Toward Explanation Bottleneck Models

Shin'ya Yamaguchi *
NTT, Kyoto University

Kosuke Nishida
NTT

Abstract

This paper proposes a novel interpretable model called *explanation bottleneck models* (XBMs), which are based on vision-language foundation models. XBMs generate a text explanation from the input and then predict a final task prediction based on the generated explanation by leveraging pre-trained vision-language encoder-decoder models. To achieve both the target task performance and the explanation quality, we train XBMs through the target task loss with the regularization penalizing the explanation decoder via the distillation from the frozen pre-trained decoder. Our experiments confirm that XBMs provide accurate and fluent natural language explanations, and the explanation can be intervened by human feedback.

1 Introduction

Although deep learning models can achieve remarkable performance on many applications, they are black-box, i.e., their output predictions are not interpretable for humans. Introducing concept bottleneck models (CBMs, [1]) is a promising approach to interpreting the output of deep models. In contrast to black-box models that directly predict output labels from input in an end-to-end fashion, CBMs first predict *concept* labels from input and then predict final target class labels from the predicted concepts. However, the existing CBMs depend on the fixed pre-defined concept sets to predict final labels; they can not provide interpretability to any other than the pre-defined concepts. We argue that this limitation presents a fundamental challenge for CBMs in achieving interpretable deep models. Although recent CBM variants leveraging foundation large language models [2, 3] enable to express concepts of arbitrary target classes, the interpretability is still restricted to a fixed and small number of concepts in order to guarantee the concept learnability and the final performance by limiting the number [4, 3, 5].

This paper tackles a research problem where we do not assume pre-defined concept sets for constructing interpretable deep neural networks. To this end, we propose a novel family of interpretable models called *explanation bottleneck models* (XBMs), which leverage pre-trained multi-modal encoder-decoder models that can generate text descriptions from input data (e.g., BLIP [6, 7]). Leveraging pre-trained multi-modal encoder-decoder enables capturing concepts that actually appeared in the input beyond pre-defined concept sets. Our key idea is to decode concepts as text explanations from input and then predict the final label with a classifier that takes the decoded explanations (Fig. 1). In contrast to CBMs, which make predictions based on pre-defined concepts, XBMs make predictions based on concepts actually appeared in the input data through the decoded explanations and can provide an intuitive interpretation of the final prediction tied to the input. Through end-to-end training, XBMs aim to generate explanations focusing on the textual features for solving the target task.

A major challenge for XBMs is forgetting the text generation capability during training on target tasks. Since target datasets usually lack ground-truth text labels, it is challenging to avoid catastrophic forgetting. To generate high-quality explanations, we introduce a training technique called *explanation distillation*, which penalizes the text decoders by the reference explanations generated by frozen

*Corresponding author. shinya.yamguchi@ntt.com

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

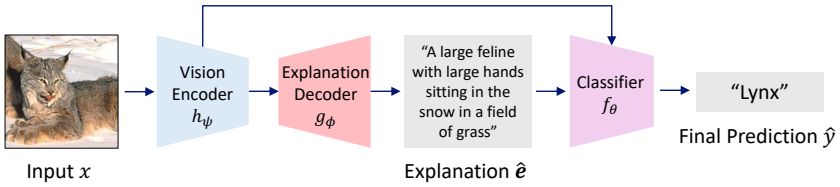


Figure 1: Explanation bottleneck models (XBMs). We propose an interpretable model that generates text explanations for the input embedding with respect to target tasks and then predicts final task labels from the explanations.

pre-trained text decoders. Solving target tasks with explanation distillation enables XBMs to decode explanations from input data in natural sentences without corruption.

We conduct experiments to evaluate XBMs on multiple datasets by comparing them to existing CBMs and black-box baselines regarding interpretability and target task performance. Our experiments show that XBMs can provide a more relevant explanation to input than the pre-defined concepts of existing CBMs while achieving competitive performance to black-box baselines and largely outperforming CBMs in target test accuracy. Further, we confirm the reliability and practicality of the XBMs' explanations through the experiments intervening with the random texts and the ground-truth explanations.

2 Explanation Bottleneck Models

This section introduces the principle of explanation bottleneck models (XBMs). XBMs are interpretable deep learning models that predict a final label from the generated explanation text from XBMs themselves. Since the predicted final labels are based on the generated explanation of input images, we can naturally interpret the explanation as the reason for the prediction of XBMs. Figure 2 illustrates the overview of training an XBM. An XBM consists of a visual encoder h_ψ, an explanation decoder g_ϕ, and a classifier f_θ for predicting final target labels. Among them, h_ψ and g_ϕ are initialized by an arbitrary pre-trained multi-modal encoder-decoder like BLIP [6]. f_θ is a multi-modal classifier built on a transformer that takes the generated explanations as input and conditions the cross-attention layers with image embeddings; this design is inspired by hybrid post-hoc CBMs [2] that uses input embeddings to complement missing concepts not in the predicted concepts. We also confirm the practicality when using a text classifier in Section E.1. In this section, we mainly describe XBMs with a multi-modal classifier. XBMs are trained by the target classification loss in an end-to-end manner. Since naïve training leads to collapse in generated text explanation, we avoid the collapse by *explanation distillation*. Explanation distillation penalizes the explanation decoder with a reference text generated from a frozen pre-trained text decoder g_{p} to prevent the decoders from forgetting the text generation capability.

2.1 Problem Setting

We consider a K-class image classification task as the target task. We train neural network models $h_\psi : \mathcal{X} \to \mathbb{R}^{d_\mathcal{X}}$, $g_\phi : \mathbb{R}^{d_\mathcal{X}} \to \mathcal{E}$, and $f_\theta : (\mathbb{R}^{d_\mathcal{X}}, \mathcal{E}) \to \mathcal{Y}$ on a labeled target dataset $\mathcal{D} = \{(x^i, y^i) \in \mathcal{X} \times \mathcal{Y}\}_{i=1}^N$, where $\mathcal{X}$, $\mathcal{E}$, and $\mathcal{Y}$ are the input, text explanation, and output label spaces, respectively. The text explanation space consists of token sequences of the length L with token vocabulary $\mathcal{V}$, i.e., $\mathcal{E} = \mathcal{V}^L$. h_ψ is a vision encoder, which embeds an input x into $d_\mathcal{X}$ dimensional space, g_ϕ is an auto-regressive text decoder that generates a text explanation $\boldsymbol{e} \in \mathcal{E}$ from an input embedding $h_\psi(x)$, and f_θ is a classifier that predicts a final target task label y. We assume that h_ψ and g_ϕ are initialized by pre-trained multi-modal model's parameters ψ_{p} and ϕ_{p}, which are pre-trained on large-scale text-image paired datasets with an existing method such as BLIP [6] and LLaVA [8]. Note that we do not assume ground truth text explanation set $\{\boldsymbol{e}^i\}_{i=1}^N$ in $\mathcal{D}$ for training g_ϕ.

This setting is similar to that of concept bottleneck models (CBMs, [1]), where a model predicts a final label y from a set of concepts $\{c^j \in \mathcal{C}\}_{j=1}^M$ decoded from input x instead of using $\boldsymbol{e}$. The major difference is in the assumption of pre-defined concept sets: our setting does not explicitly specify the words and phrases for the explanations, whereas CBMs explain the model's output based on the words and phrases in a pre-defined concept set $\{c^j\}$.

2.2 Objective Function

XBMs aim to achieve high target classification accuracy while providing interpretable explanations of the predictions. To this end, XBMs solve an optimization problem with a regularization term defined by the following objective function.

$$\min_{\theta,\phi,\psi} \mathcal{L}_{\mathrm{cls}}(\theta,\phi,\psi) + \lambda \mathcal{R}_{\mathrm{int}}(\phi,\psi), \tag{1}$$

$$\mathcal{L}_{\mathrm{cls}}(\theta,\phi,\psi) = \mathbb{E}_{(x,y)\in\mathcal{D}}\, \ell_{\mathrm{CE}}(f_\theta \circ g_\phi \circ h_\psi(x), y), \tag{2}$$

where $\mathcal{R}_{\mathrm{int}}(\cdot)$ is a regularization term that guarantees the fluency of the explanations generated from g_ϕ, λ is a hyperparameter for balancing $\mathcal{L}_{\mathrm{cls}}$ and $\mathcal{R}_{\mathrm{int}}$, and ℓ_{CE} is cross-entropy loss. Through this objective, the text decoder g_ϕ is trained to focus on the textual features that are useful for minimizing $\mathcal{L}_{\mathrm{cls}}$ while keeping the interpretability by $\mathcal{R}_{\mathrm{int}}$. We found that g_ϕ easily collapses their output without $\mathcal{R}_{\mathrm{int}}$. Thus, the design of $\mathcal{R}_{\mathrm{int}}$ is crucial for training XBMs. However, since we often do not have the ground truth explanation sets in a real-world target dataset $\mathcal{D}$, we can not directly penalize g_ϕ with supervised losses as $\mathcal{R}_{\mathrm{int}}$. To overcome this challenge, we introduce a distillation-based approach using pre-trained text decoders in the next section.

2.3 Explanation Distillation

XBMs utilize pre-trained multi-modal models as the initial parameters of the text (explanation) decoder g_ϕ. As an auto-regressive sequence model, the pre-trained text decoder g_{p} can learn a conditional distribution $q(\boldsymbol{e}|x)$ as

$$q(\boldsymbol{e}|x) = \prod_{l=1}^{L} q(e_l|x, \boldsymbol{e}_{<l}), \tag{3}$$

where L is the maximum token length, e_l is the l-th token, and $\boldsymbol{e}_{<l}$ is the text sequence before e_l. Since g_{p} is trained on large-scale text-image pairs, $q(\boldsymbol{e}|x)$ is expected to be able to generate a token sequence describing important information of various inputs x.

Our key idea is to leverage $q(\boldsymbol{e}|x)$ as the reference distribution for maintaining the interpretability of the generated explanation $\hat{\boldsymbol{e}} \sim p_\phi(\boldsymbol{e}|x)$, where $p_\phi(\boldsymbol{e}|x)$ is the model distribution of g_ϕ. If $p_\phi(\boldsymbol{e}|x)$ and $q(\boldsymbol{e}|x)$ are sufficiently close, it can be guaranteed that the interpretability of the sequence generated by $p_\phi(\boldsymbol{e}|x)$ approximate to that by $q(\boldsymbol{e}|x)$. Concretely, we compute the KL divergence between $p_\phi(\boldsymbol{e}|x)$ and $q(\boldsymbol{e}|x)$ as the regularization term $\mathcal{R}_{\mathrm{int}}$ in Eq. (1).

$$\begin{aligned}\mathcal{R}_{\mathrm{int}}(\phi,\psi) &= D_{\mathrm{KL}}(q\|p_\phi) = \textstyle\sum_{\boldsymbol{e}\in\mathcal{E}} q(\boldsymbol{e}|x) \log\left(\frac{q(\boldsymbol{e}|x)}{p_\phi(\boldsymbol{e}|x)}\right) \\ &= \mathbb{E}_{\boldsymbol{e}\sim q(\boldsymbol{e}|x)} \log\left(\frac{q(\boldsymbol{e}|x)}{p_\phi(\boldsymbol{e}|x)}\right).\end{aligned} \tag{4}$$

However, $D_{\mathrm{KL}}(q\|p_\phi)$ is computationally intractable because it requires multiple sequential sampling over $\mathcal{E} = \mathcal{V}^L$ from $q(\boldsymbol{e}|x)$ and the back-propagation through all sampling processes of $p_\phi(e_l|x, \boldsymbol{e}_{<l})$. To approximate Eq. (4), we focus on the connection to knowledge distillation [9]. That is, minimizing Eq. (4) can be seen as a knowledge distillation from g_{p} to g_ϕ. In such a sense, the approximation is

$$\mathcal{R}_{\mathrm{int}}(\phi,\psi) \approx -\sum_{\boldsymbol{e}\in\mathcal{E}} \mathbb{I}_{\boldsymbol{e}=\boldsymbol{e}_{\mathrm{p}}} \log p_\phi(\boldsymbol{e}|x) = -\log p_\phi(\boldsymbol{e}=\boldsymbol{e}_{\mathrm{p}}|x), \tag{5}$$

where $\boldsymbol{e}_{\mathrm{p}}$ is the sample from $q(\boldsymbol{e}|x)$ and $\mathbb{I}$ is the indicator function returning one when $\boldsymbol{e}$ equals to $\boldsymbol{e}_{\mathrm{p}}$ or returning zero otherwise; we omit the constant terms from the approximation for the simplicity. As a concrete procedure, we first generate $\boldsymbol{e}_{\mathrm{p}}$ from g_{p} and then penalize the output logits of g_ϕ through the cross-entropy loss for each output token in a next token prediction task. This approximation technique is well-known as sequence-level knowledge distillation [10] in the field of neural machine translation, and it works well in the knowledge distillation of auto-regressive sequence models. Sequence-level knowledge distillation corresponds to matching the modes of p and q and omits to transfer the uncertainty represented by the entropy $H(q)$ [10]. Nevertheless, we consider that this is sufficient for XBMs because the goal of XBMs is to provide interpretable explanations for target task predictions, not to replicate the pre-trained models perfectly. We call the regularization with Eq. (5) *explanation distillation*, and introduce it in training XBMs to maintain the text generation capability.

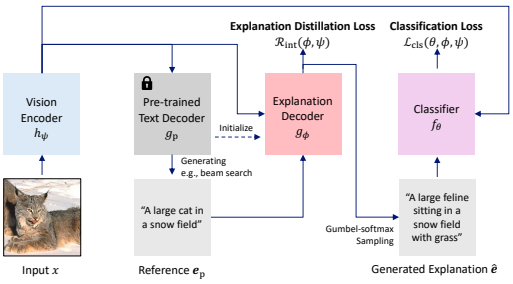


Figure 2: Training of XBMs. An XBM is optimized by the target task loss with explanation distillation. Explanation distillation leverages a reference explanation $\boldsymbol{e}_\text{p}$ generated from a pre-trained text decoder g_p for penalizing the output distribution of an explanation decoder g_ϕ to maintain the interpretable text generation capability of g_ϕ.

Algorithm 1 Training of XBMs

Require: Training dataset $\mathcal{D}$, vision encoder h_ψ, text decoder g_ϕ, classifier f_θ, pre-trained parameters $(\phi_\text{p}, \theta_\text{p})$, training batchsize B, step size η, trade-off parameter λ

Ensure: Trained models $(h_\psi, g_\phi, f_\theta)$

1: `# Initialize parameters`
2: $\phi \leftarrow \phi_\text{p}, \;\; \psi \leftarrow \psi_\text{p}$
3: **while** not converged **do**
4: $\quad \{(x^i, y^i)\}_{i=1}^B \sim \mathcal{D}$
5: $\quad$ `# Generating reference explanation`
6: $\quad \{\boldsymbol{e}_\text{p}^i\}_i^B \leftarrow \{\text{generate}(g_\text{p}, h_\text{p}(x^i))\}_i^B$
7: $\quad$ `# Gumbel-softmax sampling`
8: $\quad \{\hat{\boldsymbol{e}}^i\}_i^B \leftarrow \{\text{g_sampling}(g_\phi, h_\psi(x^i))\}_i^B$
9: $\quad$ `# Computing batch-mean losses`
10: $\quad \mathcal{L}_{\text{cls}}^B \leftarrow \frac{1}{B}\sum_{i=1}^B \ell_{\text{CE}}(f_\theta(h_\psi(x^i), \hat{\boldsymbol{e}}^i), y^i)$
11: $\quad \mathcal{R}_{\text{int}}^B \leftarrow \frac{1}{B}\sum_{i=1}^B \ell_{\text{CE}}(g_\phi \circ h_\psi(x^i), \boldsymbol{e}_\text{p}^i)$
12: $\quad$ `# Updating parameters via backprop.`
13: $\quad \theta \leftarrow \theta - \eta\nabla_\theta(\mathcal{L}_{\text{cls}}^B + \lambda\mathcal{R}_{\text{int}}^B)$, $\phi \leftarrow \phi - \eta\nabla_\phi(\mathcal{L}_{\text{cls}}^B + \lambda\mathcal{R}_{\text{int}}^B)$, $\psi \leftarrow \theta - \eta\nabla_\psi(\mathcal{L}_{\text{cls}}^B + \lambda\mathcal{R}_{\text{int}}^B)$
14: **end while**

Table 1: Performance and Interpretability Evaluation of XBMs on multiple target datasets.

	Bird			ImageNet		
	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)
Fine-tuned BLIP-ViT	83.48	N/A	N/A	65.21	N/A	N/A
Label-free CBM (ConceptNet)	15.37	0.5356	N/A	60.07	0.6826	N/A
Label-free CBM (GPT-3)	77.74	0.6904	N/A	64.28	0.7026	N/A
Frozen BLIP + $f_\theta(h_\psi(x), \hat{\boldsymbol{e}})$	68.03	0.7535	173.5	56.04	0.7732	199.5
XBM w/o $\mathcal{R}_{\text{int}}$	61.94	0.5137	431.0	66.58	0.5020	517.1
XBM (Ours)	**80.99**	**0.7942**	**166.8**	**67.83**	**0.7920**	**122.8**

2.4 Algorithm

Training We show the training procedure in Algorithm 1. In the training loop, we first generate the reference and predicted explanations $\boldsymbol{e}_\text{p}$ and $\hat{\boldsymbol{e}}$ by $\text{generate}(\cdot)$ and $\text{g_sampling}(\cdot)$, respectively (line 4 and 5). To approximate the mode of $q(\boldsymbol{e}|x)$ and ensure the quality as the reference, we generate $\boldsymbol{e}_\text{p}$ from frozen g_p by beam search following the previous work [10]. For sampling $\hat{\boldsymbol{e}}$, we introduce the Gumbel-softmax trick [11] to retain the computation graph for the end-to-end training with back-propagation. The l-th token can be approximately sampled by

$$e_l = \text{softmax}((\log(g_\phi(h_\psi(x))) + \mathbf{g})/\tau), \tag{6}$$

where $\mathbf{g} = \{\text{g}_1, ..., \text{g}_{|\mathcal{V}|}\}$ is a vector of length $|\mathcal{V}|$ where each element is sampled from $\text{Gumbel}(0, 1)$ and τ is the temperature parameter. Intuitively, the temperature τ controls the diversity of the token outputs from g_ϕ; larger τ stimulates more diverse outputs. To obtain diverse and accurate tokens for describing input, we apply exponential annealing to the temperature values according to the training steps, i.e., $\tau^{(i+1)} = \tau^{(0)} \exp(-r_\text{a} i)$, where i and r_a are training step and annealing rate. This allows XBMs to focus on the diversity of the output tokens in the early training steps and on the quality in the later steps. We evaluate this design choice in Appendix C. After sampling $\boldsymbol{e}_\text{p}$ and $\hat{\boldsymbol{e}}$, we update all trainable parameters according to the objective function Eq. (1).

3 Experiment

We evaluate XBMs on multiple visual classification tasks. We conduct qualitative and quantitative experiments on the explanation outputs of XBMs to evaluate the target performance and the interpretability. We provide the experimental setting in Appendix B.

Table 2: Qualitative evaluation of explanation outputs.

	Bird (Yellow Bellied Flycatcher) 	ImageNet (Lynx)
Pre-trained BLIP (Caption)	A bird perched on a wire fence with leaves on the ground and a blurry background.	Cat walking through the grass in the woods at night with it's eyes open.
Label-free CBMs (Top-3 Concept)	olive-colored sides (0.77) green head (0.55) a small, green body (0.52)	feline (0.98) long, sharp claws (0.53) mau (0.17)
XBMs w/o $\mathcal{R}_{\text{int}}$ (Text Explanation)	2222222222222 2222222222	when when when when when when when when when when
XBMs (Text Explanation)	A small green and yellow bird perched on a wire fence with leaves on the side.	Furry feline walking in the woods at night with its eyes open and one paw on the ground.

3.1 Design Evaluation of XBMs

Quantitative Evaluation Table 4 demonstrates the quantitative performance and interpretability of XBM-BLIP on Bird [12] and ImageNet [13]. For the target performance, our XBMs outperformed the Label-free CBM baselines [3] and achieved competitive performance with the black-box baseline in the test accuracy. In particular, XBM achieved high performance on datasets where label-free CBM did not perform well. This can be caused by insufficient pre-defined concepts due to the limited vocabulary in ConceptNet and GPT-3 about describing objects in these datasets, whereas XBMs promote multi-modal understanding by training the explanation decoder to describe arbitrary objects useful for the target dataset with unlimited vocabulary. For interpretability, XBMs outperformed CBMs in CLIP-Score (i.e., the similarity between the input image and explanation in the CLIP feature space). This indicates that the explanations from XBMs are more factual to the input images than the concept outputs of CBMs, which are in pre-defined concept sets.

Furthermore, the ablation study in the bottom rows of Table 4 shows that the objective function in Eq. (1) works effectively as we expected. Compared to the frozen BLIP baselines, which simply apply fixed pre-trained BLIP to generate text captions, our XBM significantly improved all of the test accuracy, CLIP-Score, and GPT-2 Perplexity (a text-fluency metric measured on GPT-2). This suggests that optimizing text decoders with respect to target tasks guides the generated explanation to be informative and target-related for solving the task. We also confirm that the regularization term $\mathcal{R}_{\text{int}}$ by explanation distillation (Eq. (5)) is crucial to generate meaningful explanation; XBM w/o $\mathcal{R}_{\text{int}}$ catastrophically degraded CLIP-Score and GPT-2 Perplexity.

Qualitative Evaluation Table 2 shows the qualitative studies of explanations generated from XBMs; we also show the other examples in Appendix 5. For comparison, we also show the top-3 concept outputs of CBMs and the generated captions of pre-trained BLIP, i.e., the initial states of XBMs. The text explanations of XBMs contain more detailed information than pre-trained BLIP. This is because the target classification loss $\mathcal{L}_{\text{cls}}$ forces the text decoders to describe target-related visual information to solve the task. Importantly, XBMs without explanation distillation $\mathcal{R}_{\text{int}}$ generate totally broken explanations, indicating the objective function of XBMs succeed in training the models to focus on the tokens related to the target task without the collapse of explanations. In contrast to CBM's concepts, the explanations from XBMs tend to be aligned with visual features appearing in input images rather than describing input by pre-defined knowledge. This is easy for humans to understand when interpreting the output of the models.

We also analyze the transition of the generated explanations in Fig. 3. We print the text explanation of XBMs and the top-10 word occurrence for the input class at 0, 20, and 40 epochs. According to the training epoch, the explanations and words progressively focus on detailed and target-related information in images. Concretely, in this example, the XBM is optimized to describe "yellow beak (mouth)", a key feature of California Gull. These suggest that XBMs can provide interpretable and useful explanations for humans.

3.2 Reliability Evaluation via Human Intervention

CBMs allow the debugging of the model behavior through human intervention in the predicted concepts [1]. Similarly, we can debug the behavior of XBMs by intervening in the generated

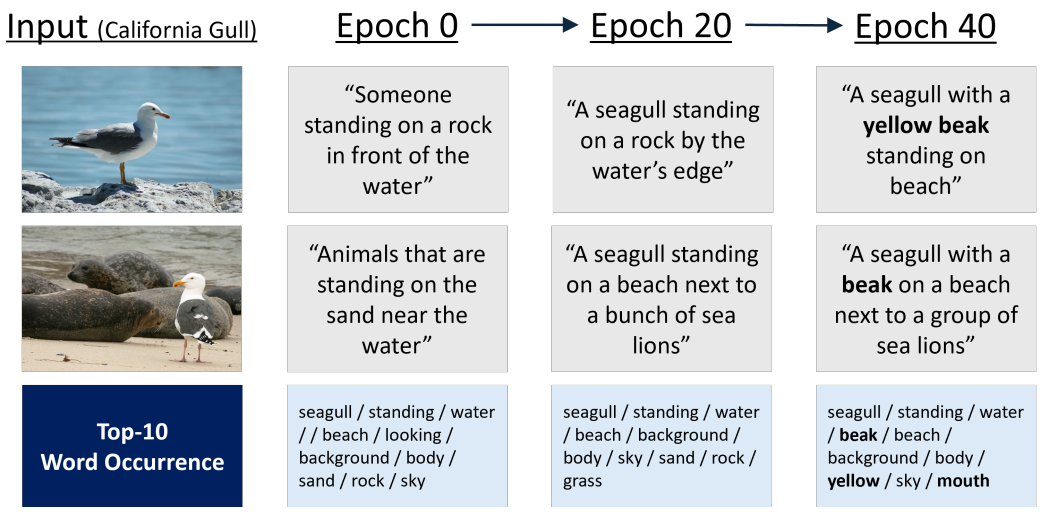


Figure 3: Transition of XBM's explanation outputs during training (please zoom in).

Table 3: Evaluation of Intervened XBMs on Bird.

	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)
XBM-BLIP	80.99	0.7942	166.8
Intervened XBM-BLIP (Randomized)	44.42	0.4497	4631.1
Intervened XBM-BLIP (Ground-Truth)	82.21	0.8179	104.5

explanations. Here, we show examples of an intervention in which all explanations are replaced to check the effect of the explanation quality on the final classification results. At inference, we replace the generated explanations from the explanation decoder with modified explanations. We tested two types of interventions: (i) randomized and (ii) ground-truth explanations. For randomized explanation, we used a token sequence uniformly sampled from the vocabulary space for the length of the originally generated explanation. For ground-truth explanation, we used the extended annotation set for Bird proposed by [14]. Table 3 shows the performance of the intervened XBM-BLIP models. The intervened explanations with randomized explanations significantly degraded the performance of XBM-BLIP, indicating that the generated explanations are essential to achieving high performance. In contrast, the intervention with ground-truth explanations largely improved the performance. This suggests that higher-quality explanations can yield higher performance, and intervening with human explanations is helpful for XBMs to improve their performance. In other words, the final prediction of XBMs largely depends on the content of the generated explanation $\hat{e}$, indicating that $\hat{e}$ is a reliable explanation for the final prediction. To conclude, these results support the debuggability of XBMs and the reliability of the generated explanations.

4 Conclusion

In this paper, we presented a novel interpretable deep neural networks called explanation bottleneck models (XBMs). By leveraging foundation vision-language models, XBMs generate explanations corresponding to input and output in the forms of natural language description. To ensure both the target task performance and the explanation quality, XBMs are optimized by the target task loss with explanation distillation, which penalizes the divergence between the distributions of the training and pre-trained text decoders. Experiments show that XBMs can achieve both high target task performance and accurate and fluent explanations; they achieve competitive performance to black-box baselines and largely outperform CBMs in target test accuracy. We believe that this work introduces a new perspective on natural language explanations and advances the study of interpretable foudation models to the next paradigm.

Appendix

A Related Work

The main research directions of the interpretability of black-box deep neural networks are briefly divided into attribution-based and concept-based methods. Attribution-based methods such as CAM [15] and GradCAM [16] generate a localization map representing important regions for the model predictions for specific classes. However, since the maps generated by attribution-based methods do not have information other than that they responded to the predictions, they are less interpretable regarding what semantic input features contribute to the output. In contrast to these methods, our XBMs can generate semantically interpretable heatmaps via cross-attention between image and text explanations, which can be decomposed at the level of noun phrases.

On the other hand, concept-based methods such as TCAV [17] and CBMs [1] compute contribution scores for pre-defined concepts on intermediate outputs of models. Among them, CBMs are highly relevant to our XBMs since both have interpretable intermediate layers in models. CBMs predict concept labels and then predict final class labels from the predicted concepts. The original CBMs have the challenge of requiring human annotations of concept labels [18, 19, 20]. Post-hoc CBMs [2] and Label-free CBMs [3] addressed this challenge by automatically collecting concepts corresponding to target task labels by querying large language models (e.g., GPT-3 [21]) or existing concept banks (e.g., ConceptNet [22]). However, CBMs' explanations are still restricted to pre-defined concepts, and they are not necessarily reliable because CBMs often predict the concepts without mapping to corresponding input regions [23]. On the contrary, our XBMs directly generate natural language explanations to interpret the model outputs without pre-defined concepts.

Similar to our work, a few works attempted to generate linguistic explanations for target classification models [24, 25]. However, these methods require ground truth text explanations for training models, which are expensive and restrict applications. Our XBMs address this limitation by learning explanation generation by the classification loss and explanation distillation using a pre-trained text decoder.

B Setting

Implementation Our basic implementation of XBMs is based on BLIP [6] because of its simplicity; we denote this model as XBM-BLIP. That is, as the visual encoder h_ψ, we used the ViT-B/32 [26]. For the classifier f_θ, we used a BERT-base transformer [27]; we input $h_\psi(x)$ into the cross-attention layers when using a multi-modal classifier inspired by BLIP [6]. We initialized ϕ and ψ by the BLIP model pre-trained on image captioning tasks in the official repository[2]. We also report the results using larger pre-trained multi-modal models of LLaVA [8]. We used v1.5 and v1.6 of LLaVA with multiple language model backbones (LLaMA2-7B [28], Vicuna-7B [29], and Mistral-7B [30]); we denote these models as XBM-LLaVA.

Baselines We compare XBMs to black-box and interpretable baselines in performance and interpretability. **Fine-tuned BLIP-ViT** is the black-box baseline, which directly optimizes the visual encoder of BLIP via fine-tuning. **Label-free CBM** [3] is a state-of-the-art concept bottleneck model, which automatically constructs pre-defined concept sets from ConceptNet [22] or GPT-3 [31] and then constructs concept embedding matrix via CLIP vision and text encoder. We used BLIP-ViT as the backbone vision encoder of label-free CBMs. **Frozen BLIP** baselines use frozen BLIP to generate text explanations and predict final labels by a multi-modal $f_\theta(h_\psi(x), \hat{\boldsymbol{e}})$ or text classifier $f_\theta(\hat{\boldsymbol{e}})$. We also show the results of **XBM w/o** $\mathcal{R}_{\mathrm{int}}$, which updates g_ϕ only on the classification loss Eq. (2).

Datasets We used four image datasets for classification tasks in various domains: **Aircraft** [32], **Bird** [12], **Car** [33], and **ImageNet** [13]. Aircraft, Bird, and Car are fine-grained image datasets, and ImageNet is a large-scale general image dataset. For datasets other than ImageNet, we randomly split a dataset into $9:1$ and used the former as the training set and the latter as the validation set. For ImageNet, we set the split ratio $99:1$ and used the official validation set as the test dataset.

[2] `model_base_caption_capfilt_large.pth` in `https://github.com/salesforce/BLIP`

Table 4: Performance and Interpretability Evaluation of XBMs on multiple target datasets.

	Aircraft			Car		
	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)
Fine-tuned BLIP-ViT	77.86	N/A	N/A	90.08	N/A	N/A
Label-free CBM (ConceptNet)	15.27	0.5561	N/A	17.67	0.6025	N/A
Label-free CBM (GPT-3)	44.47	0.6153	N/A	77.91	0.6091	N/A
Frozen BLIP + $f_\theta(h_\psi(x), \hat{\boldsymbol{e}})$	45.23	0.6824	155.8	80.53	0.6555	168.8
XBM w/o $\mathcal{R}_{\text{int}}$	70.78	0.4730	322.6	86.59	0.4792	415.3
XBM (Ours)	**74.09**	**0.7151**	**129.8**	**89.47**	**0.7173**	**131.8**
	Bird			ImageNet		
	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)
Fine-tuned BLIP-ViT	83.48	N/A	N/A	65.21	N/A	N/A
Label-free CBM (ConceptNet)	15.37	0.5356	N/A	60.07	0.6826	N/A
Label-free CBM (GPT-3)	77.74	0.6904	N/A	64.28	0.7026	N/A
Frozen BLIP + $f_\theta(h_\psi(x), \hat{\boldsymbol{e}})$	68.03	0.7535	173.5	56.04	0.7732	199.5
XBM w/o $\mathcal{R}_{\text{int}}$	61.94	0.5137	431.0	66.58	0.5020	517.1
XBM (Ours)	**80.99**	**0.7942**	**166.8**	**67.83**	**0.7920**	**122.8**

Training We trained the models by the AdamW [34] optimizer with the initial learning rate of 3.0×10^{-5} that decayed by cosine annealing. The training epochs were 100 on the Aircraft/Bird/Car datasets and 5 on the ImageNet dataset. We used mini-batch sizes of 32. The input samples were resized into resolutions of 384×384 for XBM-BLIP and 336×336 for XBM-LLaVA according to the setting of vision encoders. We used λ of 0.1 and τ of 10 with exponential annealing by $r_{\text{a}} = 1.0 \times 10^{-4}$ if not otherwise noted; we discuss the effect of λ and τ in Section G. For the experiments on XBM-LLaVA, we fine-tuned the LoRA adapter parameters [35] of backbone language models instead of the entire parameters. We selected the final model by checking the validation accuracy for each epoch. We implemented the training and evaluation with PyTorch-1.13. We ran the experiments three times on a 24-core Intel Xeon CPU with eight NVIDIA A100 GPUs with 80GB VRAM and recorded the average evaluated on the final models; we omit the standard deviations for saving spaces, but we have confirmed the statistical significance of our method with a p-value < 0.05 toward baselines.

Evaluation Metrics We report test accuracy as the target task performance. For the interpretability evaluations, we introduce **CLIP-Score** [36, 37], which is based on the cosine similarity between image embeddings and text embeddings on CLIP, i.e., higher is better. CLIP-score was originally used to evaluate image captioning based on the relevance of the output captions to the input images. Since it is highly sensitive to the hallucinations in the captions as reported in [37], CLIP-score can be used to assess the factuality of explanations. For XBMs, we measured averaged CLIP-Scores between test inputs and the output explanations. For Label-free CBMs, we measured averaged CLIP-Scores between test inputs and the output concept texts with the binary output of the concept bottleneck layer greater than 0.05; this threshold follows [3]. We also introduce **GPT-2 Perplexity** as a measure of fluency in XBM's output explanations. In general, perplexity scores on language models are calculated by the averaged cross-entropy of the next token probabilities and thus represent the fluency of the generated texts because the lower perplexity means that the sentence is composed of words that are likely to occur probabilistically. Inspired by [38], we computed perplexity scores of explanations on GPT-2 [39]. That is, the generated explanations are unbiasedly evaluated by an external language model. GPT-2 perplexity is helpful as a metric of the fluency of explanations because it shows the proximity to the natural text distribution learned by GPT-2. We used open-sourced GPT-2 in huggingface transformers [40] to maintain reproducibility.

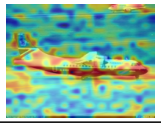
C Additional Quantitative Experiments

Table 5 shows the quantitative evaluation results on the Aircraft and Car datasets, which are omitted in the main paper due to the page constraint. The evaluation protocol is the same as Section 3.1.

D Additional Qualitative Experiments

Table 5 shows the qualitative evaluation results on the Aircraft and Car datasets, which are omitted in the main paper due to the page constraint. The evaluation protocol is the same as Section 3.1. Similar to Table 2, our method succeeded in capturing the semantic concepts of input images in the text

Table 5: Qualitative evaluation of explanation outputs.

	Aircraft (ATR-42)	Car (Hummer)
Pre-trained BLIP (Caption)	These two planes are parked on the tarmac at an airport run way.	Someone is driving a red jeep on a snowy road with trees in the background.
Label-free CBMs (Top-3 Concept)	canard foreplanes (0.70) a single-engine propeller (0.64) fixed landing gear (0.58)	2500HD model designation (0.63) off-road tires (0.40) distinct Suzuki grille (0.39)
XBMs (Text Explanation)	A small white and blue airplane on a runway at an airport with another plane in the background.	A truck with off-road tires driving through the snow in the wintertime with trees in the background.
XBMs (Top-3 Concept Phrase)	another plane in the background (0.34) a small white and blue airplane (0.31) a runway at an airport (0.22)	a truck with off-road tires (0.36) trees in the background (0.26) the wintertime (0.23)
XBMs (Cross-Attn. Heatmap)		

explanation. Also, the concept phrases and cross-attention heatmaps show that the captured semantic concepts contribute to the final output and the main focus of models is on the target objects.

E XBMs with Large Vision-Language Models

Here, we evaluate the scalability and practicality of XBMs by combining them with larger vision-language models than BLIP. Instead of BLIP, we used the LLaVA models with various language model backbones [8]. Table 6 shows that leveraging the high-performance vision-language model in XBMs yields better performance and interpretability scores, suggesting that the XBM's objective function can enhance the multi-modal understanding ability even if using the large vision-language models pre-trained on massive image-text pairs. This emphasizes the flexibility of XBM, consisting of arbitrary vision-language models.

E.1 XBMs with Text Classifier

Table 6 also evaluates XBMs with a text classifier $f_\theta(\hat{e})$, which relies only on text information for the final predictions. Although XBM-BLIP with $f_\theta(\hat{e})$ drops the performance from one with a multi-modal classifier $f_\theta(h_\psi(x), \hat{e})$, switching the backbone from BLIP to LLaVA [8] resolves the performance gap. This indicates that more sophisticated vision-language models make XBMs generate informative text explanations, and they can achieve practical performance even when not using input features $h_\psi(x)$.

F Additional Results of ImageNet Segmentation

We additionally compare our method with existing attribution methods on BLIP-ViT. Note that we omit this result from the main paper because, strictly speaking, BLIP-ViT and XBM-BLIP are different models and thus this evaluation is not a direct comparison. By following Chefer et al. [41], we tried LRP [42], partial-LRP [43], rollout [44], raw attention output from BLIP-ViT, GradCAM [16], and the method of [41]. Table 7 shows the results of ImageNet Segmentation with the same setting of Table 8. Surprisingly, the cross-attention output of the classifier f_θ (i.e., Pre-trained BLIP and XBM-BLIP) significantly outperformed the conventional visualization methods in the segmentation metric. This indicates that visualization explanation outputs of XBMs are quite accurate and reliable as the interpretation of model outputs.

Table 6: Evaluation of XBMs with large vision-language models

Aircraft	Text Classifier $f_\theta(\boldsymbol{e})$			Multi-modal Classifier $f_\theta(h_\psi(x), \boldsymbol{e})$		
	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)
Frozen BLIP	3.95	0.6824	155.8	45.23	0.6824	155.8
XBM-BLIP	24.36	0.7084	145.9	74.09	0.7151	129.8
Frozen LLaVA-v1.5-LLaMA-7B	59.21	0.7500	227.4	73.03	0.7514	227.3
XBM-LLaVA-v1.5-LLaMA-7B	64.11	0.7515	179.5	78.77	0.7595	184.8
XBM-LLaVA-v1.6-Vicuna-7B	68.64	0.7842	22.9	82.08	0.7758	34.7
XBM-LLaVA-v1.6-Mistral-7B	67.06	0.7769	39.9	81.55	0.7851	21.9
Bird	Text Classifier $f_\theta(\boldsymbol{e})$			Multi-modal Classifier $f_\theta(h_\psi(x), \boldsymbol{e})$		
	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)
Frozen BLIP	5.53	0.7535	173.5	68.03	0.7535	173.5
XBM-BLIP	19.52	0.7910	168.5	80.99	0.7942	166.9
Frozen LLaVA-v1.5-LLaMA-7B	75.20	0.7788	140.8	75.67	0.7788	107.3
XBM-LLaVA-v1.5-LLaMA-7B	80.87	0.7981	107.3	83.07	0.8037	21.8
XBM-LLaVA-v1.6-Vicuna-7B	82.33	0.8130	21.0	84.93	0.8154	21.6
XBM-LLaVA-v1.6-Mistral-7B	81.53	0.8101	16.7	84.73	0.8110	16.7
Car	Text Classifier $f_\theta(\boldsymbol{e})$			Multi-modal Classifier $f_\theta(h_\psi(x), \boldsymbol{e})$		
	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)
Frozen BLIP	7.98	0.6091	168.8	77.91	0.6091	168.8
XBM-BLIP	27.57	0.7127	168.5	89.47	0.7173	131.8
Frozen LLaVA-v1.5-LLaMA-7B	83.50	0.7236	99.8	91.19	0.7300	97.2
XBM-LLaVA-v1.5-LLaMA-7B	86.18	0.7300	97.2	92.82	0.7322	83.8
XBM-LLaVA-v1.6-Vicuna-7B	86.70	0.8081	35.7	93.85	0.8032	41.4
XBM-LLaVA-v1.6-Mistral-7B	86.75	0.8086	25.8	92.41	0.8071	27.9
ImageNet	Text Classifier $f_\theta(\boldsymbol{e})$			Multi-modal Classifier $f_\theta(h_\psi(x), \boldsymbol{e})$		
	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)
Frozen BLIP	9.97	0.7732	199.5	56.04	0.7732	199.5
XBM-BLIP	18.26	0.8007	148.1	67.83	0.7920	122.8
Frozen LLaVA-v1.5-LLaMA-7B	64.01	0.7773	236.8	70.21	0.7773	100.8
XBM-LLaVA-v1.5-LLaMA-7B	71.41	0.8008	127.2	72.95	0.7998	82.6
XBM-LLaVA-v1.6-Vicuna-7B	73.73	0.8140	36.74	74.42	0.8037	32.3
XBM-LLaVA-v1.6-Mistral-7B	72.14	0.8037	20.67	74.04	0.8130	21.7

Table 7: Evaluation of cross-attention map of XBMs on ImageNet Segmentation.

	Pixel Acc. (↑)	mIoU (↑)	mAP (↑)
LRP [42](BLIP-ViT)	46.25	29.69	48.51
partial-LRP [43] (BLIP-ViT)	53.59	36.29	65.06
rollout [44] (BLIP-ViT)	52.73	35.81	66.78
Raw Attention (BLIP-ViT)	57.12	39.00	67.92
GradCAM [16] (BLIP-ViT)	61.84	39.68	63.48
Chefer et al. [41] (BLIP-ViT)	59.92	42.30	69.51
XBM-BLIP w/ Fixed Decoder	78.67	57.90	79.72
XBM-BLIP	80.90	60.80	80.18

G Detailed Analysis

In this section, we provide detailed analyses of XBMs. In particular, we assess temperature annealing in the Gumbel softmax sampling (Eq. (6)), the hyperparameter λ in Eq. (1), and the localization ability of the cross-attention heatmaps introduced in Section 2.4.

G.1 Evaluations of Cross-Attention Heatmap

The cross-attention heatmap explanation of XBMs visualizes the local input space regions correlated to the text explanation in the classifier. To assess the validity of XBMs on improving multi-modal understanding, we evaluate the generated heatmaps on the ImageNet segmentation task by following [41] and [45]. That is, we generate the heatmaps on the test set of ImageNet Segmentation [46] and compute the pixel accuracy, mean IoU (mIoU), and mean average precision (mAP) with the ground truth segmentation masks. Through this evaluation, we can evaluate how heatmaps cover the object of target classes in the pixel spaces. Table 8 shows the results. Compared to the frozen BLIP, XBM-BLIP improved all of the segmentation metrics. This means that the training objective of XBMs encourages the multi-modal understanding of target class objects on the models. In Appendix F, we further compare the XBM's heat maps with existing attribution methods, such as GradCAM [16].

Table 8: Evaluation of cross-attention map of XBMs on ImageNet Segmentation.

	Pixel Acc. (↑)	mIoU (↑)	mAP (↑)
Frozen BLIP + Multi-modal Classifier	78.67	57.90	79.72
XBM-BLIP	80.90	60.80	80.18

Table 9: Effects of temperature τ and annealing for Gumbel softmax sampling of XBMs (Car).

	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)
$\tau^{(0)} = 1$	86.60	0.7138	140.8
+ Annealing	87.12	0.7258	127.3
$\tau^{(0)} = 10$	86.71	0.7148	143.9
+ Annealing	88.65	0.7253	133.0
$\tau^{(0)} = 100$	88.03	0.7168	146.2
+ Annealing	88.56	0.7272	143.6

G.2 Evaluations of Temperature Annealing

We introduce the temperature annealing strategy for determining τ in Eq. (6). Here, we evaluate the effects by varying the initial temperature $\tau^{(0)}$ in $\{1, 10, 100\}$. Table 9 shows the test performance and interpretability scores. We tested the cases leveraging a constant temperature $\tau^{(0)}$ and applying exponential temperature annealing, i.e., + Annealing. In the cases of constant temperatures, we confirm that the larger temperatures tend to achieve better target performance but degrade perplexity scores. This is because using a larger temperature increases the entropy of the generative distribution of tokens in the Gumbel softmax sampling, and thus, it slightly loses the naturalness of the generated sentences. On the other hand, applying the temperature annealing improved all scores in all initial temperatures. This implies that, by gradually reducing the temperature, XBMs can try to generate diverse tokens in the early stages of learning, and it narrows down only vocabulary with high likelihood in the later stages while the sentence naturalness is maintained.

G.3 Effects of Hyperparameter λ

The hyperparameter λ in Eq (1) balances the target task training and the regularization to avoid the collapse of text explanations. Table 10 shows the results when varying λ. It demonstrates that the cases of $\lambda > 0$ can avoid the collapse of the interpretability and improve target performance. We see that there is a trade-off between the target test accuracy and the GPT-2 Perplexity scores. In contrast, fortunately, CLIP-Score was less sensitive to the value of $\lambda > 0$, suggesting high-performance XBMs can still generate explanations well-related to inputs. Therefore, we recommend determining λ based on whether fluency or accuracy is a priority according to the application's requirements.

G.4 Explanation Distillation vs. Other Regularization

Here, we evaluate our explanation distillation regularization $\mathcal{R}_{\text{int}}$ through a comparison to another regularization method. We compare our explanation distillation to L2SP [47], which penalizes the model parameters by minimizing the l2 distance from the pre-trained parameters. Table 11 shows the results on the Car dataset. We confirm that our explanation distillation outperforms L2SP in all performance metrics. Our method largely improves clip score, while L2SP degrades it from frozen BLIP. These results suggest that directly regularizing the decoder's output helps XBMs explore the vocabulary needed for a task through classification loss while preserving natural sentences and minimizing the gap in the parameter space, which is harmful to this purpose.

H Broader Impacts

A potential negative effect introduced by our work is that XBMs may output biased explanations if the backbone language model is extremely biased. This can be avoided by purifying the language model with existing debiasing methods such as [48] before training XBMs. Since the target tasks

Table 10: Effects of hyperparameter λ of XBMs (Car).

	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)
$\lambda = 0$	86.59	0.4792	415.3
$\lambda = 0.01$	89.18	0.7158	163.4
$\lambda = 0.1$	89.47	0.7172	145.4
$\lambda = 0.3$	89.09	0.7148	138.9
$\lambda = 0.5$	88.62	0.7167	132.6
$\lambda = 0.7$	87.58	0.7158	131.7
$\lambda = 1.0$	87.59	0.7138	127.3

Table 11: Effects of Regularization in XBMs (Car).

	Test Acc. (↑)	CLIP-Score (↑)	GPT-2 Perplexity (↓)
Frozen BLIP	77.91	0.6091	168.8
Explanation Distillation (Ours)	90.48	0.7173	131.8
L2SP [47]	87.47	0.5059	159.4

handled by XBMs are no different from those in general models, off-the-shelf defence methods may be directly applicable to other risks such as adversarial attacks.

References

[1] 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*, 2020.

[2] Mert Yuksekgonul, Maggie Wang, and James Zou. Post-hoc concept bottleneck models. In *International Conference on Learning Representations*, 2023.

[3] Tuomas Oikarinen, Subhro Das, Lam M Nguyen, and Tsui-Wei Weng. Label-free concept bottleneck models. In *International Conference on Learning Representations*, 2023.

[4] Vikram V Ramaswamy, Sunnie SY Kim, Ruth Fong, and Olga Russakovsky. Overlooked factors in concept-based explanations: Dataset choice, concept learnability, and human capability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

[5] 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*, 2023.

[6] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International conference on machine learning*, 2022.

[7] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, 2023.

[8] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, 2023.

[9] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[10] Yoon Kim and Alexander M Rush. Sequence-level knowledge distillation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2016.

[11] Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. In *International Conference on Learning Representation*, 2017.

[12] P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, and P. Perona. Caltech-UCSD Birds 200. Technical report, California Institute of Technology, 2010.

[13] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3), 2015.

[14] Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. Learning deep representations of fine-grained visual descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[15] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016.

[16] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.

[17] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, 2018.

[18] Mateo Espinosa Zarlenga, Pietro Barbiero, Gabriele Ciravegna, Giuseppe Marra, Francesco Giannini, Michelangelo Diligenti, Frederic Precioso, Stefano Melacci, Adrian Weller, Pietro Lio, et al. Concept embedding models. In *Advances in Neural Information Processing Systems*, 2022.

[19] Mazda Moayeri, Keivan Rezaei, Maziar Sanjabi, and Soheil Feizi. Text-to-concept (and back) via cross-model alignment. In *International Conference on Machine Learning*, 2023.

[20] Xinyue Xu, Yi Qin, Lu Mi, Hao Wang, and Xiaomeng Li. Energy-based concept bottleneck models: unifying prediction, concept intervention, and conditional interpretations. In *International Conference on Learning Representations*, 2024.

[21] 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*, 2020.

[22] Robyn Speer, Joshua Chin, and Catherine Havasi. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2017.

[23] Qihan Huang, Jie Song, Jingwen Hu, Haofei Zhang, Yong Wang, and Mingli Song. On the concept trustworthiness in concept bottleneck models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 21161–21168, 2024.

[24] Lisa Anne Hendricks, Zeynep Akata, Marcus Rohrbach, Jeff Donahue, Bernt Schiele, and Trevor Darrell. Generating visual explanations. In *Proceedings of the European Conference on Computer Vision*, 2016.

[25] Kosuke Nishida, Kyosuke Nishida, and Shuichi Nishioka. Improving few-shot image classification using machine-and user-generated natural language descriptions. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1421–1430, 2022.

[26] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[27] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Conference of the North American Chapter of the Association for Computational Linguistics*, 2019.

[28] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[29] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality, March 2023.

[30] AQ Jiang, A Sablayrolles, A Mensch, C Bamford, DS Chaplot, D de las Casas, F Bressand, G Lengyel, G Lample, L Saulnier, et al. Mistral 7b (2023). *arXiv preprint arXiv:2310.06825*, 2023.

[31] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, 2020.

[32] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. *arXiv*, 2013.

[33] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition*, Sydney, Australia, 2013.

[34] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[35] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

[36] 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*. PMLR, 2021.

[37] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7514–7528, 2021.

[38] David Chan, Suzanne Petryk, Joseph Gonzalez, Trevor Darrell, and John Canny. Clair: Evaluating image captions with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

[39] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[40] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

[41] Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 782–791, 2021.

[42] Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, Klaus-Robert Müller, and Wojciech Samek. Layer-wise relevance propagation for neural networks with local renormalization layers. In *International Conference on Artificial Neural Networks*. Springer, 2016.

[43] Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Annual Meeting of the Association for Computational Linguistics*, 2019.

[44] Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Annual Meeting of the Association for Computational Linguistics*, 2020.

[45] Yossi Gandelsman, Alexei A. Efros, and Jacob Steinhardt. Interpreting CLIP's image representation via text-based decomposition. In *International Conference on Learning Representations*, 2024.

[46] Matthieu Guillaumin, Daniel Küttel, and Vittorio Ferrari. Imagenet auto-annotation with segmentation propagation. *International Journal of Computer Vision*, 110:328–348, 2014.

[47] Xuhong Li, Yves Grandvalet, and Franck Davoine. Explicit inductive bias for transfer learning with convolutional networks. In *International Conference on Machine Learning*, 2018.

[48] Masahiro Kaneko and Danushka Bollegala. Debiasing pre-trained contextualised embeddings. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1256–1266, 2021.