Conceptual-Learning via Latent Approximations for Reinforcing Interpretability and Transparency

Maor Dikter, Tsachi Blau, Chaim Baskin
Technion – Israel Institute of Technology
{maor.dikter, tsachiblau}@campus.technion.ac.il chaimbaskin@technion.ac.il

Abstract

Concept Bottleneck Models (CBMs) have emerged as critical tools in domains where interpretability is paramount. These models rely on predefined textual descriptions to inform their decision-making process and offer more accurate reasoning. However, the selection of concepts used in the model is of utmost significance. In this study, we propose a CBM that approximates the embedding of concepts within the latent space of a Vision-Language Model (VLM). A concept selection process is then employed to optimize the similarity between the learned embeddings and the predefined ones. The derived bottleneck offers insights into the CBM's decision-making process, enabling more comprehensive interpretations. To evaluate the effectiveness of our approach, we conducted extensive experiments and achieved state-of-the-art performance on various benchmarks.

1. Introduction

In recent years, there has been an unprecedented increase in the utilization of neural networks across diverse fields, driven by their groundbreaking performance in solving complex problems and their capacity to extract patterns from vast datasets. This massive increase has caused the need to gain insight into how decisions are made. As deep networks become more complex, it is more viable to focus on explaining the decision using a post-hoc method, which performs an analysis of the model after its training. Methods such as feature importance scores and heat maps provide insights into which input features influenced a particular decision. However, in sensitive fields such as healthcare, relying solely on post-hoc explanations is insufficient, and understanding the elements that shaped the decision and the reasoning behind it is essential.

To address the need for a thorough understanding of the inner workings of models, interpretable-by-design models are being used. These inherently interpretable models constrain explanations within the model architecture , thereby ensuring transparency and offering a more reliable form of reasoning. A type of interpretable-by-design models are Concept Bottleneck Models (CBMs) [2]. The idea in this type of models is, given an input, first predicting an intermediate set of specified concepts, then using it directly to predict the target. These models enable interpretation in terms of high-level concepts and also allow for human interaction. Like many contemporary challenges, this framework employs Multi-Modal Models. Given our aim to quantify the relationship between an image and its corresponding textual representation, leveraging Visual-Language foundation models like CLIP [4] becomes a natural choice. The embedding space provided by these models enables us to gauge the similarity between the considered images and text. When experts can create the concepts in the bottleneck by hand, a shared aspect among current methods is their reliance on Large Language Models (LLMs) to identify concepts, for example, GPT-3 [1]. These techniques revolve around employing diverse prompts to direct the LLMs in understanding and extracting meaningful concepts. In a recent work by Yang et al. [6], the LaBo framework was proposed- a CBM that generates descriptions and employs submodular optimization to filter the concepts per class to form the bottleneck. The framework then uses cosine similarity between the encoded image and the selected concepts to make its predictions. Following this path, Yan et al. [5] demonstrate that relevant concepts can be derived from an approximation of the embedding space of Vision-Language Models (VLMs), and then locate the textual descriptions using nearest neighbor search.

While these methods present notable strengths, they also carry certain limitations that merit consideration. LaBo [6], for example, requires extensive annotations to accurately represent the data. An overly expansive bottleneck is not comprehensible to humans, could compromise the quality of concepts, and, as demonstrated in Yan et al. [5], can achieve comparable results to the use of random annotations. On the other hand, while it is essential to formulate a small and accurate set of descriptors that efficiently

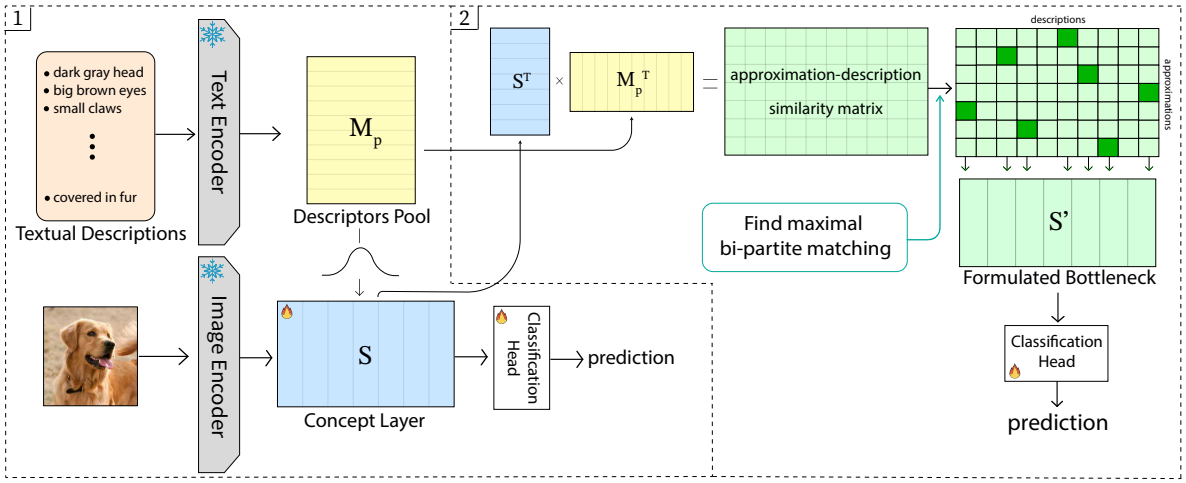

Figure 1. An overview of our model. Step 1 learns the embedding approximations, and step 2 selects the concepts and integrates our bottleneck.

represents our data, achieving so with a rough estimation from a limited number of descriptors falls short. In Yan et al. [5], the set of concepts is relatively small, which limits the framework's ability to achieve a comprehensive representation. A relatively large set of prior concepts is necessary to accurately estimate the distribution of textual embeddings. Once this distribution is established, reducing the number of concepts can improve interpretability. Furthermore, the method's concept selection process, being greedy, may lead to less than optimal outcomes.

Our approach addresses these shortcomings by utilizing a sufficiently large pool of prior concepts, a descriptors pool, to effectively approximate the latent space of the VLM. This enables us to learn a set of embeddings for our textual concepts, which involves training the embeddings as a single linear layer within our model. This process allows for the dynamic adjustment and refinement of embeddings directly based on the model's learning, ensuring they are optimally suited to represent the concepts. Subsequently, we construct a similarity matrix by calculating the cosine similarity between the learned embeddings and the descriptors pool. Finally, by employing the Hungarian method [3], we find a maximum perfect matching that yields the optimal set of textual representations based on their estimations.

2. Method

We outline the problem we aim to address and detail the approach in developing our model, illustrated in figure 1.

2.1. Problem Formulation

We consider a dataset D consisting of images and their corresponding labels, denoted as $D = \{(x, c)\}$, where each class $c \in C$ is associated with a set of attributes $A_c = \{a_{c_1}, \ldots, a_{c_k}\}$. We define the union of all these attribute sets as our descriptors pool $A = \cup_{c \in C} A_c$. Using a Vision-Language Model, equipped with text and image encoders E_T and E_I respectively, we project our dataset into the VLM's embedding space, $\mathbb{R}^d$. Through these encoders, we denote the obtained embeddings for our images and for our descriptors pool $I = \{E_I(x) | (x, c) \in D\}$ and $P = \{E_T(a) | a \in A\}$.

Our method explores Concept Bottleneck Models, which are models of the form $f(g(x))$ and comprises two functions: $g : \mathbb{R}^d \to \mathbb{R}^k$, mapping an input x to a concept space (the bottleneck of the model), and $f : \mathbb{R}^k \to \mathbb{R}$, mapping data from the concept space to make the final prediction. Learning a CBM can follow one of three approaches: an independent bottleneck approach, where the functions f and g are learned separately with their respective loss functions minimized independently; a sequential bottleneck strategy, which involves training f and g separately by learning g first, fixing it and then learning f; or a joint bottleneck technique, where f and g are trained simultaneously with an objective to minimize the combined sum of their losses.

Our research adopts a sequential-bottleneck approach to CBM construction. Initially, we learn an approximation to the function g, elaborated upon in section 2.2. Following this, we move on to formally construct the function g, in section 2.3. Finally once g is established, we fix it and train a function f, detailed in section 2.4. This methodology ensures that each component of the model is optimally trained for its specific role in the overall architecture.

2.2. Embedding Approximation Learning

Our initial goal is to construct an approximation of the conceptual bottleneck. We focus on learning a linear function, $S : \mathbb{R}^d \to \mathbb{R}^k$, characterized by its weight matrix,

Method	Datasets														
	CIFAR-10			CIFAR-100			Flower			CUB			Food		
	8	10	20	64	100	200	32	102	204	32	200	400	64	101	202
LaBo [6]	-	78.11	84.84	-	75.10	76.94	-	80.98	86.76	-	60.93	62.61	-	79.95	81.33
Yan et al. [5]	77.47	80.09	87.99	73.31	75.12	**77.29**	80.88	87.26	89.02	60.27	63.88	64.05	**78.41**	80.22	**81.85**
Our method	**80.24**	**83.6**	**89.87**	**73.61**	**75.52**	77.17	**87.94**	**90.19**	**91.07**	**65.29**	**70.03**	**69.86**	77.41	**80.44**	81.72

Table 1. The comparison of the proposed model compared to baseline state-of-the-art methods on various benchmarks

$[S] \in \mathbb{R}^{d\times k}$. This matrix serves as a collection of k unique d-dimensional embeddings. Since S transforms an image embedding into the concept space, appropriately training S results in optimal embeddings that accurately embody our concepts. To facilitate the learning of S, we train it along a function $W : \mathbb{R}^k \to \mathbb{R}^{|C|}$, responsible for making the model's classification. To guide the model towards accurate classification, we employ the Cross-Entropy loss function, defined as

$$CE(i,c) = \sum_{j=1}^{|C|} \delta_{j,c} \cdot log(W(S(i))_j)$$

Where $W(S(i))_j$ denotes the model's predicted probability for class j and $\delta_{j,c} = \begin{cases} 1 & \text{if } j = c \\ 0 & \text{if } j \neq c \end{cases}$ is the indicator of whether class c is the correct classification for image i.

Despite the structural resemblance of these functions to a Concept Bottleneck Model, they lack integration with the prior conceptual knowledge we wish to incorporate. Given that our textual concepts are within the embedding space of a VLM, we estimate the distribution of this space using our predefined descriptor pool P, ensuring S aligns with this space, while integrating image information towards the classification. To anchor our concept learning within the embedding space of the VLM, we use Mahalanobis distance. This statistical measure evaluates the distance between a point to a distribution. The Mahalanobis distance between a point $x \in \mathbb{R}^n$ and a distribution Q with mean and covariance matrix $\mu \in \mathbb{R}^n$ and $\Sigma \in \mathbb{R}^{n\times n}$, is $\sqrt{(x-\mu)^T \cdot \Sigma^{-1} \cdot (x-\mu)}$.

In our case, the distance between each row in $[S]^T$ and the distribution of the embedding space approximated by P is:

$$MA(S) = \frac{1}{k} \sum_{j=1}^{k} \sqrt{([S]_j^T - \mu)^T \cdot \Sigma^{-1} \cdot ([S]_j^T - \mu)}$$

Where μ and Σ are the mean and covariance of P.

The composite loss function guiding our model's optimization is formulated as $L = \lambda \cdot MA(S) + CE(i,c)$ and, in essence, for a given image embedding i, our model determines its prediction by

$$\hat{c} = argmax(W(S(i)))$$

2.3. Concept Selection

The essence of interpretability in a Concept Bottleneck Model is rooted in textual concepts, making it insufficient to merely derive the bottleneck approximation S. Thus, we venture further, identifying from P a subset of descriptors that closely align with S.

Using a matrix $M_p \in \mathbb{R}^{|P|\times d}$ that its rows are the elements of P, we construct an approximation-description similarity matrix $Sim \in \mathbb{R}^{k\times|P|}$ by:

$$Sim = [S]^T \cdot M_p^T$$

Here, $Sim_{i,j}$ measures the similarity between the $i-th$ conceptual approximation in $[S]$ and the $j-th$ descriptor in P. Our aim is to uniquely pair each of the k conceptual approximations with a descriptor, maximizing their joint similarity. To achieve this, we employ the Hungarian method [3], an algorithm that finds the optimal assignment that minimizes a total cost in a bipartite matching scenario. Inverting this, we turn the goal of maximizing similarity into one of minimizing cost by subtracting each matrix entry from its highest value, thereby aiming to minimize the total deviation from the maximum similarity.

Applying this approach enables us to identify the optimal set of embeddings and generate a new matrix, $[S]' \in \mathbb{R}^{d\times k}$, that incorporates the selected descriptors from P.

2.4. Bottleneck Integration

Building on the foundation laid in the previous sections, where we derived the matrix $[S]'$, we now establish the function $S' : \mathbb{R}^d \to \mathbb{R}^k$. This function transforms an image embedding x into the concept space through $S'(x) = [S]' \cdot x$. This function S' is exactly the function $g : \mathbb{R}^d \to \mathbb{R}^k$ of our CBM.

To determine the label prediction function $f : \mathbb{R}^k \to \mathbb{R}$, we fix g and proceed to learn a linear function $W' : \mathbb{R}^k \to \mathbb{R}^{|C|}$, in line with the approach in section 2.2. However, in this instance, we streamline the learning process by using the cross-entropy loss $CE(i,c)$ alone, focusing on guiding the transformation of data from the concept space to a classification. The prediction function f is then defined as $f(x') = argmax(W'(x'))$. By integrating g and f, we synthesize the complete Concept Bottleneck Model:

$$f(g(x)) = argmax(W'(S'(x))$$

No. of concepts	8	16	32	64	128	256
CIFAR-10	80.24	89.08	92.35	93.67	94.12	**94.28**
CIFAR-100	33.28	50.83	65.6	73.61	76.59	**77.33**
Flower	60.68	76.07	87.94	89.41	90.78	**90.98**
CUB	32.46	56.21	65.29	70.09	**70.17**	69.83
Food	34.01	56.29	72.24	77.41	80.11	**81.39**

Table 2. Impact of different concepts in the bottleneck on model performance.

		No. of concepts in Flower							
		8	16	32	64	128	256	102	204
Pool size	503	50.49	73.03	81.37	84.11	88.03	89.70	87.74	89.70
	5250	**60.68**	**76.07**	**87.94**	**89.41**	**90.78**	**90.98**	**90.19**	**91.07**
Selection method	NN	**60.68**	**76.07**	87.45	88.82	90.29	90.68	**90.39**	**91.37**
	Hungarian	**60.68**	**76.07**	**87.94**	**89.41**	**90.78**	**90.98**	90.19	91.07

Table 3. A study on the impact of the size of the descriptors pool, and the concept selection method. Our method's configuration is presented in the last row of each analysis.

		No. of concepts in CIFAR-100							
		8	16	32	64	128	256	100	200
Reg.	L_2	27.26	45.9	60.59	70.34	74.57	76.12	73.55	75.65
	MA	**33.28**	**50.83**	**65.60**	**73.61**	**76.59**	**77.33**	**75.52**	**77.17**

Table 4. An analysis of the effects of the regularization function.

3. Experiments

3.1. Experimantal Setup

Implementation Details In our study, we employed the CLIP [4] package developed by OpenAI, using the base architecture of ViT-B/32 for image encoding. The pool of descriptors employed is the descriptions provided in LaBo, which were generated by GPT-3 [1] and filtered per class. The training of the linear layers was facilitated using the PyTorch library.

Datasets Our methodology was evaluated across five datasets in the task of image classification: CIFAR-10, CIFAR-100, CUB, Flower, and Food. Training was conducted on the full datasets, with performance assessed based on the accuracy obtained from the test sets.

Baselines We compared our method with two other interpretable CBM strategies—LaBo and the framework developed by Yan et al., elaborated upon in section 1. The results of the test accuracies on the varying bottleneck sizes are presented in Table 1.

3.2. Ablation Study

To dissect the contribution of individual components within our model, an ablation study was conducted, focusing on four key aspects:

Bottleneck Size- Investigating the influence of concept quantity, we experimented with selections of 8, 16, 32, 64, 128, and 256 concepts, with findings documented in Table 2. The results suggests that incorporating more concepts leads to an improved representation of our data. Specifically, within the CUB dataset, 128 concepts slightly outperform 256 concepts. This might be attributed to the CUB classes being more conceptually related, all featuring various bird species.

Pool Size- We evaluated the impact of the size of the descriptors pool by comparing the pool used in our method with the one used in Yan et al [5]. Results reported in Table 3 supports the idea that having a more extensive pool of descriptors enhances our ability to closely match the descriptor distribution.

Concept Selection- The efficacy of our concept selection method was evaluated by contrasting the Hungarian method with the Nearest Neighbor (NN) algorithm, as employed by Yan et al [5]. Results outlined in Table 3 shows that choosing concepts by maximizing joint similarity yields better results compared to a greedy approach. The occasional suboptimal outcomes using this method could stem from the imperfect nature of the embedding approximations.

Regularization- The effectiveness of our regularization approach was measured by comparing the Mahalanobis distance in our loss function against the Euclidean distance (L_2 norm). Table 4's findings indicate that employing Mahalanobis distance for regularization proves to be more effective than using Euclidean distance.

References

[1] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020. 1, 4

[2] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International conference on machine learning*, pages 5338–5348. PMLR, 2020. 1

[3] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 2, 3

[4] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 4

[5] An Yan, Yu Wang, Yiwu Zhong, Chengyu Dong, Zexue He, Yujie Lu, William Yang Wang, Jingbo Shang, and Julian McAuley. Learning concise and descriptive attributes for visual recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3090–3100, 2023. 1, 2, 3, 4

[6] Yue Yang, Artemis Panagopoulou, Shenghao Zhou, Daniel Jin, Chris Callison-Burch, and Mark Yatskar. Language in a bottle: Language model guided concept bottlenecks for interpretable image classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19187–19197, 2023. 1, 3