposes to convert MLC in concept prediction into a unified binary classification in projected concept embeddings $\mathbf{u}_1, \ldots, \mathbf{u}_K$. We propose a linear classifier with a pair of anchor points for the concept predictor. Define a pair of unified and trainable anchor points $\mathbf{u}_+, \mathbf{u}_- \in \mathbb{R}^m$, each concept c_k is classified as activated and inactivated according to the Euclidean distance from its corresponding projected embedding $\mathbf{u}_k$ to the two anchors: projections of positive concepts are closer to $\mathbf{u}_+$ and negative concepts are closer to $\mathbf{u}_-$. Such classification rules add nonlinearity to a conventional linear model with the computation of the Euclidean distances. Inspired by Triplet Loss in [36], the probability of the concept c_k being activated, *i.e.*, the prediction of the real concept $\hat{c}_k = 1$, is

$$p(\hat{c}_k = 1 | \mathbf{Z}, \{\mathbf{v}_i\}_{i=1}^K) = \sigma\left(a\left(\|\mathbf{u}_k - \mathbf{u}_-\|_2 - \|\mathbf{u}_k - \mathbf{u}_+\|_2 - m_{c_k}\right)\right), \tag{2}$$

where $\sigma(\cdot)$ represents the sigmoid function, $a > 0$ is a learnable scaling parameter, and $m_{c_k} \geq 0$ is an optional decision margin depending on true label

c_k. For example, we can set m_{c_k} to a positive constant m to penalize false positives. To encourage the model to project both activated and inactivated concepts closer to the corresponding anchors for inputs from each class, we set the margins as

$$m_{c_k} = \mathbf{1}_{(c_k=1)}m + \mathbf{1}_{(c_k=0)}(-m), m > 0, \tag{3}$$

where $\mathbf{1}_{(\cdot)}$ is the instructional function. We refer to this setting as the *symmetric margins*, which will be further discussed through experiments in Section 5.1.

3.3.3 Class Predictor

To minimize the number of model parameters, we employ a simple linear classifier for predicting the M final classes. Different from conventional CBMs, we use the K projected concept embeddings $[\mathbf{u}_1, \ldots, \mathbf{u}_K]$ as inputs of the class predictor instead of concept predictions $p(\hat{c}_k = 1|\mathbf{Z}, \{\mathbf{v}_i\}_{i=1}^K)$, to improve model's performance and smooth the training process by leveraging the richer information in embeddings.

3.3.4 Training Strategy and Loss Function

Since we calculate concept and task probabilities with the model, we apply Binary Cross-Entropy (BCE) loss to concept prediction and Cross-Entropy (CE) loss to class label prediction. In this work, we employ the *joint* CBM training strategy, which is to train both concepts and labels simultaneously by minimizing a weighted sum of the two losses:

$$\mathcal{L} = \mathbb{E}_{(\mathbf{x},\mathbf{c},y)}\left[\mathcal{L}_{CE}(y, \hat{y}) + \alpha\mathcal{L}_{BCE}(\mathbf{c}, \hat{\mathbf{c}})\right]. \tag{4}$$

3.4 Parameter Reduction in ViP-CBM

Our ViP-CBM reduces parameters mainly in concept prediction. Consider a *minimal* CBM with the same backbone CNN as our ViP-CBM, in which binary concepts are predicted from extracted visual features $\mathbf{Z} \in \mathbb{R}^{m\times d}$ by a simple linear model:

$$p(\hat{c}_k = 1|\mathbf{Z}) = \sigma\left(\mathbf{w}^\top \bar{\mathbf{z}} + b\right), \tag{5}$$

where $\bar{\mathbf{z}} \in \mathbb{R}^{md}$ represents the flattened vector of matrix $\mathbf{Z}$, and the parameters of the linear model are $(\mathbf{w}, b) \in \mathbb{R}^{Kmd} \times \mathbb{R}$. In comparison, the parameters in the concept predictor of our ViP-CBM are $(\{\mathbf{v}_i\}_{i=1}^K, \mathbf{u}_+, \mathbf{u}_-) \in \mathbb{R}^{K\times d} \times \mathbb{R}^m \times \mathbb{R}^m$. Therefore, the number of training parameters in the concept predictor in our ViP-CBM is approximately m times less than that in the minimal scalar

CBM, and $m \times d'$ times less than embedding-based CBMs where d' represents the dimension of embeddings.

We now dig further into the mechanism of our ViP module for concept prediction. The concept predictor in (2) without margins is equivalent to a linear classifier in the final decision. With a linear classifier of the form

$$p(c_k = 1|\mathbf{u}_k) = \sigma\left(\tilde{\mathbf{w}}^\top \mathbf{u}_k + \tilde{b}\right), \tilde{\mathbf{w}} \in \mathbb{R}^m, \tilde{b} \in \mathbb{R}, \tag{6}$$

we can rewrite the concept predicting step as follows, neglecting the nonlinearity in projection:

$$p\left(\hat{c}_k = 1|\mathbf{Z}, \{\mathbf{v}_i\}_{i=1}^K\right) = \sigma(\tilde{\mathbf{w}}^\top(\mathbf{Z}\mathbf{v}_k) + \tilde{b}) = \sigma\left(\mathrm{tr}[(\mathbf{v}_k\tilde{\mathbf{w}}^\top)\mathbf{Z}] + \tilde{b}\right), \tag{7}$$

which is similar to (5) with the weight matrix $\mathbf{v}_k\tilde{\mathbf{w}}^\top$ of rank 1. This reveals that our model reduces the concept prediction module to a rank-1 classifier while using the two-anchor mechanism to encourage the separation of the positive and negative samples, and the nonlinear function $\theta(\cdot)$ to preserve the performance.

Despite the parameter reduction in the concept layer, our ViP-CBM uses projected embeddings for class predicting instead of probabilities, which makes the number of parameters of the class predictor in our ViP-CBM m times larger than that in the minimal CBM that predicts classes from scalars. These increases and reductions in parameters collectively lead to the total number of training parameters being comparable to the minimal *scalar* CBM. Nevertheless, due to the parameter reduction in concept predicting, our ViP-CBM has only less than half as many training parameters as the minimal CEM and ProbCBM. We compare the number of training parameters with examples in experiments in Table 1.

Table 1: Comparison of the number of training parameters for the CUB dataset.

Model	scalar-CBM	CEM	ProbCBM	ViP-CBM (ours)
Training Params #	254296	744681	746185	289625

3.5 Intervention

CBMs are trustworthy models since they predict concepts before performing downstream tasks so that we can perform *intervention* on the concept layer to correct misclassified concepts to correct the downstream class prediction results. Compared to CEM and ProbCBM, our ViP-CBM classifies all the concepts by measuring the distance between the projected concept embeddings to 2 unified and fixed anchors.

We propose an intervention method for our ViP-CBM as follows: for a mistakenly predicted concept c_k, we replace its corresponding projected embedding $\mathbf{u}_k$ with the anchor point $\mathbf{u}_+$ or $\mathbf{u}_-$ according to the correct label. Since our ViP-CBM is trained jointly, we keep the projected embeddings of other correctly predicted concepts to ensure that maximum label-irrelevant semantics contained in concept embeddings are maintained for the subsequent class predictor to attain better performance after intervention.

4 Experimental Setup

In this section, we introduce the datasets, the detailed setup for our ViP-CBM and the baseline models, and the metrics for evaluation for our experiments.

4.1 Datasets

CUB-200-2011
The Caltech-UCSD Birds-200-2011 Dataset (CUB-200-2011) [42] contains 11,788 images of 200 subcategories belonging to birds annotated with 312 binary concepts. We use the preprocessed dataset in [19] where the number of concepts is reduced to $K = 112$ and concepts are denoised to class-level, which means images from the same class share the same concept annotations. Concept labels in CUB-200-2011 are of the form "`{general_concept}::{detail}`" so that we can naturally group concepts into 28 groups based on the general concepts, with the largest group having 6 concepts.

AwA2
Animals with Attributes 2 Dataset (AwA2) [45] contains 37,322 images of 50 categories of animals with 85 binary attributes, *e.g.*, color, stripe, *etc.* AwA2 provides a category-attribute matrix that contains concept labels for each category so that concepts are also class-level. We artificially summarize these 85 concepts into 30 groups of color, pattern, habit, *etc.*, with the largest group having 14 concepts.

CelebA
CelebFaces Attributes Dataset (CelebA) [24] contains 202,599 celebrities' face images, each with 40 binary attribute annotations of facial features, *e.g.*, beard, hair color, *etc.* Following Zarlenga *et al.* [50], we select the 8 most balanced binary attributes to generate $2^8 = 256$ artificial classes for downstream class prediction and select the 6 most balanced attributes as concepts to mimic the circumstances where the concept bottleneck is narrow and inadequate to cover all the classes.

4.2 Experimental Setup

4.2.1 Data Preprocessing

We apply data augmentation on CUB and AwA2 datasets to artificially increase the diversity of the training data to improve the model's generalizability. We first perform color jittering and random horizontal flipping on the images, then resize them to 256×256. We randomly crop the images with a scale of $(0.8, 1.0)$ and resize images from CUB to 224×224 and images from AwA2 to 256×256. For the CelebA dataset, we downsample each human face image to 64×64 and normalize each entry of the pixels with a mean value of 0 and a standard error of 0.5. Since the original CelebA dataset is large and the experiments do not intend to obtain high performance for real applications, we randomly select a subset of 1/12 of the entire dataset for each run in our experiment.

4.2.2 ViP-CBM Settings

The structure of the visual feature extractor of our ViP-CBM is shown in Figure 3. We use a ResNet34 [10] pre-trained on ImageNet-1k [7] as the backbone and extract outputs of the layer before the global average pooling with a size of $(512, l, l)$ where l denotes the side length of the feature map. We then use a 1×1 convolution layer with d channels and flatten the outputs to get a feature representation of size (d, l^2), where d is the dimension of concept embeddings. Then we use a Fully Connected (FC) layer to reduce the l^2-entry inputs to m entries to get the feature matrix $\mathbf{Z} \in \mathbb{R}^{m \times d}$. The settings of the concept and class predictor follow the descriptions in Section 3.3. We set symmetric margins following (3) with $m = 0.1$ to study the effects of margins.

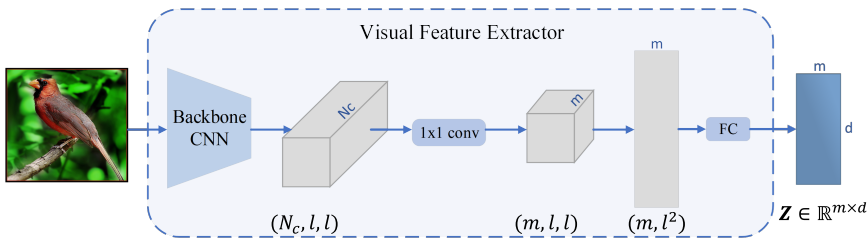


Figure 3: Detailed structure of the visual feature extractor in ViP-CBM.

4.2.3 Baselines

We compare our ViP-CBM with typical and conventional CBMs such as joint scalar CBM [19], CEM [50], and ProbCBM [17] with equivalent parameters. For all baseline models, we use the same pre-trained ResNet34 and 1×1 d-channel convolution layer to get a feature representation of size (d, 7, 7) and flatten it to a $49d$ dimensional vector. For baseline CBM, we use a 2-layer MLP for the concept predictor with a hidden layer size of 128 and apply ReLU as the activation function. We use a linear model to predict class labels directly from concept predictions as the class predictor.

For baseline CEM, we use a simple linear model to predict the positive and negative d-dimensional embeddings of each concept to align the dimensions with our model. We use a shared scoring function and linear class predictors as in [50].

For baseline ProbCBM, we use a linear model to generate K visual embeddings of d dimensions from the original features of size (d, l, l), and learn K pairs of positive and negative anchors for concept prediction in the embedding space of d dimensions. Since our only concern is the model's performance, we omit the sampling step and refine the class predictor to a simple linear model where the inputs are the K visual embeddings for this work.

4.2.4 Hyperparameters Settings

We set the weight between the two losses $\alpha = 5$ for the CUB and AwA2 datasets, and $\alpha = 1$ for the CelebA dataset. We use an SGD optimizer with a learning rate of 0.01 for the CUB dataset, 0.002 for the AwA2 dataset, and 0.005 for the CelebA dataset, all with momentum of 0.9 and weight decay of 5×10^{-4}. We train 400 epochs on the training split for the CUB dataset and the CelebA dataset, and 250 epochs for the training split of the AwA2 dataset.

4.3 Metrics

We use class accuracy as the criterion of the model's task performance. Denote the k-th concept prediction of the i-th sample as $c_k^{(i)} \in \{0, 1\}$. For the MLC of concepts, we define two metrics as follows.

- To evaluate the model's accuracy on each concept individually, we use the **Hamming score (HS)**:

$$HS = \frac{1}{NC} \sum_{i=1}^{N} \sum_{k=1}^{K} \mathbf{1}_{(\hat{c}_k^{(i)} = c_k^{(i)})}. \tag{8}$$

- To evaluate the model's ability to predict *all* concepts correctly, we use the **exact match ratio (EMR)**:

$$EMR = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}_{(\hat{\mathbf{c}}^{(i)}=\mathbf{c}^{(i)})}. \tag{9}$$

We also introduce **group EMR** to evaluate the model's performance on each concept group. In summary, we use the Hamming score to measure the individual concept accuracy, EMR for all concepts to represent the overall concept accuracy, and the minimum group EMR to evaluate concept prediction in the hardest group.

5 Experimental Results

5.1 Model Performance

We set the original concept embedding dimensions $d = 32$ and the dimension of the projected space $m = 12$ for the experiments. We add additional ablation studies of margins and nonlinearity, denoting ViP-CBM with symmetric margins as (3) as "+margin" and ViP-CBM using linear projection $\mathbf{u}_k = \mathbf{Z}\mathbf{v}_k$ in ViP module as "LP". As is discussed in Section 3.4, the concept predictor of the "LP" version of ViP-CBM is equivalent to a rank-1 linear classifier, and thus we directly employ a linear layer as (6) to substitute the concept predictor for simplicity. We conduct experiments with 5 different random seeds on each dataset to compute the average scores and standard errors, marked as "mean ± std" in our results. All models are trained on an entire NVIDIA GeForce RTX 2080Ti.

Table 1 shows that our model has only 40% of the training parameters of other embedding-based CBM, which is comparable to the scalar-CBM, with experiments in the CUB dataset as an example. For all models trained with a batch size of 128, the GPU memory usage is 5,826MB and the training time per epoch is approximately 15 seconds. Thus, our model does not increase the computation cost.

Table 2 shows the performance in concept and class prediction of our ViP-CBM and other baseline models. For each metric, we mark the highest score in **bold**, the second highest in purple, and the third highest in green. Note that we directly use the output of the backbone CNN of size (d, l, l) as the visual features for the baseline models as described in Section 4.2, which is larger than the visual features used in our ViP-CBM, suggesting that we are comparing with larger CBMs than we proposed in Section 3.4.

For the experiments on the AwA2 dataset with few concept and class labels and a large amount of data, our ViP-CBM model ranks third in individual

Table 2: Performance of our ViP-CBM in comparison with other baseline CBMs for CUB, AwA2, and CelebA datasets. The *starred* results indicate training instability, with more than half of the experiments failing by gradient explosion.

Model	CUB (112 concepts, 200 classes)			
	Hamming Score	Overall EMR	Min Group EMR	Class Accuracy
scalar-CBM	**0.9529**±0.0004	**0.4270**±0.0069	**0.7819**±0.0027	0.7197±0.0033
CEM	0.9506±0.0003	0.3887±0.0078	0.7651±0.0026	0.7186±0.0048
ProbCBM	0.9517±0.0010	0.4032±0.0132	0.7743±0.0058	**0.7268**±0.0113
ViP-CBM (ours)	0.9496±0.0009	0.3784±0.0223	0.7646±0.0037	0.7169±0.0048
+margin	0.9500±0.0013	0.3898±0.0090	0.7670±0.0053	0.7115±0.0075
LP	0.9398±0.0014*	0.3647±0.0148*	0.7248±0.0337*	0.6462±0.0463*

Model	AwA2 (85 concepts, 50 classes)			
	Hamming Score	overall EMR	Min Group EMR	Class Accuracy
scalar-CBM	**0.9708**±0.0006	0.7816±0.0040	0.8288±0.0020	0.8782±0.0046
CEM	0.9696±0.0007	0.7646±0.0035	0.8288±0.0020	0.8794±0.0028
ProbCBM	0.9704±0.0008	0.7798±0.0084	0.8366±0.0049	**0.8822**±0.0026
ViP-CBM (ours)	0.9702±0.0010	0.7857±0.0055	0.8403±0.0047	0.8818±0.0031
+margin	0.9701±0.0009	**0.7906**±0.0071	**0.8431**±0.0047	0.8807±0.0037
LP	0.9692±0.0006	0.7741±0.0037	0.8335±0.0050	0.8801±0.0024

Model	CelebA (6 concepts, 128 classes)			
	Hamming Score	overall EMR	Min Concept Acc	Class Accuracy
scalar-CBM	0.8789±0.0018	0.4882±0.0103	0.7598±0.0080	0.3609 ±0.0084
CEM	0.8828±0.0021	0.4952±0.0096	0.7612±0.0062	0.3660±0.0077
ProbCBM	**0.8872**±0.0028	**0.5100**±0.0142	0.7687±0.0092	0.3774±0.0055
ViP-CBM (ours)	0.8830± 0.0013	0.4918 ± 0.0067	0.7730±0.0032	0.3733±0.0081
+margin	0.8851±0.0018	0.4961±0.0038	0.7697±0.0050	**0.3796**±0.0124
LP	0.8834±0.0011*	0.4987±0.0043*	**0.7771**±0.0052*	0.3728±0.0010*

concept accuracy and second in all other metrics, and outperforms CEM by over 0.02 in concept overall accuracy and over 0.002 in class accuracy. The "+margin" version of ViP-CBM ranks highest in overall concept accuracy and group overall concept accuracy.

For the experiments on the CelebA dataset with very few concepts and inadequate concept bottleneck, our ViP-CBM and its variants rank in the top 3 on every metric among all models. Note that the "LP" version tends to get better results than the original ViP-CBMs due to the very small number of concepts since embeddings of 6 concepts in a $\mathbb{R}^{12}$ space are easier for linear separation.

For the experiments on the CUB dataset with a larger concept set and a smaller amount of data, due to the reduction of parameters and rank in concept prediction, our ViP-CBM model underperforms CBM significantly in overall concept accuracy, but is comparable to CBM in individual concept accuracy and class accuracy with a loss of less than 0.003. Our ViP-CBM achieves comparable or slightly superior performance to CEM with low embedding dimensions and slightly inferior performance to ProbCBM by less than 0.015 in overall concept accuracy and class accuracy. Compared to CEM

which learns $2K$ embeddings in total for each input image and ProbCBM which requires $2K$ concept anchors in total with K individually extracted visual features of d dimensions to embed in each concept space, our ViP-CBM learns only K concepts embeddings independent to inputs and the dimension of projected space m is much smaller than K. Thus, our ViP-CBM improves the efficiency of concept learning with concept representations of the same dimensions.

The ablation study for margins shows that symmetric margins in ViP-CBM enhance the overall accuracy for concepts and the stability for different initializations, consistent with our intent to penalize projected embeddings close to the classifying surface. In the ablation study of nonlinearity in visual projection, we mark a "*" on some of the results of the "LP" model to indicate that more than half of the experiment failed due to gradient explosion in training, and the metrics are computed from fewer experiments. These results reveal that nonlinearity in visual projection also allows higher learning rates, which increase the convergence speed and stability in training. Besides, the results of the successful experiments in the CUB and AwA2 datasets also prove that nonlinearity is necessary to bridge the performance gap due to the reduction of parameter numbers.

5.2 Sensitivity to Dimensions of the Representation Spaces

To reveal the impact of the choices of the anchors and representations, we experiment with our ViP-CBM under different settings in the dimension of the projected space $m = 6, 12, 24$ and original concept dimension $d = 16, 32, 64$. Table 3 shows all metrics for the CUB dataset, where we underline the highest scores for the same m and mark the globally highest scores in **boldface**. The results show that larger m and d lead to overfitting, while smaller m and d lead to underfitting. Our parameter selection in Section 5.1 is empirically optimal.

Table 3: Performance of our ViP-CBM with different m and d for the CUB dataset.

m	d	Hamming Score	Overall EMR	Min Group EMR	Class Accuracy
6	16	0.9475	0.3690	0.7604	0.7106
	32	0.9376	**0.3963**	0.7109	0.6348
	64	0.9491	0.3678	0.7568	0.7123
12	16	0.9470	0.3583	0.7546	0.6999
	32	**0.9505**	0.3819	**0.7724**	**0.7176**
	64	0.9463	0.3495	0.7503	0.7057
24	16	0.9448	0.3478	0.7377	0.6947
	32	0.9500	0.3638	0.7711	0.7156
	64	0.9177	0.2075	0.6253	0.5373

5.3 Interpretability

To study the interpretability of our ViP module proposed for MLC on a certain label set, we look into the spatial distributions of the visual features and the projected concept embeddings. With the hypothesis of the continuity of visual feature space [48], the visual features and the projected embeddings should cluster by class regardless of concept labels. For AwA2 dataset with class-level concepts, we select 3 concepts *"black", "white" and "blue"* and 2 classes *"cow"* and *"dophin"*, where *"black"* and *"white"* are positive for *"cow"* and *"white"* and *"blue"* are positive for *"dophin"*.

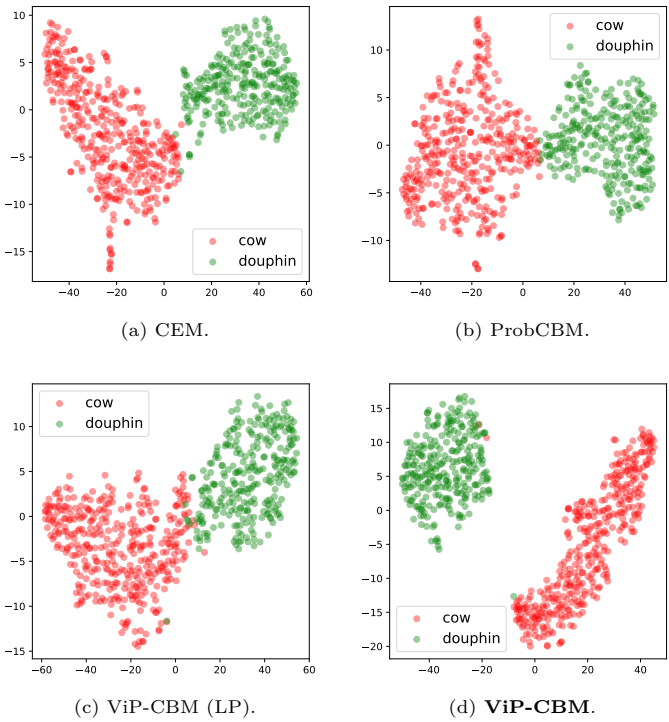
5.3.1 Explaining Class Prediction via Latent Feature Visualization

We first visualize the spatial distributions of the extracted visual feature representations of images from both 2 classes in the test split of the AwA2 dataset for all embedding-based CBMs using a 2-dimensional t-SNE plot in Figure 4. Visual feature representation in our ViP-CBM is defined as $\mathbf{Z} \in \mathbb{R}^{m \times d}$ in Section 3.3, and is defined as the combination of all concept representations extracted from the input images in CEM and ProbCBM. As is shown in Figure 4, all models generate two clear clusters of visual feature representations for the 2 class labels. Figure 4d further shows that visual feature clusters in our ViP-CBM with nonlinear projection are the most compact with a significant gap between the two classes.

5.3.2 Explaining Concept Prediction via Concept Representations Visualization

To compare the interpretability in concept prediction, we visualize the representations of the 3 selected concepts using t-SNE in Figure 5. The concept representations in our ViP-CBM are the projected concept embeddings $\mathbf{u} = \theta(\mathbf{Z}\mathbf{v}) \in \mathbb{R}^m$, while in baseline models are d-dimensional vectors directly extracted from the input images. Comparing Figure 5c and Figure 5d, the projected embeddings of inactivate concepts (blue and grey dots) and activated concepts are clearly separated on both figures, but ViP-CBM with nonlinear projection can form clear and compact clusters in the projected space. Focusing on Figure 5d, projected embeddings of each concept for inputs in the same class (represented by points with labels in each column of the legend) are clearly separated from each other, indicating the separation in original concept embeddings $\mathbf{v}$. Clusters of common activated concept *"white"* (green and purple dots) overlap significantly, indicating that the ViP module gathers the same concepts in the projected space for all input images containing the concept.

Next, we compare the explainability in the concept embedding space of our ViP-CBM with CEM and ProbCBM. In Figure 5a, embeddings for each

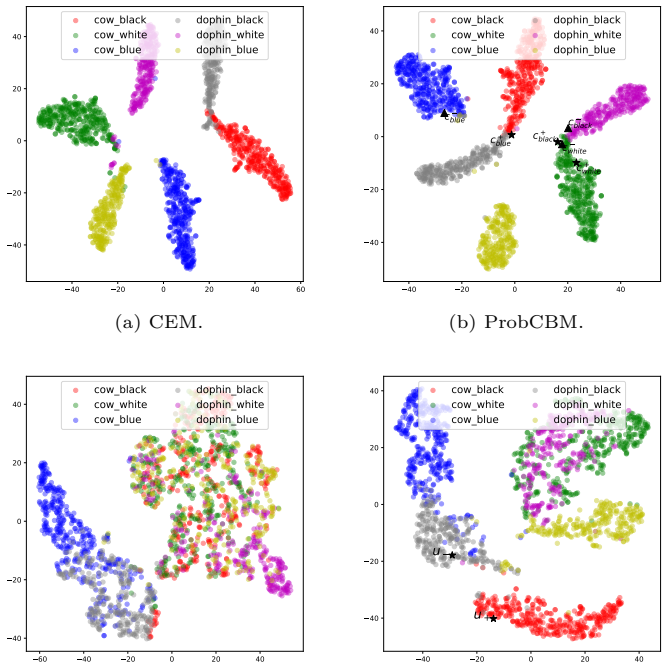


(a) CEM. (b) ProbCBM.

(c) ViP-CBM (LP). (d) **ViP-CBM**.

Figure 4: t-SNE plot of the original visual features $\mathbf{Z}$ in vector space for CEM, ProbCBM and our ViP-CBM.

concept in CEM from the same class also form separated clusters, but there is no explainable spatial relationship between concepts of the same activation state (concept *"white"* in this case). For ProbCBM, we mark the positive and negative anchor points $\mathbf{c}_k^+, \mathbf{c}_k^-$ for each concept with stars and triangles in Figure 5b. As is shown in Figure 5b, the concept representations of the commonly activated concept *"white"* from the two classes *"cow"* and *"dolphin"* are separately close to c_{white}^+ and c_{white}^+, which is contrary to the design of ProbCBM. Also, the distances from the representations of concept *"black"* of the two classes to the two anchors are significantly further than the distance between two anchors, and the postive and negative anchors for the commonly activated concept *"white"* are close to each other, indicating that the concept predictor of ProbCBM doesn't work exactly as designed under joint training. In summary, the concept representations of our ViP-CBM can be better explained via visualization after dimension reduction compared with CEM and ProbCBM.



(a) CEM. (b) ProbCBM.

(c) ViP-CBM (LP). (d) **ViP-CBM**.

Figure 5: 2D t-SNE plot of the concept representations in the embedding space for CEM, ProbCBM, and our ViP-CBM.

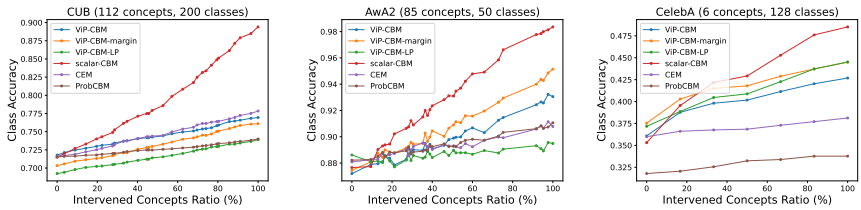
5.4 Intervention

In this section, we compare the intervention efficiency of our ViP-CBM and other baseline models. Following the paradigm in Koh *et al.* [19], we correct the entire concept group during each intervention step and perform intervention group by group until all concepts are corrected. We apply the original intervention method on scalar-CBM, CEM, and ProbCBM as described in their original works [19, 50, 17]. For the "LP" version of our ViP-CBM, since we use a simple linear layer for concept prediction as is described in Section 5.1, we change our intervention method to modify the projected embedding $\mathbf{u}_k$ of each predicted concept c_k to its *symmetric point* of the classification surface in (6):

$$\mathbf{u}'_k = \mathbf{u}_k - 2\frac{\tilde{\mathbf{w}}^\top \mathbf{u}_k + \tilde{b}}{\|\tilde{\mathbf{w}}\|_2^2}\tilde{\mathbf{w}}. \tag{10}$$

Note that all models in our experiments are *jointly* trained, meaning that the task predictor may not attain correct results with all correct concept inputs, since concept predicting loss works as an auxiliary loss in (4). Thus, the class accuracy with full intervention cannot be guaranteed to reach 100%.

Figure 6 shows the increasing curves of downstream classification accuracy w.r.t. the intervened concept ratio of different models in different datasets.



(a) CUB. (b) AwA2. (c) CelebA.

Figure 6: Effects of intervention for different models in CUB, AwA2, and CelebA.

For all three datasets, the efficiency of intervention of scalar-CBM is higher than all embedding-based CBMs. Since intervention in scalar-CBM directly modifies the concept probabilities, which are the inputs of its task predictor, intervening concept prediction in scalar-CBMs is more likely to correct class labels.

Figures 6b and 6c show that our ViP-CBM's efficiency of intervention is higher than other embedding-based CBMs. Compared to other embedding-based CBMs, our ViP-CBM introduces a uniform binary classification of concepts in the projected space, and projects embeddings of the same concepts of the same activate state to the same region in space as is shown in Figure 5d in Section 5.3. Thus, our intervention method on the projected space benefits from the interpretability in the latent space and is more efficient.

In Figure 6a, the efficiency of intervention of our ViP-CBM is close to CEM and still higher than ProbCBM, but all embedding models are significantly inferior to scalar-CBM in the efficiency of intervention for the CUB dataset. For complex tasks with a large number of concepts and classes, more label-irrelevant concept semantics are fed to the class predictor because of the higher dimension of its input due to larger K. Therefore, correcting concept embeddings leads to more changes in the input of the class predictor, resulting in worse intervention performance compared to scalar CBMs.

For all three datasets, ProbCBM has the lowest efficiency of intervention under joint training. ProbCBM provides positive and negative anchors for each concept in the embedding space, and each visual embedding extracted for each concept contains much more semantics besides label information. The class predictor of ProbCBM is trained with the noisiest inputs, and thus

the model has very little response to intervention. In summary, our ViP-CBM provides a more efficient intervention method than the selected typical embedding-based CBMs while keeping the same simplicity in implementation.

5.5 Robustness and Trustworthiness and Generalization

In this section, we discuss the robustness, trustworthiness, and generalization of our ViP-CBM detailedly through experiments with sparse, noisy, or incomplete concept supervision.

5.5.1 Robustness under Noisy or Missing Concepts

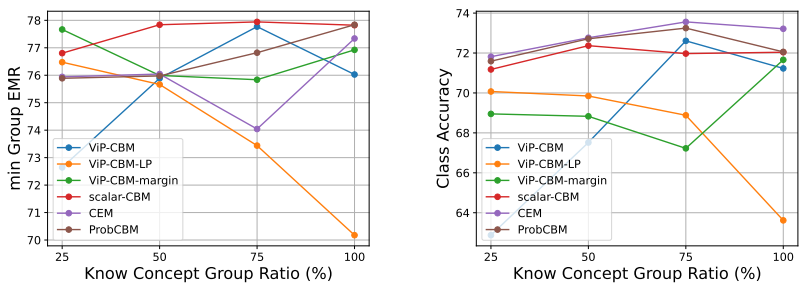
In real-world applications, concept annotations are often imprecise, containing random errors or even missing information. For instance, in the CUB dataset, some bird images may provide ambiguous or incomplete information about certain conceptsfor example, an image showing only the head of a bird might offer no data about its back color, while light and shadows could confound color recognition. In addition to the data augmentation done in Section 4.2.1, we perform an experiment on the CUB dataset with the original annotations. To address noisy or missing concepts, we incorporate the 4-degree *certainty* annotations via non-expert crowdsourcing from Wah *et al.* [42] by weighting the concept prediction loss with the normalized certainty scores, and compare the performance against class-level concept annotations in Table 4, where the metrics are colored and bolded following the description in Section 5.1. Experimental results show that our ViP-CBM-margin is still comparable to baseline models as analyzed in Section 5.1. All models experienced a slight degradation in performance on the concept EMR metric. However, the magnitude of these decreases is comparable across models, and our ViP-CBM-margin model exhibits the smallest drop. This suggests that the robustness of our model is at least comparable to that of the original model.

Table 4: Model performance of our ViP-CBM in comparison to baseline models in the CUB dataset with uncertain concepts. The up-arrows and down-arrows denote the differences in performance with all known concepts.

Model	Hamming Score	Overall EMR	Min Group EMR	Class Accuracy
scalar-CBM	**0.9525** (↓0.0004)	**0.4035** (↓0.0235)	**0.7801** (↓0.0018)	0.7216 (↑0.0019)
CEM	0.9501 (↓0.0004)	0.3612 (↓0.0275)	0.7653 (↑0.0002)	**0.7294** (↑0.0108)
ProbCBM	0.9516 (↓0.0001)	0.3840 (↓0.0192)	0.7756 (↑0.0013)	0.7276 (↑0.0008)
ViP-CBM (ours)	0.9483 (↓0.0013)	0.3507 (↓0.0277)	0.7561 (↓0.0085)	0.7114 (↓0.0055)
+margin	0.9487 (↓0.0013)	0.3776 (↓0.0008)	0.7553 (↓0.0117)	0.7069 (↓0.0046)

5.5.2 Trustworthiness under Concept Sparsity

In addition to the experiments with the CelebA dataset in Table 2, we perform experiments on the CUB dataset with 25%, 50%, 75%, and 100% of the 28 concept groups known to study the robustness and trustworthiness under inadequate concept sets. Figure 7 shows the minimum group EMRs and class accuracies of different models with different known concept ratios.



(a) Minimum group EMR. (b) Class Accuracy.

Figure 7: Minimum group EMR and class accuracy of different models under 25%, 50% ,75% and 100% concept groups known in the CUB dataset.

We explain the results through the view of information leakage [9], which describes the phenomenon that *jointly*-trained CBMs tend to achieve better task performance with additional information in concept prediction other than the existence of the concepts. Figure 7b shows that baseline joint models (scalar-CBM, CEM, and ProbCBM) achieve similar task accuracy in different concept sparsity, while the class accuracy in our ViP-CBM and its variant "margin" increases significantly as the number of known concepts grows. Thus, our ViP-CBM achieves less information leakage through the uniform binary concept classification in the projected space under joint training. Besides, Figure 7a shows that the minimum group EMR of our model also increases significantly with more known concepts, indicating that more knowledge of concepts leads to more precise prediction in each concept group. The performance of the "LP" version of our ViP-CBM is remarkably high with few concepts, but drops with the increase of known concepts, indicating that nonlinear projection could better separate the concept embeddings in the projected space. In summary, our ViP module alleviates the information leakage in jointly-trained CBMs with increased interpretability, which makes our model more trustworthy.

Taken together, these results suggest that ViP-CBM not only improves interpretability but also maintains strong generalization performance under varying levels of concept completeness and quality, indicating its potential suitability for real-world applications.

6 Conclusions

We present ViP-CBM, a parameter-efficient concept bottleneck architecture that simultaneously addresses model complexity and interpretability in multi-label concept learning. Experimental results show that our ViP-CBM, whose number of training parameters is comparable to a minimal scalar CBM, achieves competitive performance to conventional CBMs, and outperforms CEM in concept learning with low embedding dimensions. By visualizing the projected concept embedding space, our ViP-CBM provides more convincing explanations for concept learning than other embedding-based CBMs. The intervention experiments reveal that our ViP-CBM is more sensitive during intervention with joint training than baseline models, indicating less trade-off in our embedding representation. Experiments with sparse and noisy or missing concepts further demonstrate the trustworthiness and robustness of our ViP-CBM. In conclusion, our ViP-CBM is a low-parameter substitution for embedding-based CBMs with more interpretability and better intervention efficiency.

References

[1] D. Alvarez Melis and T. Jaakkola, "Towards robust interpretability with self-explaining neural networks", *Advances in neural information processing systems*, 31, 2018.

[2] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba, "Network Dissection: Quantifying Interpretability of Deep Visual Representations", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[3] M. Biehl, B. Hammer, and T. Villmann, "Prototype-based models in machine learning.", *Wiley interdisciplinary reviews. Cognitive science*, 7 2, 2016, 92–111, https://api.semanticscholar.org/CorpusID:26942723.

[4] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.*, "Language models are few-shot learners", *Advances in neural information processing systems*, 33, 2020, 1877–901.

[5] L. Chen, H. Zhang, J. Xiao, W. Liu, and S.-F. Chang, "Zero-shot visual recognition using semantics-preserving adversarial embedding networks", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, 1043–52.

[6] Z. Chen, Y. Bei, and C. Rudin, "Concept whitening for interpretable image recognition", *Nature Machine Intelligence*, 2(12), 2020, 772–82.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database", in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, 248–55, DOI: 10.1109/CVPR.2009.5206848.

[8] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov, "Devise: A deep visual-semantic embedding model", *Advances in neural information processing systems*, 26, 2013.

[9] M. Havasi, S. Parbhoo, and F. Doshi-Velez, "Addressing leakage in concept bottleneck models", *Advances in Neural Information Processing Systems*, 35, 2022, 23386–97.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, 770–8.

[11] T. Hoefler, D. Alistarh, T. Ben-Nun, N. Dryden, and A. Peste, "Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks", *Journal of Machine Learning Research*, 22(241), 2021, 1–124.

[12] J. Hu, L. Cao, T. Tong, Q. Ye, S. Zhang, K. Li, F. Huang, L. Shao, and R. Ji, "Architecture disentanglement for deep neural networks", in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, 672–81.

[13] L. Hu, T. Huang, H. Xie, C. Ren, Z. Hu, L. Yu, and D. Wang, "Semi-supervised concept bottleneck models", *arXiv preprint arXiv:2406.18992*, 2024.

[14] L. Hu, C. Ren, Z. Hu, H. Lin, C.-L. Wang, H. Xiong, J. Zhang, and D. Wang, "Editable concept bottleneck models", *arXiv preprint arXiv:2405.15476*, 2024.

[15] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, 3128–37.

[16] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, *et al.*, "Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav)", in *International conference on machine learning*, PMLR, 2018, 2668–77.

[17] E. Kim, D. Jung, S. Park, S. Kim, and S. Yoon, "Probabilistic concept bottleneck models", *arXiv preprint arXiv:2306.01574*, 2023.

[18] R. Kiros, R. Salakhutdinov, and R. S. Zemel, "Unifying visual-semantic embeddings with multimodal neural language models", *arXiv preprint arXiv:1411.2539*, 2014.

[19] P. W. Koh, T. Nguyen, Y. S. Tang, S. Mussmann, E. Pierson, B. Kim, and P. Liang, "Concept bottleneck models", in *International conference on machine learning*, PMLR, 2020, 5338–48.

[20] M. Lamm, J. Palomaki, C. Alberti, D. Andor, E. Choi, L. B. Soares, and M. Collins, "QED: A Framework and Dataset for Explanations in Question Answering", *Transactions of the Association for Computational Linguistics*, 9, 2021, 790–806, ed. B. Roark and A. Nenkova, DOI: 10.1162/tacl_a_00398, https://aclanthology.org/2021.tacl-1.48/.

[21] S. Lee, X. Wang, S. Han, X. Yi, X. Xie, and M. Cha, "Self-explaining deep models with logic rule reasoning", *ArXiv*, abs/2210.07024, 2022, https://api.semanticscholar.org/CorpusID:252873586.

[22] O. Li, H. Liu, C. Chen, and C. Rudin, "Deep learning for case-based reasoning through prototypes: a neural network that explains its predictions", in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, *AAAI'18/IAAI'18/EAAI'18*, New Orleans, Louisiana, USA: AAAI Press, 2018, ISBN: 978-1-57735-800-8.

[23] J. Liu, Z. Zhang, Z. Guo, L. Jin, X. Li, K. Wei, and X. Sun, "Kept: Knowledge enhanced prompt tuning for event causality identification", *Knowledge-based systems*, 259, 2023, 110064.

[24] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep Learning Face Attributes in the Wild", in *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

[25] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions", *Advances in neural information processing systems*, 30, 2017.

[26] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space", *arXiv preprint arXiv:1301.3781*, 2013.

[27] V. Monga, Y. Li, and Y. C. Eldar, "Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing", *IEEE Signal Processing Magazine*, 38(2), 2021, 18–44.

[28] T. Oikarinen, S. Das, L. M. Nguyen, and T.-W. Weng, "Label-free concept bottleneck models", *arXiv preprint arXiv:2304.06129*, 2023.

[29] W. Pan and C. Zhang, "The definitions of interpretability and learning of interpretable models", *arXiv preprint arXiv:2105.14171*, 2021.

[30] K. Panousis, D. Ienco, and D. Marcos, "Coarse-to-fine concept bottleneck models", *Advances in Neural Information Processing Systems*, 37, 2024, 105171–99.

[31] J. Pearl and D. Mackenzie, *The book of why: the new science of cause and effect*, Basic books, 2018.

[32] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification", *Machine learning*, 85, 2011, 333–59.

[33] M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why should i trust you?' Explaining the predictions of any classifier", in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, 1135–44.

[34] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead", *Nature machine intelligence*, 1(5), 2019, 206–15.

[35] B. Schölkopf, F. Locatello, S. Bauer, N. R. Ke, N. Kalchbrenner, A. Goyal, and Y. Bengio, "Toward Causal Representation Learning", *Proceedings of the IEEE*, 109, 2021, 612–34, https://api.semanticscholar.org/CorpusID:261325982.

[36] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering", in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, 815–23.

[37] C. Shang, S. Zhou, H. Zhang, X. Ni, Y. Yang, and Y. Wang, "Incremental residual concept bottleneck models", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, 11030–40.

[38] I. Sheth and S. Ebrahimi Kahou, "Auxiliary losses for learning generalizable concept-based models", *Advances in Neural Information Processing Systems*, 36, 2024.

[39] S. Sinha, G. Xiong, and A. Zhang, "A Self-explaining Neural Architecture for Generalizable Concept Learning", in *International Joint Conference on Artificial Intelligence*, 2024, https://api.semanticscholar.org/CorpusID:269484694.

[40] R. Socher, M. Ganjoo, C. D. Manning, and A. Ng, "Zero-shot learning through cross-modal transfer", *Advances in neural information processing systems*, 26, 2013.

[41] M. Vandenhirtz, S. Laguna, R. Marcinkevis, and J. Vogt, "Stochastic concept bottleneck models", *Advances in Neural Information Processing Systems*, 37, 2024, 51787–810.

[42] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The caltech-ucsd birds-200-2011 dataset", 2011.

[43] B. Wang, L. Li, Y. Nakashima, and H. Nagahara, "Learning bottleneck concepts in image classification", in *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 2023, 10962–71.

[44] E. Wong, S. Santurkar, and A. Madry, "Leveraging sparse linear layers for debuggable deep networks", in *International Conference on Machine Learning*, PMLR, 2021, 11205–16.

[45] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-Shot LearningA Comprehensive Evaluation of the Good, the Bad and the Ugly", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(9), 2019, 2251–65, DOI: 10.1109/TPAMI.2018.2857768.

[46] X. Xu, Y. Qin, L. Mi, H. Wang, and X. Li, "Energy-based concept bottleneck models: Unifying prediction, concept intervention, and probabilistic interpretations", in *The Twelfth International Conference on Learning Representations*, 2024.

[47] M.-C. Yeh and Y.-N. Li, "Multilabel deep visual-semantic embedding", *IEEE transactions on pattern analysis and machine intelligence*, 42(6), 2019, 1530–6.

[48] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson, "Understanding neural networks through deep visualization", *arXiv preprint arXiv:1506.06579*, 2015.

[49] M. Yuksekgonul, M. Wang, and J. Zou, "Post-hoc concept bottleneck models", *arXiv preprint arXiv:2205.15480*, 2022.

[50] M. E. Zarlenga, P. Barbiero, G. Ciravegna, G. Marra, F. Giannini, M. Diligenti, F. Precioso, S. Melacci, A. Weller, P. Lio, *et al.*, "Concept embedding models", in *NeurIPS 2022-36th Conference on Neural Information Processing Systems*, 2022.

[51] M. E. Zarlenga, P. Barbiero, Z. Shams, D. Kazhdan, U. Bhatt, A. Weller, and M. Jamnik, "Towards robust metrics for concept representation evaluation", in *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37, No. 10, 2023, 11791–9.

[52] Q. Zhang, Y. Yang, H. Ma, and Y. N. Wu, "Interpreting cnns via decision trees", in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, 6261–70.

[53] W. Zhang, L. Hu, Y. Wei, and B. Wu, "Verbalizer or classifier? a new prompt learning model for event causality identification", in *2023 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2023, 1–7.