GLOBAL CONCEPT SPARSITY FOR UNSUPERVISED CONCEPT BOTTLENECK MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Concept bottleneck models (CBMs) aim to make predictions interpretable through human-understandable concepts, yet unsupervised approaches for automatically discovering concepts produce large, redundant concept vocabularies. While existing sparsity mechanisms constrain which concepts are active for particular inputs or classes, in practice they allow for redundant concepts which are used only for a small number of classes. As a result, locally sparse models may still rely on an effectively large global concept vocabulary, undermining the compactness and inspectability that concept bottlenecks are intended to provide. To address this, we propose Global Concept Gating (GCG), a global concept-gating mechanism that can be easily used within *any* unsupervised CBM. GCG learns a sparse mask shared across all inputs and classes, jointly selecting a compact task-specific concept set and training the downstream classifier. Across multiple concept bottleneck architectures and image classification tasks, GCG substantially reduces concept set size while largely preserving predictive accuracy. Further analysis shows that the retained concepts are more relevant to prediction, better semantically grounded, and less susceptible to information leakage. Our results highlight that interpretable concept-based models require not only meaningful individual concepts, but concept spaces appropriately constrained to their tasks.

1 INTRODUCTION

Deep neural networks (DNNs) have achieved promising performance across a variety of tasks, from language reasoning (Brown et al., 2020; Wei et al., 2023) to computer vision (Radford et al., 2021; Dosovitskiy et al., 2021). However, these models often process inputs in a *black box* manner, learning a compressed, often uninterpretable representation of data. This poses a critical difficulty in understanding a model's actions (Lipton, 2016; Rudin, 2019) and severely constrains our ability to monitor against potentially harmful failure modes (Amodei et al., 2016). In recent years, *Concept Bottleneck Models* (CBMs) (Koh et al., 2020), and their many variants (Espinosa Zarlenga et al., 2022; Yuksekgonul et al., 2023; Oikarinen et al., 2023; Almudévar et al., 2026; Knab et al., 2026), have emerged as an alternative DNN family to mitigate opacity and improve model interpretability. These models utilise an intermediate layer of human-interpretable concepts between the input and the final prediction, mapping inputs to semantic concepts from which the output is produced, and enabling users to more easily inspect and understand the model's decision-making process.

However, CBMs are often dependent on manually annotated concept datasets, which are labour-intensive and difficult to obtain (Yang et al., 2023; Oikarinen et al., 2023). To overcome this limitation, recent research has explored unsupervised concept discovery (Yang et al., 2023; Srivastava et al., 2025; Rao et al., 2024). Yet, while these approaches reduce concept annotation costs, they often introduce a large concept vocabulary, which can hinder the model's interpretability and auditability (Ramaswamy et al., 2023). Thus, prior works have incorporated concept-sparsity penalties to limit the *effective* size of the concept set (Oikarinen et al., 2023; Schrodi et al., 2024; Rao et al., 2024). However, such penalties are often applied at a local level (e.g., per input or class), rather than to the global concept set, meaning they fail to provide *guarantees* on the global subset of concepts the model may use in its reasoning.

As a result, models whose explanations appear highly sparse by standard metrics (e.g., concepts per image) may still require understanding the full concept vocabulary to audit the model's global

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

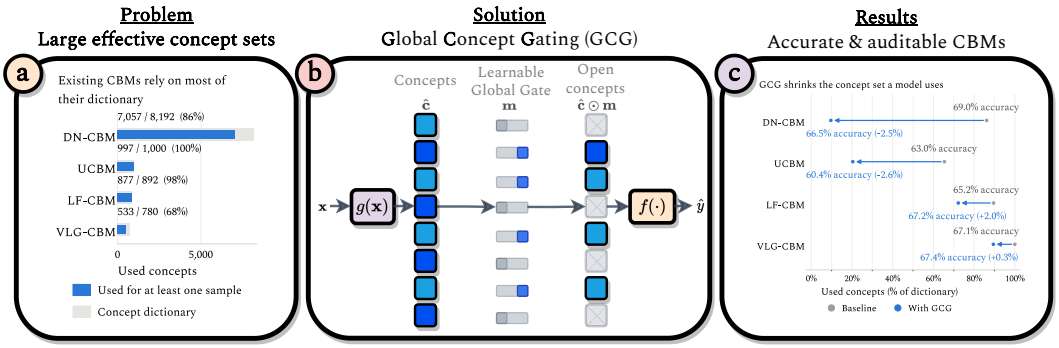


Figure 1: **(a)** Although several unsupervised CBMs include sparsity constraints, they effectively use a large portion of their entire concept vocabulary across samples. This severely limits our ability to audit the CBMs' global behaviour. **(b)** Global Concept Gating (GCG) addresses this by introducing a simple but general learnable gate that selects a sparse, task-relevant set of informative concepts. **(c)** GCG greatly reduces the global effective concept set w/o significant effects on model accuracy.

behaviour. Hence, our practical understanding of these models is constrained by our inability to reason about large sets of concepts (Miller, 1956; Lage et al., 2019), creating an opportunity for concept-filtering mechanisms with stronger global guarantees.

To address this, we propose *Global Concept Gating* (GCG) (Figure 1), a simple gating mechanism that learns a sparse, task-specific concept bottleneck on top of *any* pretrained unsupervised CBM. Rather than modifying the underlying concept representation, the gate identifies and globally suppresses concepts that are unnecessary for a downstream task, allowing the classifier to operate using only a compact subset of the original concept vocabulary.

We evaluate our approach on three image classification benchmarks spanning both fine-grained and diverse object recognition tasks. Our experiments show that our learned gates can retain only a small fraction of discovered concepts while largely preserving, and in some cases improving, classification accuracy. Beyond predictive performance, we demonstrate that the retained concepts more closely match ground-truth semantic attributes and reduce information leakage. Together, these results suggest that compact task-specific concept sets can substantially improve the global auditability of CBMs while maintaining competitive performance.

Our contributions can therefore be summarised as:

- We propose a sparse GCG mechanism that can be applied to *any* unsupervised CBM to significantly reduce the global concept set. Our approach learns task-specific sparse concept subsets without requiring concept supervision or modifying the underlying concept discovery process.
- We show that CBMs trained with GCG retain only a small fraction of the discovered concepts while maintaining competitive classification accuracy across multiple tasks. The learned mask guarantees that excluded concepts cannot contribute to predictions.
- We demonstrate that GCG improves explanation interpretability by increasing concept-grounding accuracy and reducing information leakage.

2 Background and Related Work

Concept Bottleneck Models Concept Bottleneck Models (CBMs) (Koh et al., 2020) are interpretable deep learning models that first map inputs to human-interpretable concepts and then combine them linearly to make a final prediction. CBMs consist of two stages: first, a concept bottleneck layer $g : \mathbb{R}^d \to \mathbb{R}^k$ maps an input $\boldsymbol{x} \in \mathbb{R}^d$ to a vector of k concept activations, represented as $\hat{\boldsymbol{c}} = g(\boldsymbol{x})$. Then, a linear probe $f : \mathbb{R}^k \to \mathbb{R}^l$ maps concept activations to a downstream task prediction $\hat{y} = f(g(\boldsymbol{x}))$. A CBM's parameters are learnt from a training set $\{(\boldsymbol{x}^{(i)}, y^{(i)}, \boldsymbol{c}^{(i)})\}_{i=1}^n$, where $\boldsymbol{x}^{(i)} \in \mathbb{R}^d$ is a vector of a sample's features, $y^{(i)}$ is a downstream task label, and $\boldsymbol{c}^{(i)} \in \mathbb{R}^k$ is a vector of concept "scores". CBMs therefore require explicit concept annotations, limiting their scalability and restricting the concept set to a predefined vocabulary (Oikarinen et al., 2023).

Subsequent work has addressed several limitations of the original CBM formulation. Concept Embedding Models (CEMs) (Espinosa Zarlenga et al., 2022) replace scalar concept activations with high-dimensional concept embeddings to mitigate performance losses when the concept set is incomplete. Post-hoc CBMs (Yuksekgonul et al., 2023) project the representations of an existing pretrained model onto a concept subspace, thus allowing the model to transfer concepts from other sources without retraining. Other variants address the reliability of the bottleneck itself (Sheth & Ebrahimi Kahou, 2023; Espinosa Zarlenga et al., 2025; Almudévar et al., 2026), propose probabilistic formulations of the bottleneck (Kim et al., 2023; Xu et al., 2024), or explicitly model inter-concept dependencies (Havasi et al., 2022; Vandenhirtz et al., 2024; Dominici et al., 2025).

Nevertheless, all CBM variants discussed so far require *some* training concept annotations. Yet obtaining such a set is costly: domain experts need to annotate each sample with a predefined set of concepts, restricting the use of CBMs to datasets where such annotation is feasible (Yuksekgonul et al., 2023; Oikarinen et al., 2023; Skirzynski et al., 2026) and where annotations can be obtained accurately (Ramaswamy et al., 2023; Collins et al., 2023; Penaloza et al., 2025; Debole et al., 2026). The choice of concept set also imposes a ceiling on performance, since a task cannot be solved with concepts the vocabulary does not contain, and an incomplete set forces the model to either sacrifice accuracy or *leak* missing information in the concept activations (Mahinpei et al., 2021).

Unsupervised Concept Discovery To address the difficulty of obtaining expert concept annotations, recent work has explored unsupervised concept discovery. *Unsupervised CBMs* discover concepts by (1) prompting a language model (Yang et al., 2023; Oikarinen et al., 2023), (2) using unsupervised dictionary learning (Schrodi et al., 2024), (3) exploiting open-vocabulary object detection (Srivastava et al., 2025), or (4) training a sparse autoencoder (Rao et al., 2024). Although these methods eliminate the need for a manually annotated concept set, they produce large, often redundant, concept sets (Oikarinen et al., 2023; Schrodi et al., 2024). Considering that humans struggle to reason about interactions between large concept sets (Miller, 1956; Cowan, 2001; Lage et al., 2019), large global concept sets severely restrict the models whose *global* behaviour we can practically audit. Therefore, unless the overall concept set has guaranteed size limits, unsupervised CBMs risk undermining the global interpretability that motivates concept bottlenecks in the first place.

Local Concept Sparsity To reduce the effective concept set size, prior works have incorporated concept sparsity penalties through two broad mechanisms. The first is filtering, where candidate concepts are pruned before training (e.g., discarding those with low similarity to any class or high similarity to another concept) (Oikarinen et al., 2023; Yang et al., 2023). The second is weight- or activation-level sparsity applied during training. For instance, Label-free CBMs (LF-CBMs) (Oikarinen et al., 2023) train their final layer with a proximal solver that drives per-class concept weights to zero, and Vision-Language-Guided CBMs (Srivastava et al., 2025) directly control the average number of non-zero weights per class. Discover-then-name CBMs (DN-CBMs) (Rao et al., 2024) inherit per-image sparsity from their sparse autoencoder, and input-dependent gating mechanisms, such as Concept Discovery Models (CDMs) (Panousis et al., 2023) and Unsupervised Concept Bottleneck Models (UCBMs) (Schrodi et al., 2024), impose soft constraints on how many concepts activate for any single input by penalising dense per-sample concept activations. Nevertheless, the sparsity penalties applied during training primarily constrain concept use per input or class (local sparsity), rather than across the task as a whole (global sparsity). Because active concepts can vary between classes and samples, the true *effective concept set*, that is, the set of concepts the model may use for any sample, can remain far larger than the concept set used for any individual input or class (as we show later in Section 4.2). More importantly, excluding a concept for a particular input or class does not guarantee that it is excluded across the task as a whole. Therefore, there is a clear need for global sparsity methods with stronger guarantees.

Global Feature Selection Deriving stronger guarantees on the concept set an unsupervised CBM can use closely mirrors global feature selection in classical machine learning (Tibshirani, 1996; Breiman, 2001; Guyon & Elisseeff, 2003; Chandrashekar & Sahin, 2014). These approaches share a common objective: reducing a large feature space to a smaller, task-relevant subset that can be used consistently across predictions. Hence, in this work we apply this principle to unsupervised CBMs by framing global concept sparsity as a feature selection problem on concepts. Applying global feature selection to concept-based reasoning offers a natural way to ask which concepts are task-relevant and constrain the number of concepts so that auditability is still possible.

3 GLOBAL CONCEPT GATING (GCG)

Gated Probe We propose Global Concept Gating (GCG), a simple yet efficient *global* concept-filtering extension for *any* unsupervised CBM. Given any unsupervised CBM with concept encoder $g(\boldsymbol{x}; \mathbf{W}_g, \mathbf{b}_g)$ and task predictor $f(\hat{\boldsymbol{c}}; \mathbf{W}_f, \mathbf{b}_f)$, we draw inspiration from traditional differentiable global feature selection methods (Li et al., 2016; Yamada et al., 2020) and introduce a *learnable* gate vector $\mu \in \mathbb{R}^k$ that assigns each concept j a soft gate value $\sigma(\mu_j) \in [0, 1]$, where σ is the sigmoid.

$$\hat{\boldsymbol{y}} = \mathbf{W}_f\big(\hat{\boldsymbol{c}} \odot \boldsymbol{m}\big) + \boldsymbol{b}_f, \qquad \text{where } m_j = \mathbb{1}\left[\sigma(\mu_j) > \tfrac{1}{2}\right]. \tag{1}$$

Here, $\hat{\boldsymbol{c}} = g(\boldsymbol{x})$ is the CBM's original concept bottleneck, $\odot$ denotes element-wise multiplication, and $\mathbf{W}_f, \mathbf{b}_f$ are the weights and bias of the task linear probe. The gate is applied to the output of each model's own concept layer, leaving the concept discovery process, the concept layer itself, and any existing regularisation unchanged. Since the thresholded gates have zero gradients almost everywhere, the forward pass uses m_j, and the backward pass uses the gradient of the soft gate $\sigma(\mu_j)$ instead. We provide a visualisation of the model in Figure 1.

Training We then learn μ jointly with the task probe using an additional penalty on the soft gates:

$$\mathcal{L}_{\text{gated}} = \mathcal{L}_{\text{base}} + \lambda_{\text{gate}} \sum_{j=1}^{k} \sigma(\mu_j), \tag{2}$$

where $\mathcal{L}_{\text{base}}$ is the baseline CBM probe's training objective. Intuitively, $\sigma(\mu_j)$ is a soft indicator of whether concept j is available, so $\sum_j \sigma(\mu_j)$ approximates the number of open gates $\sum_j m_j$. We apply the penalty to the soft value rather than to m_j, because it must carry a gradient, allowing standard gradient-based methods to easily optimise the sparsity objective.

Mask and Refitting GCG operates on a CBM with a pretrained concept encoder g, held fixed throughout; only the gate μ and task probe $(\mathbf{W}_f, \mathbf{b}_f)$ are optimised, to retain the concepts' original meaning. We first train the gate and probe jointly from scratch (Equation 2), then fix the retained concept set $\mathcal{K} = \{\, j : \sigma(\mu_j) > 0.5 \,\}$, which we define as *open concepts*. We discard all remaining concepts and retrain only the task probe weights on the activations $\{\hat{\boldsymbol{c}}_j\}_{j \in \mathcal{K}}$. This yields a CBM $f(g(\boldsymbol{x}))$ with exactly $|\mathcal{K}|$ concepts, with accuracy preserved from the jointly trained gated model (Appendix A.3). All accuracies and concept counts we report for GCG are those of this refit model.

Initialisation We initialise all gate logits to $+2$ so that $\sigma(\mu_j) \approx 0.88$, opening them for all concepts. This ensures the model can access the full concept set at the start and identify relevant concepts. We find that initialising with negative gate logits leads to poorer performance in our initialisation ablation (Appendix A.1). We therefore use an all-open initialisation.

4 EXPERIMENTS

We evaluate GCG by studying the following research questions:

RQ1. *Are existing unsupervised CBMs globally sparse?*
RQ2. *Does GCG lead to global concept sparsity without a significant drop in task performance?*
RQ3. *How does the concept set size selected by GCG influence task performance?*
RQ4. *Does GCG's global sparsity improve model auditability?*

4.1 SETUP

Datasets We evaluate GCG using three standard vision tasks: `CIFAR-100` (Krizhevsky, 2009), `Places-365` (Zhou et al., 2016), and `CUB` (Wah et al., 2011). `CIFAR-100` comprises 60,000 colour images across 100 classes, `Places-365` consists of 1.8 million images across 365 scene categories, and `CUB` contains 11,788 images of 200 bird species with 312 annotated high-level binary attributes. Evaluating GCG on datasets without concept annotations, such as `CIFAR-100` and `Places-365`, lets us study how GCG generalises across tasks, while the attribute annotations for `CUB` let us evaluate the nature of concepts GCG selects.

Baseline Models We compare several unsupervised CBMs with and without GCG. These baselines include: (1) DN-CBM (Rao et al., 2024), which discovers a task-agnostic concept vocabulary by

Table 1: Number of concepts making a non-zero contribution to the predicted class for each CBM. *Per image* reports the average number of contributing concepts across test images, *Across Dataset* reports the size of the union of contributing concepts over the test set, with the fraction of the full concept vocabulary in parentheses, and *Total* reports the vocabulary size. Local sparsity does not imply global sparsity: in 11 of 12 settings, models use $\geq 62\%$ of their vocabulary across the test set.

	CIFAR-100			CUB			Places-365		
Model	Per image	Across Dataset	Total	Per image	Across Dataset	Total	Per image	Across Dataset	Total
DN-CBM	$636.2_{\pm 88.4}$	7057 (86%)	8192	$579.3_{\pm 55.0}$	5047 (62%)	8192	$4.3_{\pm 2.0}$	5947 (73%)	8192
LF-CBM	$87.5_{\pm 22.4}$	877 (98%)	892	$92.8_{\pm 15.4}$	351 (94%)	370	$26.1_{\pm 8.3}$	1952 (90%)	2161
UCBM	$229.0_{\pm 103.6}$	981 (98%)	1000	$27.0_{\pm 6.3}$	200 (100%)	200	$39.0_{\pm 7.1}$	213 (6%)	3650
VLG-CBM	$6.8_{\pm 3.3}$	533 (68%)	780	$4.7_{\pm 1.3}$	617 (92%)	671	$5.4_{\pm 2.2}$	1579 (72%)	2186

training a sparse autoencoder on CLIP (Radford et al., 2021) embeddings of CC3M (Sharma et al., 2018); (2) LF-CBM (Oikarinen et al., 2023), which generates per-class candidate concepts via an LLM and learns a projection onto their CLIP text embeddings; (3) UCBM (Schrodi et al., 2024), which discovers concepts via dictionary learning over a backbone's activations; and (4) VLG-CBM (Srivastava et al., 2025), which supervises concepts using open-vocabulary object detection.

These models differ not only in concept discovery, but in concept sparsity enforcement: LF-CBM trains its final layer with GLM-SAGA, UCBM applies a learned per-concept offset with elastic-net regularisation, and VLG-CBM controls the average number of non-zero weights per class. Thus, evaluating our GCG mechanism on these approaches tests whether global redundancy persists in concept spaces produced by different discovery mechanisms and *local* sparsity constraints.

In our experiments, we use each baseline's published backbone, concept sets, and hyperparameters. To ensure a fair evaluation, we train the gate and task probe while keeping the pretrained concept encoder fixed and retaining each baseline's published non-GCG hyperparameters. We include further training details in Appendix A.2.

Metrics For each run, we report the following metrics: (1) *Task Accuracy*, standard for measuring model performance; (2) *Number of Concepts*, defined as the number of non-zero weights across the task in the final linear layer. Unlike contribution-based counts in Section 4.2, this is an upper bound on the concepts the model can use across the task, and hence the vocabulary an auditor must understand to reason about the model's explanations. (3) *Concept-Efficient Accuracy* (Zhao et al., 2025), defined as $\text{CEA} = \frac{\text{ACC}}{(\log_b m)^\beta}$, where ACC represents accuracy, m is the number of concepts used, β is the concept complexity penalty, and $b = \lceil \log_2 l \rceil$, where l is the number of output classes. CEA jointly rewards predictive performance and concept compactness, making it suitable for selecting models that balance accuracy and concept efficiency. We use this metric for λ_{gate} selection. Finally, we measure (4) *Jaccard index* for concept accuracy, when concept labels are available.

4.2 How globally sparse are the baseline models? (**RQ1**)

To motivate the need for GCG, we first show that existing unsupervised CBMs lack global concept sparsity. We do this by counting each model's concepts with non-zero contribution $|\mathbf{W}_{\hat{y},j} \times \hat{\boldsymbol{c}}_j(\boldsymbol{x})|$ to the predicted-class logit. We then count how many concepts are used across the dataset by taking the union of per-image concept sets and comparing it to the total vocabulary size. Unlike prior studies that report average concept use per sample, we explicitly measure the total set of concepts used across the dataset, which estimates the size of the concept set an expert must understand in practice to properly audit the CBM's global behaviour.

Our analysis, summarised in Table 1, shows that, across nearly all models and tasks, the number of concepts *globally* used across the dataset far exceeds the number used for any individual prediction. Here, DN-CBM on Places-365 is the starkest case: a typical prediction relies on 4.3 concepts, yet 5,947 distinct concepts (73% of the vocabulary) are used somewhere across the test set. The pattern holds regardless of how each method enforces sparsity, whether through proximal solvers on the classifier weights, per-class concept budgets, or per-image activation sparsity. The one exception is UCBM on Places-365, where the global set remains small. These results confirm that the sparsity these methods provide is predominantly local, and that a practitioner auditing a model's

Table 2: Task accuracy and number of global concepts used for each CBM with (✓) and without (✗) GCG. Concept counts for ungated models are the *effective* number of concepts used across the task. We use bold to indicate the best metric value between gated and ungated model variants. GCG reduces the global concept count in every setting, improves accuracy in half of the settings, and incurs modest losses elsewhere.

		CIFAR-100		CUB		Places-365	
Model	GCG	Acc (%, ↑)	Concepts (↓)	Acc (%, ↑)	Concepts (↓)	Acc (%, ↑)	Concepts (↓)
DN-CBM	✗	$\mathbf{68.98}_{\pm 0.10}$	$7049.80_{\pm 0.00}$	$67.72_{\pm 0.05}$	$5138.00_{\pm 0.00}$	$47.65_{\pm 0.02}$	$8192.00_{\pm 0.00}$
	✓	$67.12_{\pm 0.04}$	$\mathbf{791.60}_{\pm 2.97}$	$\mathbf{68.18}_{\pm 0.05}$	$\mathbf{242.20}_{\pm 0.40}$	$\mathbf{52.90}_{\pm 0.08}$	$\mathbf{1694.80}_{\pm 13.80}$
LF-CBM	✗	$65.16_{\pm 0.01}$	$741.40_{\pm 0.49}$	$\mathbf{64.20}_{\pm 0.17}$	$351.00_{\pm 0.00}$	$\mathbf{43.57}_{\pm 0.02}$	$2145.00_{\pm 23.00}$
	✓	$\mathbf{67.31}_{\pm 0.11}$	$\mathbf{598.80}_{\pm 0.40}$	$63.77_{\pm 0.08}$	$\mathbf{267.00}_{\pm 1.30}$	$41.06_{\pm 0.49}$	$\mathbf{166.80}_{\pm 3.60}$
UCBM	✗	$\mathbf{62.96}_{\pm 0.09}$	$654.20_{\pm 5.19}$	$74.39_{\pm 0.07}$	$200.00_{\pm 0.00}$	$47.78_{\pm 0.06}$	$1776.00_{\pm 13.94}$
	✓	$60.41_{\pm 0.13}$	$\mathbf{204.20}_{\pm 1.17}$	$\mathbf{74.57}_{\pm 0.26}$	$\mathbf{173.60}_{\pm 3.10}$	$\mathbf{50.79}_{\pm 0.14}$	$\mathbf{430.20}_{\pm 4.12}$
VLG-CBM	✗	$67.12_{\pm 0.05}$	$780.00_{\pm 0.00}$	$\mathbf{75.94}_{\pm 0.00}$	$635.00_{\pm 0.00}$	$\mathbf{45.47}_{\pm 0.00}$	$2186.00_{\pm 0.00}$
	✓	$\mathbf{67.39}_{\pm 0.14}$	$\mathbf{697.60}_{\pm 1.50}$	$72.33_{\pm 0.20}$	$\mathbf{38.60}_{\pm 1.36}$	$43.98_{\pm 0.07}$	$\mathbf{107.80}_{\pm 0.40}$

global behaviour must generally understand most of its concept vocabulary. Note that these counts only capture concepts that contribute on the test set and therefore undercount the global concepts a model can use. In Section 4.3, we therefore report the number of concepts with non-zero weights, which bounds concept use across all inputs.

4.3 Does GCG yield global concept sparsity and at what cost? (**RQ2**)

Next, we study how GCG affects a CBM's global concept set and its task accuracy by applying GCG to all baselines across three tasks. We select λ_{gate} by maximising the validation CEA over eight values between 10^{-5} and 10^{-2}, setting $\beta = 0.25$ like Zhao et al. (2025). We then train the probes for 100 epochs, with a learning rate of 0.001.

We summarise the results in Table 2. We observe that GCG reduces the concept-set size across all tasks (e.g., DN-CBM uses 4.7% of its original effective concept set in CUB), while maintaining, or sometimes improving, task accuracy. For instance, in Places-365 for DN-CBM, task accuracy improves ($1.11\times$ higher) while using a much smaller concept set ($4.84\times$ smaller). For VLG-CBM, we observe a moderate drop in performance on CUB (3.61pp drop) and Places-365 (1.49pp). However, this comes with a significant decrease in the concept set (93.9% reduction in CUB and 95.07% reduction in Places-365). These results indicate that global concept redundancy is present across models and that GCG can exploit it with little or no loss in task accuracy.

4.4 How does the concept set size influence performance? (**RQ3**)

Having observed significant redundancy in the concept vocabularies discovered by unsupervised CBMs, we next examine where this redundancy arises and why GCG can maintain high task accuracy with significantly fewer concepts. In this subsection and the next, we focus on how GCG selects concepts for DN-CBMs, allowing us to study concept masks within a fixed method. Moreover, because DN-CBM discovers a large concept vocabulary from a task-independent corpus, it provides a good basis for understanding how GCG selects task-relevant concepts.

We first empirically study the relationship between concept sparsity and downstream performance. For this, we train the gated probes across eight values of λ_{gate} between 10^{-5} and 10^{-2}. For each setting, we record the test accuracy and the number of open gates across five random seeds. To assess whether the observed performance changes are attributable to targeted concept set reductions, we evaluate the accuracy of a linear probe trained on an equally sized random concept subset.

Our ablation results in Figure 2 show a consistent pattern across tasks: moderate pruning has little effect on accuracy, but performance eventually declines as more concepts are removed. In CUB, pruning initially improves accuracy from $67.72_{\pm 0.05}\%$ to $70.47_{\pm 0.05}\%$ at $\lambda_{\text{gate}} = 0.001$. In Places-365 and CIFAR-100, accuracy instead remains near baseline until $\lambda_{\text{gate}} = 0.0001$, after which it decreases steadily. This difference may reflect both the regularising effect of feature selec-

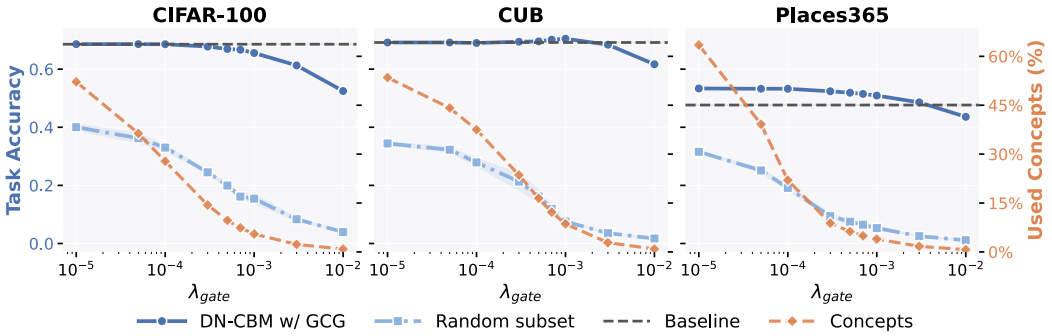


Figure 2: Effect of λ_{gate} on task accuracy and the number of open gates for DN-CBM with GCG. Small values of λ_{gate} generally preserve accuracy, while sufficiently large values reduce it. DN-CBM trained on an equally sized random set of concepts (*Random Subset*) substantially underperforms DN-CBM with GCG, showing that which concepts are kept matters, not just how many.

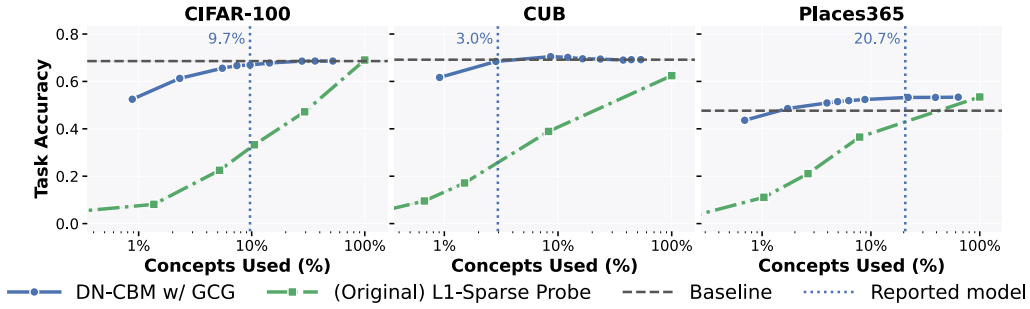


Figure 3: DN-CBM performance as the number of selected concepts varies. We control concept set size through λ_{gate} for GCG (*DN-CBM w/ GCG*) and an ℓ_1 penalty for the baseline (*(Original) L1-Sparse Probe*). *Reported Model* marks the concept count used by GCG in Section 4.3. At matched concept counts, GCG retains substantially more accuracy than DN-CBM's ℓ_1 sparsity as the concept budget shrinks.

tion (Tibshirani, 1996) and differences in concept diversity: `CIFAR-100` and `Places-365` likely require broader subsets of the DN-CBM vocabulary than fine-grained `CUB` classification. Importantly, the random baseline significantly underperforms the gated model across all tasks, indicating that GCG selects a reduced but informative concept set rather than merely reducing dimensionality.

Comparison with weight-level sparsity The advantage of GCG becomes clearer when compared with the ℓ_1 sparsity already used by DN-CBM. To see this, we train sparse probes across a range of DN-CBM regularisation strengths and compare them with GCG probes at matched numbers of active concepts. Our results in Figure 3 suggest that as the number of concepts decreases, DN-CBM's native ℓ_1-sparse probes lose predictive performance substantially faster than GCG's probes. In contrast, the gated probe maintains near-baseline accuracy with only a small subset of the concept vocabulary. Thus, although both mechanisms induce sparsity, GCG is substantially more effective at identifying a compact set of concepts that remains useful for prediction.

4.5 Does reducing the concept set improve auditability? (**RQ4**)

Although GCG incurs a small reduction in classification accuracy on some tasks, its primary objective is to improve DN-CBM auditability by identifying a compact set of task-relevant concepts. We therefore evaluate whether the retained concepts produce better semantically grounded explanations and whether reducing the concept bottleneck mitigates information leakage.

Are selected concepts more semantically grounded? Concepts that better align with how experts reason about the same task are more likely to be useful when auditing a model's global behaviour.

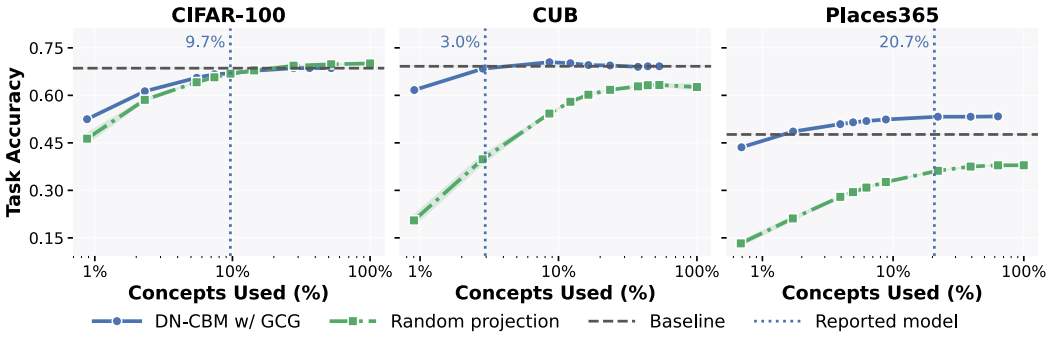


Figure 4: Probe performance with random projections versus learned concepts across bottleneck dimensions. *Reported model* marks the concept count selected by GCG in Section 4.3. On `CIFAR-100` and `CUB`, random projections match learned concepts at high dimensionality, indicating leakage. At the concept counts selected by GCG, learned concepts clearly outperform them.

As such, we explore whether GCG's selected concepts align more closely with attributes deemed important for a task. For this, we measure the concept accuracy of DN-CBM with GCG on `CUB`, where we have access to ground-truth attributes important for determining each sample's label.

First, we preprocess ground-truth attributes in `CUB` to retain the underlying specific semantic characteristic (e.g., `has_wing_color::olive` $\rightarrow$ `olive`). We do this because DN-CBM concepts do not tend to encode the localised distinctions present in the ground-truth attributes. We then map concepts retained by DN-CBM with GCG to the attribute group with the highest cosine similarity, accepting an attribute-concept match only when its similarity exceeds the average similarity across attribute-concept pairs. This process mirrors the concept accuracy study from Rao et al. (2024). Finally, we collect the model's concept predictions on the test data, map them to attributes, and calculate the Jaccard index between the mapped attribute set and `CUB`'s ground-truth attributes.

Our results are as follows: the baseline DN-CBM achieves a Jaccard index of 0.2695, while the DN-CBM with GCG achieves a Jaccard index of 0.2982 (higher is better). We test the significance of this difference at the image level: across 5,794 test images, the paired improvement is $+0.029$ (bootstrap 95% CI $[+0.024, +0.033]$; paired t-test $p < 10^{-33}$; Wilcoxon signed-rank $p < 10^{-36}$). To test whether this improvement is due simply to using fewer concepts, we also compare the GCG model with a random subset of concepts of the same size. The random control achieves a Jaccard index of 0.0216, substantially below both. Together, these results indicate that GCG selects concepts aligned with semantic ground-truth attributes, rather than merely reducing the concept count. Hence, experts may more easily audit CBMs trained with GCG than those without it.

Information leakage and interpretability While semantic grounding is important for auditability, it alone does not establish that the model's predictions are attributable to the concepts it predicted. Prior work has shown that CBMs are susceptible to *information leakage*, whereby a high-dimensional bottleneck can allow a classifier to exploit information beyond the concepts' intended semantics to achieve high accuracy (Mahinpei et al., 2021; Marconato et al., 2023; Parisini et al., 2026; Espinosa Zarlenga, 2026). Consequently, strong model performance does not necessarily translate to faithful concept-based explanations.

To determine whether GCG's improvements stem from leakage, we compare a DN-CBM trained with GCG against a model using a randomly initialised projection layer of the same dimensionality. If the random projection achieves comparable or higher accuracy, this suggests that similar performance is achievable without explicitly identified semantic concepts.

Our results, summarised in Figure 4, show that across `CIFAR-100` and `CUB`, random projection initially matches or outperforms the full 8,192-concept bottleneck, but deteriorates more rapidly as concepts are removed. This is consistent with many small non-zero weights collectively encoding useful information beyond concept semantics. In contrast, for `Places-365`, random projection remains below the learned concepts at all tested dimensions, with the gated model using only $\sim 21\%$ of the vocabulary even exceeding the full-dimensional random projection. This is consistent with

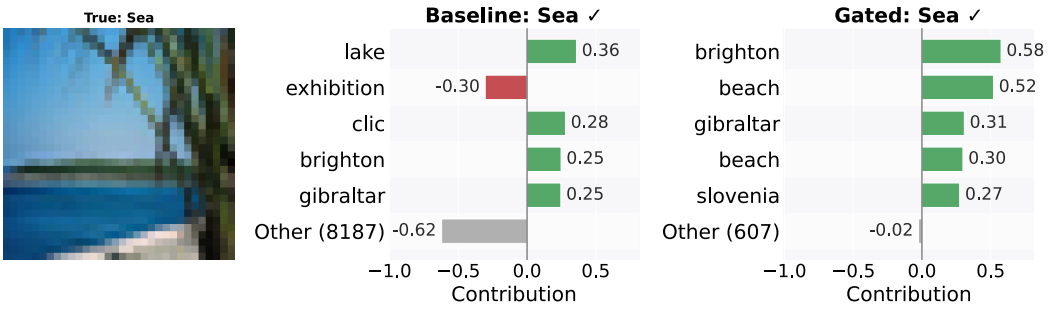


Figure 5: Explanation for a `CIFAR-100` sample before and after GCG. Bars show each concept's contribution $\mathbf{W}_{\hat{y},j} \cdot \hat{c}_j(\boldsymbol{x})$ and the grey bar aggregates all remaining concepts. GCG removes unrelated concepts (e.g., exhibition, clic) and concentrates net support in the five shown, yielding a compact explanation of the correct prediction. We provide more examples in Appendix A.4.

the dimensionality required for random projections to approach an optimal linear classifier increasing with task complexity (Srivastava et al., 2025). Thus, we find evidence of dimensionality-driven information leakage in `CIFAR-100` and `CUB`, but not within our tested range for `Places-365`. Importantly, GCG selects sets well below the crossover point on all three datasets, producing bottlenecks whose explanations more closely reflect image content and the model's decision process.

Qualitative Analysis Figure 5 shows an example explanation generated by the original DN-CBM and DN-CBM with GCG. We present the two models for the same input, allowing us to compare the concepts used to support the prediction before and after concept selection. Specifically, the baseline's top concepts include `exhibition` and `clic`, which bear no relation to the image, while the thousands of concepts outside the explanation exert a substantial net pull against the correct class. Gating removes these irrelevant concepts, while concepts outside the five shown make a near zero-net contribution to the prediction. This yields a more compact explanation that may help users understand the prediction and, where needed, debug the model.

5 Discussion and Conclusion

Limitations Because gate distributions differ across CBMs as shown in Appendix A.5, the same threshold (i.e., 0.5) can have a different effect on which concepts are retained for each method, making GCG sensitive to threshold choice. Therefore, future work could explore better ways of subsampling the set of concepts used for refitting than a fixed thresholded approach. Refitting introduces an additional training stage, but is necessary to build a model whose effective concept set is guaranteed to match the open concept set. Future work could explore whether comparable predictive performance can be achieved without this stage, as in stochastic gates (Yamada et al., 2020), although those may not come with the hard guarantees on effective concept size that refitting does. Furthermore, gating selects task-relevant concepts but does not correct errors in the underlying concept vocabulary (as shown in Figure 5). Our grounding results indicate that the retained set is better semantically grounded *on average* than the baseline, but do not imply that individual concepts are reliably interpretable. Therefore, future work could explore directly incorporating GCG into the discovery process to improve the quality of the discovered concepts.

Conclusion We have shown that existing unsupervised CBMs constrain concept usage locally, per image or per class, while often retaining large concept sets across the task as a whole. Such large global vocabularies can make CBMs prohibitively difficult to audit in practice. To address this, we introduced Global Concept Gating (GCG), a simple but effective method that learns a global concept filter that can be applied on top of any existing unsupervised CBM and substantially reduces the effective concept set while maintaining competitive accuracy. Our results show that GCG can remove substantial concept redundancy while preserving competitive accuracy, and that the retained concepts more closely match ground-truth attributes and yield concept representations with less information leakage. Together, our results suggest that unsupervised CBMs should be assessed not only by how many concepts contribute to an individual prediction, but also by how many a practitioner must consider to understand the model's behaviour as a whole.

AI USE STATEMENT

In this work, we used generative AI tools for rephrasing sentences, generating tables in the paper, and providing code assistance for experiments. We have not used generative AI tools for any other tasks with required disclosure. Additionally, we used generative AI tools for language editing, code assistance, and debugging. All AI-assisted work was reviewed by the authors. AI-generated code was manually inspected, tested, and verified for correctness, and AI-assisted text was reviewed and revised by the authors to ensure that it accurately reflects the methods, results, and claims of the work. We take responsibility for the final content of this work, including all text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve human subjects, human participant data, personally identifiable information, or sensitive personal data. All experiments are conducted using publicly available datasets and pre-existing open-source models. The work does not introduce methods intended to identify, profile, or make decisions about individuals.

REPRODUCIBILITY STATEMENT

We provide the hyperparameter configurations used for all experiments in the appendix, together with detailed descriptions of the datasets, experimental setup, and evaluation procedures. All models used in our experiments are based on publicly available open-source implementations, and we use the corresponding published model configurations and concept vocabularies. The code used to run our experiments and reproduce our results will be released as open source alongside the paper. Together, these resources provide the information and implementation necessary to reproduce our experimental results.

REFERENCES

Antonio Almudévar, José Miguel Hernández Lobato, and Alfonso Ortega. There was never a bottleneck in concept bottleneck models. In *International Conference on Learning Representations*, volume 2026, pp. 114146–114165, 2026.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety, 2016. URL https://arxiv.org/abs/1606.06565.

Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared 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 M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher 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, 2020. URL https://arxiv.org/abs/2005.14165.

Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers & electrical engineering*, 40(1):16–28, 2014.

Katherine Maeve Collins, Matthew Barker, Mateo Espinosa Zarlenga, Naveen Raman, Umang Bhatt, Mateja Jamnik, Ilia Sucholutsky, Adrian Weller, and Krishnamurthy Dvijotham. Human uncertainty in concept-based ai systems. In *Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 869–889, 2023.

Nelson Cowan. The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and Brain Sciences*, 24(1):87–114, 2001. doi: 10.1017/S0140525X01003922.

Nicola Debole, Andrea Passerini, Stefano Teso, Andrea Pugnana, and Emanuele Marconato. Concepts worth having: Refining vlm-guided concept bottleneck models with minimal annotations. *arXiv preprint arXiv:2605.16405*, 2026.

Gabriele Dominici, Pietro Barbiero, Mateo Espinosa Zarlenga, Alberto Termine, Martin Gjoreski, Giuseppe Marra, and Marc Langheinrich. Causal concept graph models: Beyond causal opacity in deep learning. In *International conference on learning representations*, volume 2025, pp. 55102–55131, 2025.

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, 2021. URL https://arxiv.org/abs/2010.11929.

Mateo Espinosa Zarlenga. In defense of information leakage in concept-based models. *arXiv preprint arXiv:2606.10669*, 2026.

Mateo Espinosa Zarlenga, Pietro Barbiero, Gabriele Ciravegna, Giuseppe Marra, Francesco Giannini, Michelangelo Diligenti, Zohreh Shams, Frederic Precioso, Stefano Melacci, Adrian Weller, Pietro Lio, and Mateja Jamnik. Concept embedding models: Beyond the accuracy-explainability trade-off, 2022. URL https://arxiv.org/abs/2209.09056.

Mateo Espinosa Zarlenga, Gabriele Dominici, Pietro Barbiero, Zohreh Shams, and Mateja Jamnik. Avoiding leakage poisoning: concept interventions under distribution shifts. In *Proceedings of the 42nd International Conference on Machine Learning*, ICML'25. JMLR.org, 2025.

Isabelle Guyon and André Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

Marton Havasi, Sonali Parbhoo, and Finale Doshi-Velez. Addressing leakage in concept bottleneck models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 23386–23397. Curran Associates, Inc., 2022. doi: 10.52202/068431-1699. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/944ecf65a46feb578a43abfd5cddd960-Paper-Conference.pdf.

Eunji Kim, Dahuin Jung, Sangha Park, Siwon Kim, and Sungroh Yoon. Probabilistic concept bottleneck models. *arXiv preprint arXiv:2306.01574*, 2023.

Patrick Knab, David Steinmann, Christian Bartelt, Kristian Kersting, Bernt Schiele, Thomas Seidl, Udo Schlegel, and Wolfgang Stammer. What's in the bottle? a survey and roadmap of concept bottleneck models. *Transactions on Machine Learning Research*, 2026.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models, 2020. URL https://arxiv.org/abs/2007.04612.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Isaac Lage, Emily Chen, Jeffrey He, Menaka Narayanan, Been Kim, Sam Gershman, and Finale Doshi-Velez. An evaluation of the human-interpretability of explanation. *arXiv preprint arXiv:1902.00006*, 2019.

Yifeng Li, Chih-Yu Chen, and Wyeth W Wasserman. Deep feature selection: theory and application to identify enhancers and promoters. *Journal of Computational Biology*, 23(5):322–336, 2016.

Zachary C. Lipton. The mythos of model interpretability. *arXiv preprint arXiv:1606.03490*, 2016. URL https://arxiv.org/abs/1606.03490.

Anita Mahinpei, Justin Clark, Isaac Lage, Finale Doshi-Velez, and Weiwei Pan. Promises and pitfalls of black-box concept learning models, 2021. URL https://arxiv.org/abs/2106.13314.

Emanuele Marconato, Stefano Teso, Antonio Vergari, and Andrea Passerini. Not all neuro-symbolic concepts are created equal: Analysis and mitigation of reasoning shortcuts. *Advances in Neural Information Processing Systems*, 36:72507–72539, 2023.

George A Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, 63(2):81, 1956.

Tuomas Oikarinen, Subhro Das, Lam M. Nguyen, and Tsui-Wei Weng. Label-free concept bottleneck models, 2023. URL https://arxiv.org/abs/2304.06129.

Konstantinos P. Panousis, Dino Ienco, and Diego Marcos. Sparse linear concept discovery models, 2023. URL https://arxiv.org/abs/2308.10782.

Enrico Parisini, Tapabrata Chakraborti, Chris Harbron, Ben D MacArthur, and Christopher RS Banerji. Leakage and interpretability in concept-based models. *Journal of Machine Learning Research*, 27(174):1–40, 2026.

Emiliano Penaloza, Tianyue H. Zhang, Laurent Charlin, and Mateo Espinosa Zarlenga. Addressing concept mislabeling in concept bottleneck models through preference optimization. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 48762–48788. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/penaloza25a.html.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. URL https://arxiv.org/abs/2103.00020.

Vikram V. Ramaswamy, Sunnie S. Y. Kim, Ruth Fong, and Olga Russakovsky. Overlooked factors in concept-based explanations: Dataset choice, concept learnability, and human capability, 2023. URL https://arxiv.org/abs/2207.09615.

Sukrut Rao, Sweta Mahajan, Moritz Böhle, and Bernt Schiele. Discover-then-name: Task-agnostic concept bottlenecks via automated concept discovery, 2024. URL https://arxiv.org/abs/2407.14499.

Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead, 2019. URL https://arxiv.org/abs/1811.10154.

Simon Schrodi, Julian Schur, Max Argus, and Thomas Brox. Concept bottleneck models without predefined concepts, 2024. URL https://arxiv.org/abs/2407.03921.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 2556–2565, Melbourne, Australia, 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1238. URL https://aclanthology.org/P18-1238/.

Ivaxi Sheth and Samira Ebrahimi Kahou. Auxiliary losses for learning generalizable concept-based models. *Advances in Neural Information Processing Systems*, 36:26966–26990, 2023.

Julian Skirzynski, Harry Cheon, Shreyas Kadekodi, Meredith Stewart, and Berk Ustun. Measuring what matters: Synthetic benchmarks for concept bottleneck models. *arXiv preprint arXiv:2606.04326*, 2026.

Divyansh Srivastava, Ge Yan, and Tsui-Wei Weng. Vlg-cbm: Training concept bottleneck models with vision-language guidance, 2025. URL https://arxiv.org/abs/2408.01432.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996. doi: 10.1111/j.2517-6161.1996.tb02080.x.

Moritz Vandenhirtz, Sonia Laguna, Ričards Marcinkevičs, and Julia E. Vogt. Stochastic concept bottleneck models, 2024. URL https://arxiv.org/abs/2406.19272.

C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The caltech-ucsd birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023. URL https://arxiv.org/abs/2201.11903.

Xinyue Xu, Yi Qin, Lu Mi, Hao Wang, and Xiaomeng Li. Energy-based concept bottleneck models: Unifying prediction, concept intervention, and probabilistic interpretations. In *International Conference on Learning Representations*, volume 2024, pp. 6944–6966, 2024.

Yutaro Yamada, Ofir Lindenbaum, Sahand Negahban, and Yuval Kluger. Feature selection using stochastic gates. In *International conference on machine learning*, pp. 10648–10659. PMLR, 2020.

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 *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19187–19197. IEEE, 2023.

Mert Yuksekgonul, Maggie Wang, and James Zou. Post-hoc concept bottleneck models. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net.

Delong Zhao, Qiang Huang, Di Yan, Yiqun Sun, and Jun Yu. Partially shared concept bottleneck models, 2025. URL https://arxiv.org/abs/2511.22170.

Bolei Zhou, Aditya Khosla, Agata Lapedriza, Antonio Torralba, and Aude Oliva. Places: An image database for deep scene understanding, 2016. URL https://arxiv.org/abs/1610.02055.

A Appendix

A.1 Gate Initialisation

Table 3: Gate-initialisation ablation for DN-CBM with GCG on `CIFAR-100` ($\lambda_{\text{gate}} = 5 \times 10^{-4}$, hard-forward gate, 100 epochs, mean $\pm$ std over 3 seeds; 8,192 concepts). Every gate logit starts at the constant $\sigma(\mu_j)$. Any gate that starts closed ($\sigma(\mu_j) \leq 0.5$) never opens: the masked concept receives no task gradient and the sparsity penalty only pushes its logit down, so the model collapses to chance. Among all-open starts, the result is insensitive to the exact value.

Gate init. $\mu_j(0)$	$\sigma(\mu_j)$	**Open gates**	**Task Acc. (%)**	**Task Acc. refit (%)**
-4	0.02	0	1.0 ± 0.0	1.0 ± 0.0
-2	0.12	0	1.0 ± 0.0	1.0 ± 0.0
0	0.50	0	1.0 ± 0.0	1.0 ± 0.0
2 (default)	0.88	791.7 ± 3.4	66.2 ± 0.1	66.5 ± 0.1
4	0.98	824.7 ± 1.7	66.5 ± 0.1	66.7 ± 0.1

A.2 Hyperparameters

Shared protocol across models All four pipelines are *post-hoc*: the image backbone and the concept extractor are frozen, concept activations $c(x) \in \mathbb{R}^{|\mathcal{C}|}$ are computed once and cached, and only the final linear layer (baseline) or the gated head (GCG) is trained on the cached activations. For each (model, dataset) pair, the baseline and the GCG head are trained on *identical* cached activations. Only the seed-dependent final stage (head initialisation and minibatch order) is repeated for each seed $s \in \{0, 10, 20, 30, 40\}$. Seed-independent stages (concept discovery, the projection or concept layer, activation caching) run once, using the seed of the original implementation. All models were trained with a batch size of 64. We provide the GCG-related hyperparameters for training the models below.

Table 4: Hyperparameters of the sparse baselines and the gated (GCG) head trained on top of each backbone. $|\mathcal{C}|$ is the concept dictionary size and λ_{gate} the gate penalty of the GCG head. For every GCG run, the gate uses the hard forward straight-through estimator with all gate logits initialised to $+2$. DN-CBM and UCBM retain their baseline ℓ_1 weight penalty during gated training. All results are means over five seeds $\{0, 10, 20, 30, 40\}$.

Model	**Dataset**	**Backbone**	$\lvert\mathcal{C}\rvert$	**Epochs**	λ_{gate}
DN-CBM	`CIFAR-100`	CLIP RN50	8192	100	5×10^{-4}
	`CUB`	CLIP RN50	8192	100	3×10^{-3}
	`Places-365`	CLIP RN50	8192	100	1×10^{-4}
LF-CBM	`CIFAR-100`	CLIP RN50	892	100	1×10^{-4}
	`CUB`	ResNet-18	351	150	7×10^{-4}
	`Places-365`	ResNet-50	2161	100	5×10^{-4}
UCBM	`CIFAR-100`	ResNet-50	1000	20	1×10^{-4}
	`CUB`	ResNet-18	200	150	5×10^{-4}
	`Places-365`	ResNet-18	3650	20	3×10^{-4}
VLG-CBM	`CIFAR-100`	CLIP RN50	780	2000	5×10^{-5}
	`CUB`	ResNet-18	671	4000	5×10^{-5}
	`Places-365`	CLIP RN50	2186	200	1×10^{-4}

A.3 Refitting

Gated vs. refit models We distinguish two models obtained from a GCG run. The *gated* model is the jointly trained model. During training, the forward pass applies the hard mask $\mathbf{m}$ and gradients flow through the soft gate $\sigma(\boldsymbol{\mu})$ (Section 3). At evaluation, we apply the soft gates directly, so each

concept enters the probe scaled by its gate value:

$$\hat{\mathbf{y}} = \mathbf{W}_f\big(\hat{\mathbf{c}} \odot \sigma(\boldsymbol{\mu})\big) + \mathbf{b}_f.$$

The concept count, however, is defined by the hard threshold, $\mathcal{K} = \{\, j : \sigma(\mu_j) > \frac{1}{2} \,\}$. Thus, a concept with $\sigma(\mu_j) \le \frac{1}{2}$ is counted as closed, yet it still contributes $\sigma(\mu_j)\,\hat{c}_j$ to the prediction. The gates also do not need to approach 0 for this to matter, because the probe can compensate for a small gate by increasing the corresponding weights in $\mathbf{W}_f$. The accuracy of the gated model therefore partly relies on concepts outside $\mathcal{K}$.

Therefore, we use a *refit* model to resolve this mismatch. We fix $\mathcal{K}$, discard every concept $j \notin \mathcal{K}$ together with the gate, and retrain only the task probe on the surviving activations $\{\hat{\boldsymbol{c}}_j\}_{j\in\mathcal{K}}$ with $\mathcal{L}_{\text{base}}$. This gives $\hat{\boldsymbol{y}} = \tilde{\mathbf{W}}_f \hat{\boldsymbol{c}}_{\mathcal{K}} + \tilde{\boldsymbol{b}}_f$, a standard CBM that uses exactly the $|\mathcal{K}|$ reported concepts. Because the concept encoder g stays fixed, refitting only changes how the concepts in $\mathcal{K}$ are weighted, not the concepts themselves. The difference between the two models therefore measures how much the gated model relied on concepts it reports as closed.

Since refitting retrains the task probe, we verify that it does not substantially change the model's behaviour by comparing the accuracy of the *gated* and *refit* models in Table 5. Refitting changes accuracy by at most 3.3 points and by under 0.5 points in 6 of 12 settings. Thus, refitting ensures that the reported performance is attributable solely to the retained concepts, confirming that the open concept set is sufficient for the task, while showing minimal differences in task performance from the *gated* model.

Table 5: Gated head vs. mask-and-refit head for every GCG run. *Gated* is the test accuracy of the trained gated head (hard-forward STE gate) evaluated as is; *Refit* is the accuracy after fixing the open set and refitting a plain linear head on those $|\mathcal{K}|$ concepts alone (20 epochs). $|\mathcal{C}|$ is the concept dictionary size; Δ is Refit − Gated in accuracy points. Mean ± std over 5 seeds; λ_{gate} as in Table 4. For VLG-CBM/CUB the refit SAGA head additionally zeroes some open concepts, so the concepts it actually uses number $38.6_{\pm 1.4}$.

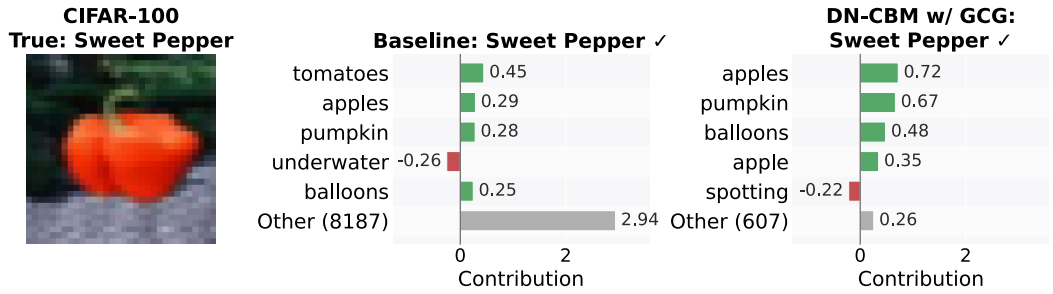
Model	Dataset	$\lvert\mathcal{C}\rvert$	$\lvert\mathcal{K}\rvert$	Gated Acc. (%)	Refit Acc. (%)	Δ
DN-CBM	CIFAR-100	8192	$791.60_{\pm 2.97}$	$66.19_{\pm 0.10}$	$67.12_{\pm 0.04}$	$+0.93$
	CUB	8192	$242.20_{\pm 0.40}$	$68.38_{\pm 0.07}$	$68.18_{\pm 0.05}$	-0.20
	Places-365	8192	$1694.80_{\pm 13.80}$	$52.82_{\pm 0.04}$	$52.90_{\pm 0.08}$	$+0.08$
LF-CBM	CIFAR-100	892	$598.80_{\pm 0.40}$	$67.15_{\pm 0.10}$	$67.31_{\pm 0.11}$	$+0.16$
	CUB	370	$267.00_{\pm 1.30}$	$63.25_{\pm 0.24}$	$63.77_{\pm 0.08}$	$+0.52$
	Places-365	2145	$166.80_{\pm 3.60}$	$41.95_{\pm 0.02}$	$41.06_{\pm 0.49}$	-0.86
UCBM	CIFAR-100	1000	$204.20_{\pm 1.17}$	$57.91_{\pm 0.09}$	$60.41_{\pm 0.13}$	$+2.50$
	CUB	200	$173.60_{\pm 3.10}$	$73.43_{\pm 0.24}$	$74.57_{\pm 0.26}$	$+1.14$
	Places-365	3650	$430.20_{\pm 4.12}$	$50.53_{\pm 0.15}$	$50.79_{\pm 0.14}$	$+0.26$
VLG-CBM	CIFAR-100	780	$697.60_{\pm 1.50}$	$67.77_{\pm 0.02}$	$67.39_{\pm 0.14}$	-0.38
	CUB	671	$44.80_{\pm 1.47}$	$72.80_{\pm 0.15}$	$72.33_{\pm 0.20}$	-0.47
	Places-365	2186	$107.80_{\pm 0.40}$	$40.65_{\pm 0.07}$	$43.98_{\pm 0.07}$	$+3.33$

A.4 Explanations

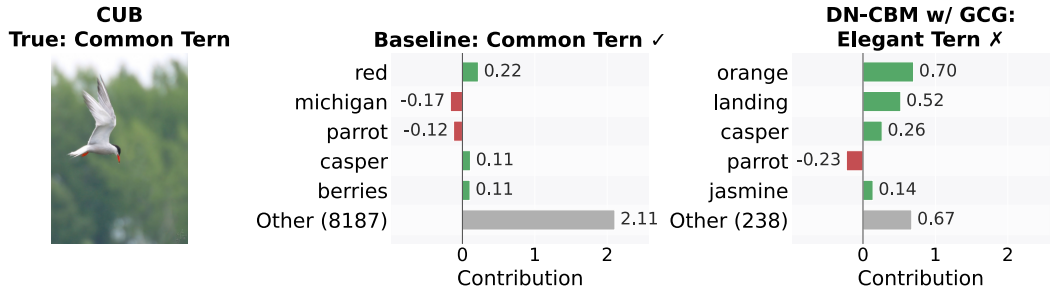
We show further examples of explanations generated by the baseline DN-CBM and its GCG counterpart. In Figure 6a, both models predict correctly, but the baseline's five shown concepts account for only 1.01 of the predicted logit, against 2.94 from the 8,187 concepts omitted from the explanation. After gating, the shown concepts contribute 2.00 against 0.26 from the 607 remaining, so the explanation accounts for most of the decision rather than a fraction of it. Although individual concept names remain difficult to interpret, owing to the underlying DN-CBM vocabulary, GCG concentrates the prediction on fewer, larger contributions.

In Figure 6b, DN-CBM with GCG misclassifies a Common Tern as the visually similar Elegant Tern, while the baseline predicts correctly. However, the baseline's correct prediction relies almost entirely on concepts outside the explanation (2.11 from 8,187 unshown concepts, against 0.15 from the top five), whereas for DN-CBM with GCG the top five concepts contribute 1.39, and the top

concept alone (0.70) accounts for as much of the logit as the 238 remaining concepts combined (0.67). The gated model's error is therefore easier to trace to specific concepts.



(a) Example explanation for CIFAR-100

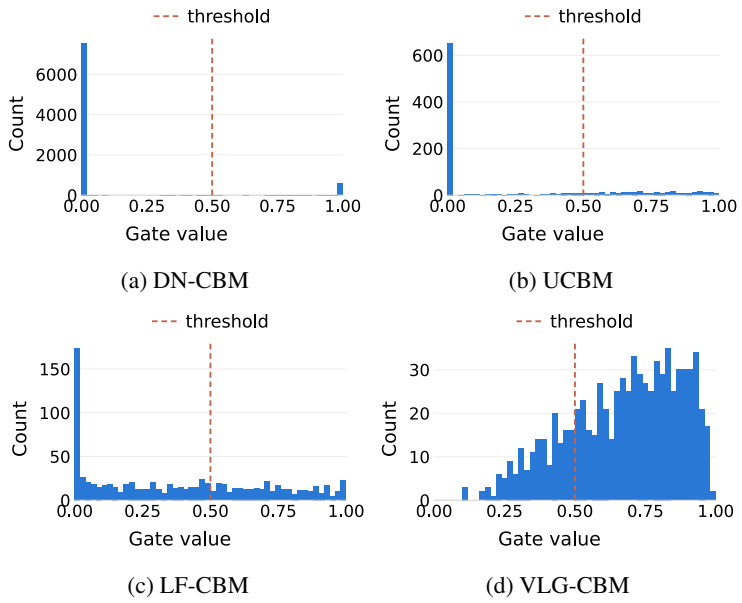


(b) Example explanation for CUB

Figure 6: Example explanations from DN-CBM with and without GCG. In both cases, GCG concentrates the prediction's support in the concepts shown, while the baseline relies heavily on unshown concepts or spreads support thinly. Repeated names (e.g., apple/apples) denote distinct concepts, which is a limitation of the vocabulary set used by DN-CBM.

A.5 Gate Distributions

We measure how the gate distributions of the four models for CIFAR-100, showing how the gate distributions vary per method. While for DN-CBM, the gates are fully bimodal and converge to either 0 or 1, this is not the case for the other models. This highlights the need for refitting to ensure that, after applying the 0.5 threshold, the concepts identified as "open" are sufficient to support the model's predictions. Refitting therefore allows us to evaluate the performance of the model using only the retained concepts, rather than relying on the soft gate values learned during training.



(a) DN-CBM (b) UCBM

(c) LF-CBM (d) VLG-CBM

Figure 7: Gate distributions of the models. We display the gate values in bins and the number of gates with values corresponding to each bin. We note that the gate distribution is drastically different per model.