ViP-CBM: Reducing Parameters in Concept Bottleneck Models by Visual-Projected Embeddings

Ji Qi, Huisheng Wang, H. Vicky Zhao
Department of Automation, Tsinghua University, Beijing, China
E-mail: {qij21,whs22}@mails.tsinghua.edu.cn, vzhao@tsinghua.edu.cn

***Abstract*—With the widespread application of deep neural networks (DNN) in key areas of personal and property security, the interpretability and trustworthiness of DNN models become increasingly significant. Concept Bottleneck Models (CBM) are popular interpretable models in which hidden layer neurons are trained to predict human-understandable concepts of the final task. However, existing CBMs encounter low efficiency and interpretability in the multi-label classification (MLC) of concepts: correlations of concepts are ignored by many CBMs, while others expressing correlations with complex models have very limited improvements in the model's performance. To address the challenge of massive parameters and limited interpretability in concept MLC problem, we propose a novel *Visual-Projected CBM (ViP-CBM)*, which transforms MLC of concepts into 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% with the number of training parameters when compared to other *embedding-based* CBMs. Experimental results show that our ViP-CBM achieves comparable or even better performance in concept prediction.**

I. 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. Researches on XAI mainly follow two routines: the post-hoc explanation that looks into already-trained black-box models, and human-interpretable modeling that aims to build a DNN with an entirely human-understandable inference process. Traditional post-hoc explanation methods such as LIME [1] and SHAP [2] try to identify the most important parts of the input data that support the model's decision. Network Dissection [3] attempts to explain image classification by convolutional neural networks (CNNs) through features in the intermediate layers, but cannot align these features perfectly with human cognition. In summary, end-to-end trained black-box DNNs usually cannot be entirely explained in a human-interpretable way, leading us to achieve interpretability by model construction from scratch. A natural conceit is to train models to provide information in the intermediate layers that support the decisions, which requires step-by-step design in reasoning with additional supervision. Concept-based models are popular human-interpretable models that explain the decisions of models with supporting high-level concepts. Concept Whitening [4] performs affine transformations in the latent space to align axes with concepts of interests, and research in [5] assigns each layer of a CNN to capture certain concepts from low-level (e.g., colors, textures) to high-level (e.g., objects). The above methods provide layer-wise concept explanations with massive annotation and high training costs.

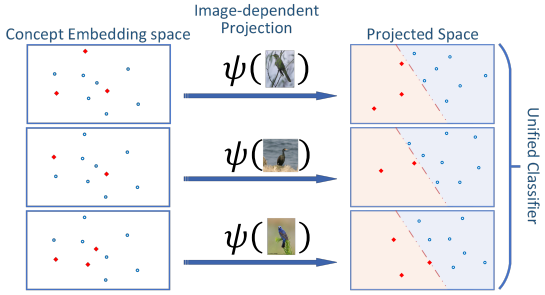


Fig. 1. Our ViP-CBM converts multi-label classification of concepts to unified binary classification of concept embeddings projected by visual features of input images. $\psi(\cdot)$ denotes the visual feature extractor.

In this work, we focus on a certain type of interpretable concept-based model for image classification, the Concept Bottleneck Model (CBM) [6]. CBMs split end-to-end prediction into two steps: first to predict the concepts and then to predict classes using only concepts. 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.

As the bottleneck of CBM, concept prediction in conventional CBMs usually learn concepts from an N-way binary classifier for all N concepts. Since the classifier treats all labels disconnected, correlations between concepts are neglected, leading to a loss of accuracy, especially in cases where the concept label set is large and a loss of interpretability as well. To address this issue, the work in [7] uses an autoregressive architecture inspired by the classifier chain [8] in multi-label classification (MLC) to capture correlations among concepts, which improves concept accuracy and task accuracy significantly. ECBM [9] employs an energy-based model to learn the relevance of concepts through inferences on a probability graph, where gradient descent is applied to minimize the energy function for prediction. In summary, current studies on solving the MLC problems in CBM involve more parameters and a complex model structure and require extensive samplings and calculations with limited performance improvements.

To improve the efficiency and interpretability of the MLC problem in concept prediction, we propose the **Visual-Projecting Concept Bottleneck Models (ViP-CBM)**. In ViP-CBM, visual features extracted from the input image are used 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. The contributions of this 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 other embedding-based CBMs. Our ViP-CBM achieves similar performance in both concept prediction and class prediction when compared to other CBMs.

II. Related Works

In this section, we review related works in CBMs and visual-semantic embedding that inspire our work.

A. Concept Bottleneck Models

Concept Bottleneck Models (CBM) [6] consists of two parts, the concept predictor and the class predictor. The concept predictor predicts human-specified concepts from the input image, while the class predictor receives predictions from the concept predictors as input to predict the classes. Earlier studies in CBMs employ scalar variables in the last layer before downstream tasks as concept predicting logits or probabilities. Assuming that the incompleteness of the concept set prevents the CBM from achieving higher task accuracy, side channels [7] are introduced in hard CBMs to represent undiscovered binary concepts to enhance model performance. Coop-CBM [10] adds a side branch before the concept prediction layer for immediate task prediction, leading to higher accuracy in concept prediction. Post hoc CBM [11] predicts concepts by projecting extracted features on Concept Activation Vectors (CAVs) trained from other supporting datasets by SVM or multimodal models, and Label-free CBM [12] further employ GPT-3 [13] for concept annotation to eliminate the need for densely annotated data. Both models convert pre-trained black-box models into CBMs and maintain task accuracy comparable to the original black-box networks.

To improve the expressivity of concepts in the model, the Concept Embedding Model (CEM) [14] learns a pair of positive and negative embeddings for each concept from the input to extend feature representations to higher dimensions. ProbCBM [15] and ECBM [9] 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 using Boltzmann energy models. However, all 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 and increases training costs, and deviates from the original intent of achieving interpretability on basic small models by feature supervision.

B. Visual-semantic Embeddings Models

Visual-semantic embedding models exploit label semantic relationships by leveraging the textural data from the label set to map visual features into a rich-semantic embedding space and are frequently used in image-text matching problems. In image classification with large-number labels, the Deep Visual Semantic Embedding (DeViSE) model [16] outperforms conventional one-way models since the latter treats all labels as disconnected and unrelated. The DeVisE model uses a pre-trained word2vec [17] model to embed words into vectors with semantic information preserved such as synonymy and a pre-trained visual model to extract feature vectors from input images. Classifying an input image is to assign the most relevant label to the image based on the similarities between the image and labels, which is measured by a generalized dot product of the visual feature vectors and concept embedding vectors where the metric matrix is trainable.

Visual-semantic embedding models are extended to sentence-level problems such as image description generation in [18], [19] using RNNs, and are used in zero- and few-shot learning due to the continuity of visual space and the use of unannotated text in [20]. In [21], the DeViSE model is extended for MLC, where the compatibility of an image and a label is represented by the embedding vectors projected by matrices of visual features extracted from the images. However, this model predicts only 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 require a large corpus to learn semantic embeddings of the label texts from semantic relations and syntactic components. Furthermore, these semantic embeddings are independent of the final tasks, 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. On the other hand, BotCL [22] learns a visual-semantic concept bottleneck in SENN [23] 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 in comparison to CBM and does not support intervention.

III. The Proposed ViP-CBM Methods

A. Motivation

To address the challenge of massive parameters and low interpretability in MLC of concepts, we propose ViP-CBM to transfer the MLC problem into a binary classification problem

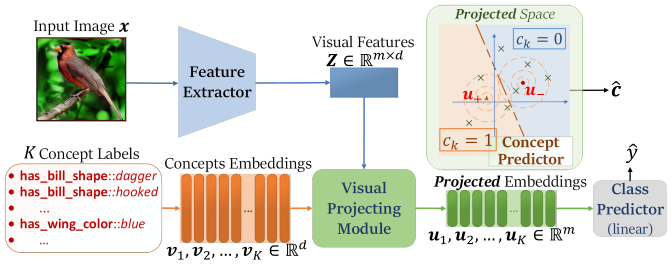


Fig. 2. Model structure of our ViP-CBM.

of all concept embeddings projected by the visual features of each input image individually. We modify the model in [21] and propose the ViP module to predict all concepts where embeddings are trained simultaneously with visual features.

Our main idea is as follows: The MLC problem in concept prediction can be transformed into finding bisections of the concept label set for different inputs, which is equivalent to finding binary classifications of concept embedding vectors for different input images in the concept embedding space. With a well-defined input-dependent projection that projects different classification surfaces for each input image into a unified hyperplane, the original concept bottleneck layer, which individually predicts each concept as positive or negative, can be transferred into a unified binary classifier for projected concept embedding vectors. By constructing a projection function based on the visual features, we can build a concept bottleneck model that predicts concepts from visual-projected concept embeddings. Therefore, our ViP-CBM leverages both visual and semantic information from the data, where both visual and concept representations are learnable.

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 $y^{(j)} \in \{1, \ldots, M\}$.

B. 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.

Concept embeddings and the ViP module. For concept embedding, we embed all K concepts $\{c_1, \ldots, c_K\}$ in $\mathbf{R}^d$ space as $\{\mathbf{v}_1, \ldots, \mathbf{v}_K\}$. Given an input image $\mathbf{x}$, a backbone CNN $\phi(\cdot)$ extracts visual features as a matrix $\mathbf{Z} = \phi(\mathbf{x}) \in \mathbf{R}^{m \times d}$. We use a nonlinear projection $\mathbf{u}_k = \theta(\mathbf{Z}\mathbf{v}_k) \in \mathbf{R}^m, k = 1, \ldots, K$, by adding a nonparametric nonlinear function $\theta(\cdot)$ on a simple linear projection. In this work, we propose the nonlinear projection as

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

where $\mathbf{z}_j \in \mathbf{R}^d, j = 1, \ldots, m$ is 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 nonlinearity.

Concept predictor. ViP-CBM converts 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 trainable anchor points $\mathbf{u}_+, \mathbf{u}_- \in \mathbf{R}^m$ and classify the concept c_k according to the Euclidean distance between its corresponding projected embedding $\mathbf{u}_k$ and the two anchors, which adds nonlinearity to a linear model, with projections of positive concepts closer to $\mathbf{u}_+$ and negative concepts closer to $\mathbf{u}_-$. Inspired by Triplet Loss in [24], the probability of the concept c_k being activated is

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

where $\sigma(\cdot)$ represents the sigmoid function, $a > 0$ is a learnable scaling parameter, $m_{c_k} \geq 0$ 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

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

where $\mathbf{1}_{(\cdot)}$ is the instructional function.

Class predictor. To reduce the model parameter set to the minimum, we use a simple linear classifier as the task predictor to predict the M final classes. Different from conventional CBMs, we employ 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.

Training strategy and loss function. Since we calculate concept probabilities and task probabilities, 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:

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

C. Parameter Reduction in ViP-CBM

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

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

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

In addition, the concept predictor in (2) without margins is equivalent to a linear classifier in the final decision. Thus, we can rewrite the concept predicting step with a linear classifier of the form $p(c_k = 1|\mathbf{u}_k) = \tilde{\mathbf{w}}^T\mathbf{u}_k + \tilde{b}, \tilde{\mathbf{w}} \in \mathbf{R}^m, \tilde{b} \in \mathbf{R}$ as follows, neglecting the nonlinearity in projection:

$$p\left(\hat{c}_k = 1|\mathbf{Z}, \{\mathbf{v}_i\}_{i=1}^K\right) = \sigma(\tilde{\mathbf{w}}^T(\mathbf{Z}\mathbf{v}_k) + \tilde{b}) = \sigma(\text{tr}[(\mathbf{v}_k\tilde{\mathbf{w}}^T)\mathbf{Z}] + \tilde{b}), \quad (6)$$

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

When using projected embeddings for class predicting instead of probabilities, the number of parameters of the class predictor in our ViP-CBM is m times that in the minimal CBM that predicts classes from scalars, which makes the total number of training parameters 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.

IV. Experimental Setup

A. Datasets

CUB-200-2011. The Caltech-UCSD Birds-200-2011 (CUB-200-2011) [25] dataset contains 11,788 images of 200 subcategories belonging to birds annotated with 312 binary concepts. We use the preprocessed dataset in [6] 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 (AwA2) [26] dataset 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.

B. Experimental Setup

Data augmentation. For data augmentation, 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.

ViP-CBM model settings. The visual feature extractor of our ViP-CBM is shown in Figure 3. We use a ResNet34 [27] pre-trained on ImageNet-1k [28] 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

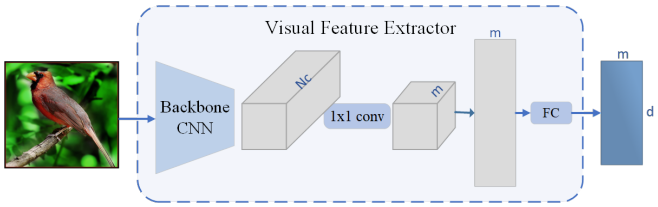


Fig. 3. Detailed structure of the visual feature extractor in ViP-CEM.

use a Fully Connected (FC) layer to reduce the l^2-entry inputs to m entries to get the feature matrix $\mathbf{Z} \in \mathbf{R}^{m \times d}$. The settings of the concept and class predictor follow the descriptions in Section III-B. We set symmetric margins following (3) with $m = 0.1$ to test the effects of margins.

Baselines. We compare our ViP-CBM to conventional joint CBM, CEM, and ProbCBM 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 [14].

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.

Hyperparameters settings. We set the weight between the two losses $\alpha = 5$. We use an SGD optimizer with a learning rate of 0.01 for the CUB dataset and 0.002 for the AwA2 dataset, both with momentum of 0.9 and weight decay of 5×10^{-4}. We train 400 epochs on the training split and compare model performance on the test split for the CUB dataset, and 250 epochs for the training split of the AwA2 dataset.

C. 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)**:

$$HS = \sum_{i=1}^{N}\sum_{k=1}^{K} \mathbf{1}_{(\hat{c}_k^{(i)} = c_k^{(i)})}/NC. \quad (7)$$

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

$$EMR = \sum_{i=1}^{N} \mathbf{1}_{(\hat{\mathbf{c}}^{(i)} = \mathbf{c}^{(i)})}/N. \quad (8)$$

TABLE I
PERFORMANCE OF OUR VIP-CBM IN COMPARISON WITH OTHER BASELINE CBMS FOR CUB AND AWA2 DATASETS

Model	CUB				AwA2			
	Hamming Score	Overall EMR	Min Group EMR	Class Accuracy	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	**0.9708**±0.0006	0.7816±0.0040	0.8288±0.0020	0.8782±0.0046
CEM	0.9506±0.0003	0.3887±0.0078	0.7651±0.0026	0.7186±0.0048	0.9696±0.0007	0.7646±0.0035	0.8288±0.0020	0.8794±0.0028
ProbCBM	0.9517±0.0010	0.4032±0.0132	0.7743±0.0058	**0.7268**±0.0113	0.9704±0.0008	0.7798±0.0084	0.8366±0.0049	**0.8822**±0.0026
ViP-CBM (ours)	0.9496±0.0009	0.3784±0.0223	0.7646±0.0037	0.7169±0.0048	0.9702±0.0010	0.7857±0.0055	0.8403±0.0047	0.8818±0.0031
+margin	0.9500±0.0013	0.3898±0.0090	0.7670±0.0053	0.7115±0.0075	0.9701±0.0009	**0.7906**±0.0071	**0.8431**±0.0047	0.8807±0.0037
LP	0.9398±0.0014*	0.3647±0.0148*	0.7248±0.0337*	0.6462±0.0463*	0.9692±0.0006	0.7741±0.0037	0.8335±0.0050	0.8801±0.0024

TABLE II
COMPARISON OF THE NUMBER OF TRAINING PARAMTERS

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

We can also use **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.

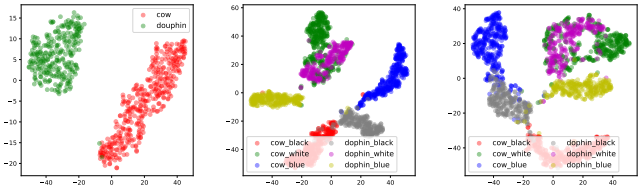
V. EXPERIMENTAL RESULTS

A. Model Performance

We set 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 using margins as "+margin" and ViP-CBM using linear projection $\mathbf{u}_k = \mathbf{Z}\mathbf{v}_k$ in ViP module as "LP". We conduct experiments with 5 different random seeds on the CUB dataset and 4 on the AwA2 dataset to compute the average scores and standard errors, marked as "mean±std" in our results. Table II shows that our model has only 40% of the training parameters of other *embedding*-based CBM, which is comparable to the scalar-CBM.

Table I 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 cerulean. 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 IV-B, 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 III-C.

For the experiments on the CUB dataset with a relatively large concept set and a small 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 concept 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. For the experiments on the AwA2 dataset with fewer concept and class labels and a larger amount of data, our ViP-CBM model ranks third in individual 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.



(a) Visual features $\mathbf{Z}$. (b) $\mathbf{u} = \mathbf{Z}\mathbf{v}$. (c) $\mathbf{u} = \theta(\mathbf{Z}\mathbf{v})$.

Fig. 4. t-SNE plot of the original visual features $\mathbf{Z}$ and concept representations $\mathbf{u}$ by both linear and nonlinear projections.

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 show that more than half of the experiment failed due to gradient explosion in training, indicating that nonlinearity in visual projection also allows larger learning rates, which increase the convergence speed and stability in training. Besides, the results of the successful experiments also prove that nonlinearity is necessary to bridge the performance gap due to the reduction of parameter numbers.

B. Interpretability

To reveal the interpretability of our ViP module proposed for MLC on a certain label set, we look into the spatial distributions of the projected concept embeddings. With the hypothesis of the continuity of visual feature space, 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"*. We visualize the spatial distributions of the original visual features $\mathbf{Z}$ and the linear and nonlinear projected embeddings of the 3 concepts $\mathbf{u}$ by visual features of images from both 2 classes

in the test dataset using a 2-dimensional t-SNE [29] plot, as is shown Figure 4. Original visual features are clearly clustered in Figure 4a, and so are projected embeddings in Figures 4b and 4c. In Figures 4b and 4c, 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"* overlap significantly, indicating that the ViP module gathers the same concepts in the projected space for all input images containing the concept. Furthermore, activated and inactivated concept embeddings are much more separated in Figure 4c than in Figure 4b, which reveals that nonlinear projection enhances performance in concept prediction.

VI. Conclusions

We propose ViP-CBM to reduce the number of parameters and enhance interpretability in the MLC problem in concept learning for CBMs. Experiments show that our ViP-CBM, whose number of training parameters is comparable to a minimal scalar CBM, achieves comparable performance to conventional CBMs, and outperforms CEM in concept learning with low embedding dimensions. The interpretability of ViP-CBM is also shown in experiments by visualizing the projected space. Our ViP-CBM is a low-parameter substitution for embedding-based CBMs with more interpretability.

References

[1] 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, pp. 1135–1144.

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

[3] 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)*, Jul. 2017.

[4] Z. Chen, Y. Bei, and C. Rudin, "Concept whitening for interpretable image recognition," *Nature Machine Intelligence*, vol. 2, no. 12, pp. 772–782, 2020.

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

[6] P. W. Koh, T. Nguyen, Y. S. Tang, *et al.*, "Concept bottleneck models," in *International conference on machine learning*, PMLR, 2020, pp. 5338–5348.

[7] M. Havasi, S. Parbhoo, and F. Doshi-Velez, "Addressing leakage in concept bottleneck models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 23 386–23 397, 2022.

[8] J. Read, B. Pfahringer, G. Holmes, and E. Frank, "Classifier chains for multi-label classification," *Machine learning*, vol. 85, pp. 333–359, 2011.

[9] 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.

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

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

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

[13] T. Brown, B. Mann, N. Ryder, *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[14] M. E. Zarlenga, P. Barbiero, G. Ciravegna, *et al.*, "Concept embedding models," in *NeurIPS 2022-36th Conference on Neural Information Processing Systems*, 2022.

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

[16] A. Frome, G. S. Corrado, J. Shlens, *et al.*, "Devise: A deep visual-semantic embedding model," *Advances in neural information processing systems*, vol. 26, 2013.

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

[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] 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, pp. 3128–3137.

[20] 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, pp. 1043–1052.

[21] M.-C. Yeh and Y.-N. Li, "Multilabel deep visual-semantic embedding," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 6, pp. 1530–1536, 2019.

[22] 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, pp. 10 962–10 971.

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

[24] 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, pp. 815–823.

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

[26] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 9, pp. 2251–2265, 2019. DOI: 10.1109/TPAMI.2018.2857768.

[27] 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, pp. 770–778.

[28] 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, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[29] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne.," *Journal of machine learning research*, vol. 9, no. 11, 2008.