Walking the Web of Concept-Class Relationships in Incrementally Trained Interpretable Models

Susmit Agrawal, Deepika Vemuri, Sri Siddarth Chakaravarthy P, Vineeth N. Balasubramanian

Indian Institute of Technology Hyderabad
{ai22mtech12002, ai22resch11001}@iith.ac.in, {sri.siddarth, vineethnb}@cse.iith.ac.in

Abstract

Concept-based methods have emerged as a promising direction to develop interpretable neural networks in standard supervised settings. However, most works that study them in incremental settings assume either a static concept set across all experiences or assume that each experience relies on a distinct set of concepts. In this work, we study concept-based models in a more realistic, dynamic setting where new classes may rely on older concepts in addition to introducing new concepts themselves. We show that concepts and classes form a complex web of relationships, which is susceptible to degradation and needs to be preserved and augmented across experiences. We introduce new metrics to show that existing concept-based models cannot preserve these relationships even when trained using methods to prevent catastrophic forgetting, since they cannot handle forgetting at concept, class, and concept-class relationship levels simultaneously. To address these issues, we propose a novel method - **MuCIL**- that uses multimodal concepts to perform classification without increasing the number of trainable parameters across experiences. The multimodal concepts are aligned to concepts provided in natural language, making them interpretable by design. Through extensive experimentation, we show that our approach obtains state-of-the-art classification performance compared to other concept-based models, achieving over 2× the classification performance in some cases. We also study the ability of our model to perform interventions on concepts, and show that it can localize visual concepts in input images, providing post-hoc interpretations.

Code — https://github.com/Susmit-A/MuCIL
Appendix — https://susmit-a.github.io/misc/appendix.pdf

1 Introduction

Concept-based models have gained the attention of the computer vision community in recent years (Kim et al. 2023; Margeloiu et al. 2021; Barker et al. 2023; Koh et al. 2020), as a means of interpreting the output prediction in terms of learned or human-defined atomic interpretable '*concepts*'. These models attempt to predict the target class generally as a weighted linear combination of meaningful concepts. For example, such a model may identify the concept set

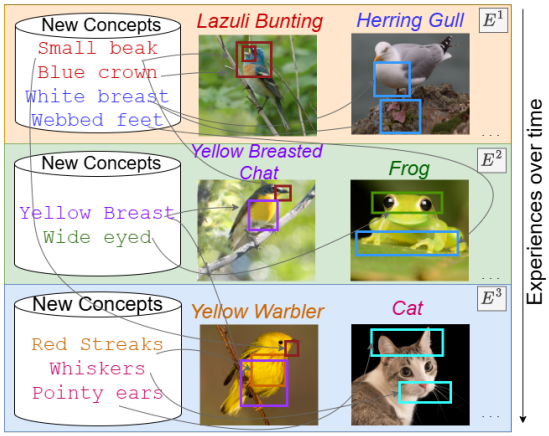

Figure 1: Illustration of our setting. Concepts introduced in an experience are shared among classes from other experiences. **MuCIL** focuses on the difficult challenge of capturing and preserving this web of class-concept relationships over multiple experiences.

{*whiskers, four legs, pointy ears, sociable*} as key semantic attributes that help make a prediction *cat* on a given image. But with the recent growth of incremental model learning, a pertinent question arises – can the abovementioned model be adapted later to identify *golden retriever* in addition to *cat*? The two classes share some common concepts such as {*four legs, sociable*}, while having discriminative concepts such as {*whiskers, pointy ears*} in case of *cat* and {*golden fur, floppy ears*} in case of *golden retriever*. Incrementally learning such concept-based models with newer classes as well as concepts forms the key focus of this work.

Teaching neural network models about new distinctive concepts (e.g. *golden fur*) of new classes (e.g. *dog*), while reusing previously known concepts (*four legs, sociable*) is a highly nontrivial problem. While concept-based models have seen a range of efforts in recent years, there has been very little work in incrementally learning such models. The limited existing efforts either assume that all new classes share the same pool of attributes as older ones (Marconato et al. 2022), or have independent non-overlapping attribute sets (Rymarczyk et al. 2023). In this work, we propose a

more general and realistic setting in which classes seen at a later stage may share concepts from past classes while introducing new concepts of their own. This creates a complex web of concept-class relationships across experiences, as illustrated in Figure 1, which needs to be preserved and expanded as the model learns to classify and explain new classes. Note that, as in standard incremental learning, the model is required to achieve good classification performance on newly introduced classes, while maintaining classification performance on previously seen ones.

While traditional deep learning models are susceptible to forgetting old classes as new ones are introduced (*catastrophic forgetting* (Hadsell et al. 2020)), we find that concept-based models are susceptible to forgetting *concept-class relationships* as well. Overcoming two levels of forgetting in incrementally learning concept-based models presents new challenges that need to be addressed explicitly.

To this end, we propose MuCIL, a novel **Mu**ltimodal **C**oncept-Based **I**ncremental **L**earner. We combine embeddings of text concepts, called *concept anchors*, with image representations to create *multimodal concept embeddings* for a given image. These embeddings are latent vectors containing information that helps in classification while also providing interpretations. We propose the use of *concept grounding*, which allows the interpretation of multimodal concepts in terms of text-based concepts. All these components are integrated with the fundamental consideration that the model should be able to incorporate new classes and new concepts continually, while also forming new concept-class associations that may emerge in the process.

Our key contributions can be summarized as follows: (i) We propose a new method for the relatively new setting of concept-based incremental learning, where a model adapts dynamically to new classes as well as new concepts; (ii) We introduce *multimodal concept embeddings*, a combination of image embeddings and interpretable concept anchors, as part of our method to perform classification. Our approach is primarily intended to allow scalability of concept-based models to newer experiences without increase in parameters; (iii) We perform a comprehensive suite of experiments to evaluate our method on well-known benchmark datasets. We study our method's performance both in a incremental as well as standard supervised settings, achieving state of the art results; (iv) We propose three new metrics to evaluate concept-based models in the proposed setting: Concept-Class Relationship Forgetting, Concept Linear Accuracy and Active Concept Ratio. Our approach can offer concept-specific localizations implicitly as a means of interpreting the model prediction (see Figure 5), without being explicitly trained to do so.

2 Related Work

Interpretability of Deep Neural Network Models: Interpretability methods in DNN models can be broadly classified into post-hoc and ante-hoc methods. Post-hoc methods aim to interpret model predictions after training (Selvaraju et al. 2017; Chen et al. 2020; Chattopadhay et al. 2018; Sattarzadeh et al. 2021; Yvinec et al. 2022; Benitez et al. 2023; Sundararajan and Najmi 2020; Wang, Wang, and Inouye 2020; Jethani et al. 2021; Wang, Wang, and Inouye 2020; Dabkowski and Gal 2017; Fong and Vedaldi 2017; Petsiuk, Das, and Saenko 2018; Montavon et al. 2019). Recent efforts have highlighted the issues with post-hoc methods and their reliability in reflecting a model's reasoning (Rudin 2019; Vilone and Longo 2021; Nauta et al. 2023). On the other hand, ante-hoc methods that jointly learn to explain and predict provide models that are inherently interpretable (Sokol and Flach 2021; Benitez et al. 2023). Ante-hoc methods have also been found to provide interpretatations that help make the model more robust and reliable (Alvarez-Melis and Jaakkola 2018; Chattopadhyay et al. 2022). We focus on this genre of methods in this work. (Koh et al. 2020) proposed Concept Bottleneck Models (CBMs), a method that uses interpretable, human-defined concepts, combining them linearly to perform classification. CBMs also allow human interventions on concept activations (Shin et al. 2023; Steinmann et al. 2023) to steer the final prediction of a model. (Kim et al. 2023; Collins et al. 2023; Yan et al. 2023) obtained the intermediate semantic concepts by replacing domain experts with Large Language Models (LLMs). This allows for ease and flexibility in obtaining the concept set. Using LLMs to obtain concepts also allow grounding of neurons in a bottleneck layer to a human-understandable concept, an issue with CBMs that was highlighted in (Margeloiu et al. 2021). Other concept-based methods (Alvarez-Melis and Jaakkola 2018; Chen, Bei, and Rudin 2020; Kazhdan et al. 2020; Rigotti et al. 2021; Benitez et al. 2023) use a different notion of concepts based on prototype representations (see appendix §A1).

Concept-Based Incremental Learning: While Incremental Learning in standard supervised settings has been widely explored (Wang et al. 2023), Concept-Based Incremental Learning has remained largely unstudied. We identify (Marconato et al. 2022) as an early effort in this direction; however, this work trains CBMs in a continual setting under an assumption that all concepts, including those required for unseen classes, are accessible from the first experience itself, which does not emulate a real-world setting. More recently, (Rymarczyk et al. 2023) proposed an interpretable CL method that uses part-based prototypes as concepts. As mentioned earlier, our notion of concepts allows us to go beyond parts of an object category, as in CBM-based models.

3 MuCIL: Methodology

Preliminaries and Notations. Given a sequence of T experiences, with each experience i consisting of n image-label pairs $(\mathcal{X}^i, \mathcal{Y}^i) = \{(x_1^i, y_1^i), (x_2^i, y_2^i), ..., (x_n^i, y_n^i)\}$, a class-incremental continual learning (CIL) system aims to learn an experience t without catastrophically forgetting the previous $t-1$ experiences. In the scenario where finer class details are available as concepts for classification, each experience i consists of n image-label-concept tuples $(\mathcal{X}^i, \mathcal{Y}^i, \mathcal{C}^i) = \{(x_1^i, y_1^i, \mathcal{C}_1^i), (x_2^i, y_2^i, \mathcal{C}_2^i), ..., (x_n^i, y_n^i, \mathcal{C}_n^i)\}$, where $\mathcal{C}^i$ is the set of concepts known during experience i and $\mathcal{C}_k^i$ is the set of active concepts in example k. The set of concepts known during i is the union of all concept sets from experiences 1 to i. We use the same active concept set for all instances of the same class, i.e., $\mathcal{C}_{k_1}^i = \mathcal{C}_{k_2}^i$ if $y_{k_1}^i = y_{k_2}^i$,

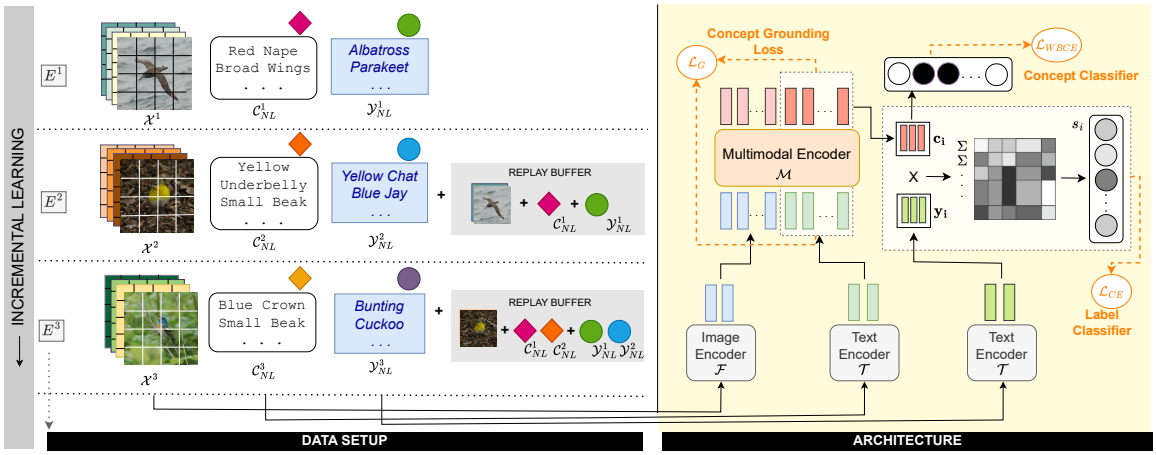

Figure 2: **Overview of our setup and proposed architecture.** Our architecture receives new classes and associated concepts across multiple experiences in a CL setting. We use pre-trained language and vision encoders to get embeddings of the input image, concepts, and classes. These are then used to create multimodal image-concept embeddings using our *Multimodal Encoder*. The multimodal concept embeddings are grounded to their concept anchors using a loss function and are used to predict both class labels and the presence/absence of corresponding concepts in the image.

as done in prior work (Koh et al. 2020; Yang et al. 2023; Oikarinen et al. 2023). Annotating instance-level concepts for large datasets is a challenging and error-prone task, requiring large amounts of time and effort by domain experts. Instead, using concepts at a class level allows us to specify the general characteristics of a class. We obtain these concepts directly from annotations in certain datasets (e.g. CUB) and use a Large Language Model (LLM) in other cases. One could view this as a weakly supervised setting with noisy concept labels.

Challenges. While concept-based models for image classification have grown in the community, extending them to learn across experiences and new classes incrementally is non-trivial, introducing new challenges. (1) **Forgetting at Two Levels:** In a traditional CIL setting, the catastrophic forgetting of previously learned classes in newer experiences is only at a class-level. In our proposed setting, it includes both concept-level and concept-class relationship forgetting. Beyond forgetting of concepts from previous experiences, concept-class relationship forgetting can be more critical, affecting the model's ability to recognize previously encountered classes based on the concepts they were originally associated with. This affects the model's ability to understand and preserve how concepts relate to classes across experiences. (2) **Parameter Scaling:** Extending existing concept-based learning frameworks to a CIL setting would necessitate the addition of new learnable parameters with each new experience, leading to larger models, which is undesirable. Therefore, designing a method capable of integrating new classes and concepts without expanding the parameter space is ideal. (3) **Information Bottleneck:** Another challenge is the use of a single representation for encoding all relevant visual and concept information, which could lead to an information bottleneck as the system scales to accommodate more concepts and classes. (4) **Semantic Misrepresentations:** Lastly, some existing methods do not accurately capture the semantics of concepts (Margeloiu et al. 2021). One widely used method to explicitly encode concept semantics is leveraging textual concept embeddings. These semantics should be respected and updated as new concepts arrive. A naive way to implement this is by projecting a network's output representation on pre-computed concept embeddings (Yang et al. 2023; Oikarinen et al. 2023), which amplifies the unified representation problem. We propose a more flexible method for overcoming this issue.

We propose a novel framework that addresses each of the abovementioned challenges. Our overall solution consists of multiple components: (1) A **Multimodal Encoder**, that fuses visual information and textual concepts to create *multimodal image-concept embeddings*. It is designed to handle an increasing number of concepts without growing the number of parameters. It also creates multiple distinct representations in parallel to avoid information bottlenecks while preserving concept semantics. (2) A **Parameter-free Clasifier** that uses the multimodal concept embeddings along with the textual descriptions of classes to perform classification. (3) **Concept Neurons** that predict whether or not a given concept exists in an image and also allow probing the model for interpretability.

Multimodal Image-Concept Encoder. We introduce a novel multimodal image-concept encoder that can be integrated into any standard transformer architecture. It is used to merge image embeddings and textual concept embeddings, thereby generating a sequence of multimodal embeddings that capture both visual and concept inputs.

Our encoder incorporates new concepts without expanding its parameter space. This is made possible by the transformer's inherent ability to handle variable-length sequences, allowing the model to accommodate additional concepts by appending them to the input sequence without

necessitating an increase in parameters. The use of transformer encoder layers enables the processing of mixed data inputs, resulting in a sequence of multimodal image and concept embeddings by simultaneously attending to the two modalities. Using a transformer is hence not a drop-in replacement to a different network architecture such as a CNN, but is a central part of our overall methodology.

Formally, the multimodal image-concept encoder, $\mathcal{M}$, is a stack of transformer encoder layers that receive image embeddings $\mathbf{x}_i$ as well as textual concept embeddings $\mathbf{c_1 c_2 ... c}_{|\mathcal{C}^i|}$ as input. Note that this includes *all concepts available until the current experience*. The output of $\mathcal{M}$ is a sequence of vectors $\{\mathcal{X}'^{(i)}, \mathcal{C}'^{(i)}\}$, where $\mathcal{X}'^{(i)}$ corresponding to the image patch embeddings $\{\mathbf{x}'_i\}$ and $\mathcal{C}'^{(i)} = \{\mathbf{c}'_\mathbf{1}, \mathbf{c}'_\mathbf{2}, ..., \mathbf{c}'_{|\mathcal{C}^\mathbf{i}|}\}$ are concept embeddings. These embeddings combine both textual and visual information and hence are multimodal by design. Similar to standard transformer-based classification where all outputs except the CLS token are discarded, the $\mathcal{X}'^{(i)}$ plays no further part in our methodology.

The concept embeddings $\mathcal{C}'^{(i)}$ do not have any semantic grounding however; we hence use a *Concept Grounding Loss* to map them to known natural language-based concept anchors, as below:

$$\mathcal{L}_G = -\frac{1}{|\mathcal{C}^i|}\sum_{k=1}^{|\mathcal{C}^i|}\frac{\mathbf{c}_k \cdot (\mathbf{W}^T\mathbf{c}'_k + \mathbf{b})}{|\mathbf{c}_k| \cdot |\mathbf{W}^T\mathbf{c}'_k + \mathbf{b}|} \tag{1}$$

$\mathbf{W}$ and $\mathbf{b}$ are learnable parameters that are shared among concepts. This loss semantically aligns the learned concept embeddings with their corresponding concept text anchors in the ground truth by ensuring that all original embeddings can be recovered using a single linear transform. Intuitively, this enforces each output concept vector to be associated with a deterministic representation of the concept, viz the text embedding. This deterministic association with a known ("grounded") reference guarantees that the vector has an associated semantic meaning, thus providing interpretability.

Parameter-free incremental classification. As stated earlier, existing concept-based learning frameworks use dense layers on top of concept embeddings, which makes it challenging to scale to newer experiences continually. We hence take inspiration from methods such as (Radford et al. 2021) to use text embeddings of classes to perform classification, thus allowing our method to be parameter-free in the classifier and allowing for scalability to newer experiences.

Classification is hence performed by returning the index of the class embedding which aligns most with the vectors in $\mathcal{C}'$ for a given input sample. The alignment between the jth concept embedding of the current sample, $\mathbf{c}'_j$, and the text embedding of the kth class $\mathbf{y}_k$ is computed as $\mathbf{c}'_j \cdot \mathbf{y}_k$, their dot product. The final classification result is hence given by $\hat{y} = \text{argmax}_s(s_1, s_2, ..., s_{|\mathcal{Y}^i|})$, where $s_k = \sum_{j=1}^{|\mathcal{C}^i|}\mathbf{c}'_j \cdot \mathbf{y}_k$, which returns the index of the class that aligns most strongly with concept embeddings of a given input. We use s_k as the logit of class k, and perform a softmax operation on the logits to get classification probabilities, which are used to train the model with the standard cross-entropy loss $\mathcal{L}_{CE}$. We note that deriving class strengths from the concept embeddings in the above manner does not require additional parameters in newer experiences; this is a key element of our framework that enables scalability to both unseen classes and concepts when deployed in a CL setting.

Concept Neurons. The presence of a concept is evaluated using a single shared dense layer with a sigmoid activation, denoted by $\sigma(\cdot)$, applied independently on each $\mathbf{c}'_\mathbf{i} \in \mathcal{C}'$. A weighted binary cross-entropy loss $\mathcal{L}_{WBCE}$ is used to train the layer using provided ground-truth concept labels. We refer to the output logits of the layer $\sigma(\mathcal{C}')$ collectively as *concept neurons*. In MuCIL, concept neurons serve two purposes: (i) They provide additional supervision to learn concept embeddings better; and (ii) They provide an interface for identifying which concepts are active or inactive, thus enhancing interpretability. This allows for probing the model to evaluate the quality of concepts learned, and study how they change over experiences (see §4). We show a detailed illustration explaining them in appendix (§A2).

Details of $\mathcal{L}_{WBCE}$. For a given image, the ratio of the number of active concepts to the number of total concepts is quite small. This necessitates penalizing the misclassification of active concepts more strongly than the misclassification of inactive concepts. We do this by weighting the loss for active concepts by the fraction of *inactive concepts*, and weighting the loss for inactive concepts by the fraction of *active concepts*. The loss $\mathcal{L}_{WBCE}$ is then defined as:

$$\mathcal{L}_{WBCE} = \frac{\text{\# inactive concepts}}{\text{\# concepts}}\sum_{i=1}^{|\mathcal{C}^{\text{active}}|}\mathcal{L}_{BCE}(\sigma(\mathbf{c}'_i), 1) + \frac{\text{\# active concepts}}{\text{\# concepts}}\sum_{i=j}^{|\mathcal{C}^{\text{inactive}}|}\mathcal{L}_{BCE}(\sigma(\mathbf{c}'_j), 0)$$

Training Procedure. MuCIL thus has the following learnable components trained simultaneously end-to-end: parameters of $\mathcal{M}$, parameters $\mathbf{W}$ and $\mathbf{b}$ in $\mathcal{L}_G$, and the weights of the dense layer used for obtaining concept neuron values, $\sigma(\cdot)$. The global objective $\mathcal{L}$ is a weighted sum of the three different objectives:

$$\mathcal{L} = \mathcal{L}_{CE} + \lambda_1\mathcal{L}_{WBCE} + \lambda_2\mathcal{L}_G \tag{2}$$

Intuitively, the terms $\mathcal{L}_{WBCE}$ and $\mathcal{L}_G$ control the concept-based learning in the framework. $\mathcal{L}_{WBCE}$ specifically trains $\mathcal{M}$ and σ jointly to enable detection of the presence/absence of concepts, thus ensuring that visual features are properly represented in the multimodal concept vectors. $\mathcal{L}_G$ trains $\mathcal{M}, \mathbf{W}$, and $\mathbf{b}$ to enforce the concept embeddings to be grounded to their textual anchors. The $\mathcal{L}_{CE}$ term trains $\mathcal{M}$ to perform classification based on the outputs in $\mathcal{C}'$.

Inference. At test-time, given an image x^{test} to be classified after experience t, we obtain its embedding vectors $\mathbf{x}^{test}$ and append them to the sequence of concept embeddings $\mathbf{c_1 c_2 ... c}_{|\mathcal{C}^t|}$. $\mathcal{M}$ receives this combined sequence as input and outputs a sequence of vectors $\{\mathcal{X}'^{(test)}, \mathcal{C}'^{(test)}\}$. $\mathcal{C}'^{(test)}$ is provided to the parameter-free classifier for obtaining the final class label. The active/inactive concepts in x^{test} can be identified by thresholding the outputs of the concept neurons $\sigma(\mathcal{C}'^{(test)})$.

4 Experiments and Results

We perform a comprehensive suite of experiments to study the performance of **MuCIL** on well-known benchmarks: CIFAR-100, ImageNet-100 (INet-100), and CalTech-UCSD Birds 200 (CUB200). To study how our method works when concept annotations are provided with the dataset or otherwise, we use the human-annotated concepts provided in case of CUB200, and derive concepts for CIFAR-100 and ImageNet-100 from GPT 3.5 as described in (Oikarinen et al. 2023). We study our method in standard supervised learning setting as well as in a Class-Incremental Learning setting as done in (Marconato et al. 2022; Rymarczyk et al. 2023), which we refer to as **SL** and **CL** settings respectively in our experiments. In the CL setting, we study our performance over 5 and 10 experiences using concept-based methods in conjunction with three well-known CL algorithms: Experience Replay (ER) (Rebuffi et al. 2017), A-GEM (Chaudhry et al. 2019), and DER++ (Buzzega et al. 2020), with a replay buffer size of 500 (we study other variations of buffer size in the Appendix). We select these CL algorithms since they can be adapted with concept-based model baselines. All implementation details (dataset details, architecture, hyperparameters) are in Appendix (§A2,§A4).

Baselines. We compare our method with existing works that perform concept-based class-incremental learning as well as by adapting other works that use concept-based learning to the CL setting (due to lack of many explicit efforts on this setting). Our baseline methods for comparison include: (i) ICIAP (Marconato et al. 2022), which makes the assumption that all concepts, including those that would likely only be provided in future experiences, are provided upfront; (ii) *Incremental CBM*, a version of the Concept Bottleneck Model (Koh et al. 2020) that we modify to adapt to a class-incremental and concept-incremental learning scenario. We grow both the bottleneck layer and the linear classification layer as new classes and new concepts are introduced. We train these two baselines in sequential (-S) and joint (-J) settings as described in (Koh et al. 2020); (iii) We also compare with *Label-Free CBM* (Oikarinen et al. 2023) and *LaBo* (Yang et al. 2023), variations of CBM that use projections of image embeddings onto natural language concept embeddings to form the bottleneck layer. Due to the frozen feature extractors and use of Generalized Linear Models (Label-Free) or Submodular Functions (LaBo) over the concept layer, extending these baselines to different CL algorithms is non-trivial. The same is applicable to CBM-S and ICIAP-S which involve multiple training stages with the feature extractor being fixed after the first stage. These methods are hence most compatible with ER, which we use for the corresponding baselines. Hence, for all ablation studies and analysis, we focus on using ER as the CL algorithm due to compatibility across all considered baseline models and to ensure fairness of comparison.

Performance Metrics: Classification. In the CL setting, we use two well-known performance metrics: *Final Average Accuracy (FAA)* and *Average Forgetting (AF)*. FAA is defined as: $FAA = \frac{1}{T}\sum_{i=1}^{T} acc_i^T$, where acc_i^T represents the model's accuracy on the validation split of experience i after training on T experiences. AF at experience T is defined as: $AF = \frac{1}{T-1}\sum_{i=1}^{T-1} acc_i^i - acc_i^T$, i.e., the difference in accuracy on the validation set of experience i when it was originally learned and the accuracy on it after the model has been trained on T experiences. We use the standard *Classification Accuracy* in the SL setting.

New Performance Metrics: Concept Evaluation. In order to evaluate the learned concepts and their evolution across experiences, we propose three new quantitive metrics: *concept linear accuracy*, *active concept ratio* and *concept-class relationship forgetting*. We briefly describe each of these metrics, before presenting our results.

Concept Linear Accuracy: We use our concept neurons to evaluate how well concepts capture the relevant semantic information (for performing classification), and to study how they preserve this information over experiences. The group of concept neurons are treated as a bottleneck layer, and a linear classifier is trained on top of the neuron logits. We denote the linear accuracy of concept neurons over a class set $\mathcal{Y}^i$ and a concept set $\mathcal{C}^j$ after training the model on t experiences as $LA(t, \mathcal{Y}^i, \mathcal{C}^j)$, where t is varied across CL experiences.

Concept-Class Relationship Forgetting: We define Concept-Class Relationship Forgetting (CCRF) of a concept set as the loss of its ability to provide relevant information to perform class-level discrimination over time. This can occur when concepts no longer align well to visual semantics in the provided image. This is different from forgetting in standard CL settings (Hadsell et al. 2020) as the model may predict concepts correctly, but instead forgets how concepts correlate to classes. Mathematically, we measure CCRF as:

$$CCRF = \frac{1}{T-1}\sum_{t=2}^{T}\frac{1}{t-1}\sum_{k=t-1}^{1}[LA(k, \mathcal{Y}^{k\setminus k-1}, \mathcal{C}^k) - LA(t, \mathcal{Y}^{k\setminus k-1}, \mathcal{C}^k)] \quad (3)$$

The term $LA(k, \mathcal{Y}^{k\setminus k-1}, \mathcal{C}^k) - LA(t, \mathcal{Y}^{k\setminus k-1}, \mathcal{C}^k)$ can also take negative values, indicating that the information captured by the concepts of experience k is enhanced after training on the future experience t instead of degrading. In our framework, the model generating concept logits for experience i is the combination of $\mathcal{M}$ and σ trained on experience i; in a CBM, this is the model upto the bottleneck layer.

Active Concept Ratio: To study the relevance of concepts to classes across experiences, we propose Active Concept Ratio (ACR) to measure how frequently concepts seen during experience i activate when classifying images from experience j. A high value indicates that concepts from experience i play an important role in understanding classes from experience j. Ideally, classes introduced in experience i should have their highest ACR values associated with the concept set from the same experience, since those concepts best explain the classes. Positive ACR values with concept sets from other experiences indicate that those concepts are also activated in response to classes from experience i. Formally, let N_j be the number of images to be classified in experience j. Then, the ACR for a concept set presented in experience i for classifying images from experience j is defined as

		CIFAR-100				CUB				INet-100			
		5Exp		10Exp		5Exp		10Exp		5Exp		10Exp	
Method	**CL Algo**	FAA	AF	FAA	AF	FAA	AF	FAA	AF	FAA	AF	FAA	AF
CBM-J (2020)	ER	0.23	0.74	0.15	0.81	0.38	0.36	0.22	0.50	0.30	0.59	0.17	0.67
CBM-J (2020)	A-GEM	0.17	0.82	0.09	0.81	0.11	0.59	0.05	0.48	0.15	0.68	0.09	0.73
CBM-J (2020)	DER++	0.22	0.76	0.13	0.84	0.38	0.40	0.27	0.49	0.27	0.64	0.16	0.69
ICIAP-J (2022)	ER	0.25	0.72	0.13	0.82	0.38	0.36	0.22	0.50	0.31	0.57	0.18	0.67
ICIAP-J (2022)	A-GEM	0.17	0.79	0.09	0.83	0.11	0.59	0.05	0.48	0.17	0.67	0.09	0.74
ICIAP-J (2022)	DER++	0.22	0.76	0.14	0.83	0.38	0.40	0.27	0.49	0.28	0.63	0.15	0.70
CBM-S (2020)	ER	0.30	0.69	0.28	0.71	0.35	0.46	0.22	0.54	0.29	0.61	0.21	0.65
ICIAP-S (2022)	ER	0.21	0.62	0.20	0.69	0.35	0.46	0.22	0.54	0.22	0.55	0.14	0.55
Label-Free (2023)	ER	0.22	0.34	0.19	**0.24**	0.31	0.38	0.42	0.48	0.07	0.31	0.11	0.26
LaBo (2023)	ER	0.30	0.76	0.10	0.80	0.29	0.57	0.07	0.67	0.41	0.53	0.05	0.61
MuCIL (Ours)	ER	0.67	0.35	**0.63**	0.38	0.78	0.11	**0.76**	0.14	0.80	0.09	0.79	**0.09**
MuCIL (Ours)	A-GEM	0.36	0.73	0.44	0.59	0.30	0.71	0.15	0.82	0.66	0.27	0.71	0.19
MuCIL (Ours)	DER++	**0.68**	**0.33**	0.62	0.39	**0.81**	**0.07**	**0.76**	**0.13**	**0.81**	**0.08**	**0.80**	**0.09**

Table 1: CL performance (FAA = Final Average Accuracy, AF = Average Forgetting) of diff methods averaged over three random model initializations, on 5 and 10 experiences with buffer size 500, and three different CL algorithms. Top super-row contains baselines adapted to different CL methods. Middle super-row contains methods that are compatible with ER (adapting to other CL algorithms is non-trivial). Bottom super-row contains results for **MuCIL** trained using three different CL algorithms. **MuCIL** consistently delivers better performance than baselines (in all cases, std deviation ≤ 0.02). Additional results for ER with buffer sizes 2000 and 5000 have been provided in the Appendix (§A3).

$\left(\sum_{n=1}^{N_j} \hat{\mathcal{C}}_n^{i\setminus(i-1)}\right) / \left(\sum_{n=1}^{N_j} \hat{\mathcal{C}}_n^{i}\right)$. Here, $\hat{\mathcal{C}}_n^{i\setminus(i-1)}$ represents the model's (binary) predictions of *unique concepts introduced* in experience i, while $\hat{\mathcal{C}}_n^{i}$ represents the model's (binary) predictions of *all* concepts present in experience i.

Results: CL Performance. Table 1 shows our results on concept-based continual learning. Our approach outperforms all baselines, with significant margins on CIFAR-100 and ImageNet-100. This is done *without adding any additional parameters* to our model with newer experiences, whereas other methods require new parameters to incorporate new classes and concepts. We also observe significantly lower forgetting across experiences. These results show that our model readily incorporates knowledge about new concepts and classes while internally forming required concept-class associations, and also remembers these associations fairly well, when trained on new experiences. We also find that CL algorithms which explicitly replay labels (ER and DER++) perform better across all methods.

Results: SL Performance. To see how MuCIL fares in standard classification settings, we evaluate it in a full-data setting on the same three datasets. We find that MuCIL considerably outperforms the next closest baseline on the CUB dataset, indicating that it is highly effective when used to differentiate between fine-grained classes. It also achieves comparable performance on ImageNet-100 and CIFAR-100, even though this setting is not our focus.

Results: CCRF. We show the $CCRF$ metric values for MuCIL versus a jointly trained CBM in Table 3. We take the concept neurons (the bottleneck layer in the case of CBM-J) and train a linear layer on the different class set-concept set pairs required for the computation of the metric, as in Eqn 3. We see that MuCIL is able to preserve significantly more information in its concept sets across experiences, as compared to CBM-J. Particulary on ImageNet100, we see an accuracy drop of over 14% on average when training the classification layer on logits obtained from a bottleneck of a future experience. This indicates that CBM-J is unable to preserve concept-class relationships of a given experience after training on future experiences. In contrast, MuCIL only results in an accuracy drop of 1.7%, indicating that concepts retain most of their information even after the model has

Method	**CIFAR-100**	**CUB**	**INet-100**
CBM-J	0.7868	0.7231	0.7773
CBM-S	0.5712	0.6932	0.4265
Label-Free	0.6431	0.7413	0.7818
LaBo	**0.8572**	0.7015	**0.8506**
MuCIL (Ours)	**0.8567**	**0.8401**	0.8466

Table 2: Classification performance of different methods in the full-data (single experience) setting.

Method	**CIFAR-100**	**CUB**	**INet-100**
CBM-J	0.0929	0.0362	0.1450
ICIAP-J	0.0941	0.0362	0.1632
MuCIL (Ours)	**0.0444**	**0.0099**	**0.0172**

Table 3: CCRF of CBM-J and MuCIL on benchmark datasets over 5 experiences with buffer size of 2000.

been trained on future experiences. We find that ICIAP performs even worse, which we attribute to the fact that it uses the entire concept set in all experiences. This causes concepts of future experiences to first learn spurious semantics and later modify them.

Visualization of CCRF. Table 3 computed using Eqn 3 showcases the ability of **MuCIL** to preserve concept-class relationships. The same formulation can be used to also study experience-level CCRF. Fixing t to some value gives us the form $CCRF(t) = \frac{1}{t-1}\sum_{k=t-1}^{1} LA(k, \mathcal{Y}^{k\setminus k-1}, \mathcal{C}^k) - LA(t, \mathcal{Y}^{k\setminus k-1}, \mathcal{C}^k)$, which represents the average CCRF value for experience t across previous concept set-class set pairs. This allows us to study how the concept-class relationships degrade over experiences. We show a visualization of this in Figure 3. In the figure, we see that the relationships learned by the CBM model consistently degrade with experiences, while the relationships learned by our framework are preserved throughout.

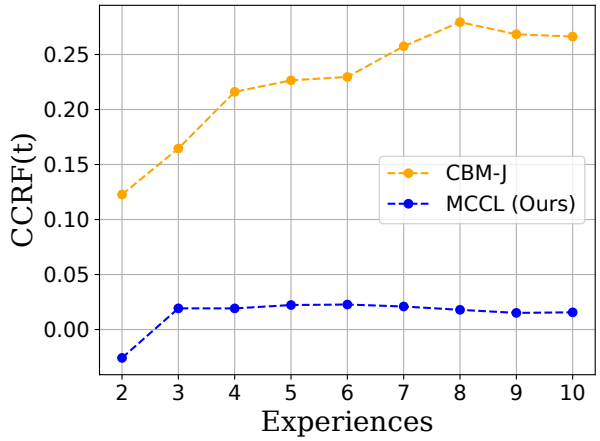

Figure 3: Visualization of Concept-Class Relationship Forgetting on ImageNet-100 across 10 experiences, for CBM and MuCIL. While the relationship between concepts and classes deteriorates over new experiences, our method maintains forgetting at the same level.

Results: ACR. We present a visualization of ACRs across 10 experiences for a CBM, a Label-Free CBM and MuCIL model in Fig. 4. We see that the CBM is unable to effectively incorporate concepts that appear in later experiences and relies heavily on early concept sets. Label-Free CBM activates a given concept similarly across experiences, leading to poor explainability in terms of concepts required for a specific experience. MuCIL has a strong diagonal ACR matrix, showing that it strongly activates concepts that appear with (and therefore explain) a set of classes. It also appropriately activates earlier concepts, particularly fundamental concepts from experience one that are shared across future experiences. We see some bias toward the concept set introduced in the final experience, a common problem in most CL settings (Buzzega et al. 2021; Mai et al. 2021). Addressing this can an interesting direction of future work.

Qualitative Results: *Visual Grounding and Attributions.*

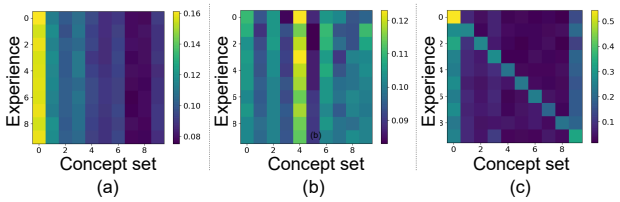

Figure 4: Comparison of ACR matrices for (a) Standard CBM, (b) Label-Free CBM, and (c) MuCIL for CIFAR-100 over 10 experiences. Each row represents the ACR scores for the class sets of the corresponding experience, with each column representing the concept set of different experiences.

Dataset	**FAA** w/ CR	**LA** w/ CR	**FAA** w/o CR	**LA** w/o CR
CIFAR-100	0.7022	0.7650	0.6722	0.4511
CUB200	0.8137	0.7914	0.7844	0.1382
ImageNet-100	0.7970	0.7722	0.7903	0.5903

Table 4: Linear layer training on top of concept neurons; CR = concept-augmented experience replay; LA = Accuracy of Linear Classifier trained on concept neurons.

Our method can be directly used as a post-hoc analysis tool by computing visualizations of attention maps learned by our model. Fig. 5 shows one such result where our method has sharp focus on specific parts of the image corresponding to a concept. This supports the achievement of the objectives of our work. In contrast, models without concept grounding have diffuse heatmaps without a clear focus on user-defined concepts (Margeloiu et al. 2021).

Figure 5: **Visual grounding of part-based concepts:** Qualitative results for localizing visual concepts using MuCIL versus when localizing the same concepts using GradCAM on CBMs. More results provided in the Appendix (§A3).

Concept Replay Evaluation: Results for $LA(T, \mathcal{Y}^T, \mathcal{C}^T)$, represented simply as LA, are shown in Table 4 with and without replaying concept labels from the buffer in addition to class labels (w/ CR and w/o CR respectively). Evidently, the concepts remain significantly more informative when using this enhanced Concept Replay. CR also enhances overall performance in terms of FAA, as shown in the same table. We use CR with all benchmarked algorithms and baselines.

***Effects of $\mathcal{L}_{WBCE}$ and $\mathcal{L}_G$.*:** We study the effects of these

losses on performance by varying their weights λ_1 and λ_2 in Eqn 2. A low λ_1 results in a significant drop in performance in LA scores. This indicates that the model forgets about relationships that exist between concept embeddings and classes, when $\mathcal{L}_{WBCE}$'s effect is reduced. Low values of λ_2 result in very low similarities between corresponding vectors in $\mathcal{C}^i$ and $\mathcal{C}'^{(i)}$, indicating that the concept embeddings lose semantic information encoded by their textual anchors when $\mathcal{L}_G$ has a reduced effect. We empirically found $\lambda_1 = 5$ and $\lambda_2 = 10$ to give the best performance overall in terms of FAA, LA and grounding similarity, with $LA = 0.7722$ and cosine similarity 0.998. A detailed table showing the effects of these terms is in the Appendix (§A3).

***Comparison with prototype-based methods*:** We compare our method's performance to ICICLE (Rymarczyk et al. 2023) - a prototype-based approach for interpretable CIL - on CUB200 with 10 experiences. Our method significantly outperforms ICICLE – we achieve an FAA score of 0.7606 *without* experience IDs using 500 exemplars, whereas ICICLE achieves an FAA score of 0.602 *with* experience IDs, and a score of 0.185 without experience IDs. We note that the two methods are different fundamentally in terms of the methodology; we provide the numbers here for completeness of understanding of our method's performance.

Evaluating concepts using interventions: To study the goodness of the learned concepts, we consider samples that are misclassified by a linear layer trained on top of concept neurons and perform interventions on the wrongly predicted concepts using the mechanism in (Koh et al. 2020). Performing interventions on a few key misclassified concepts converts the classification to a correct class label. This highlights the goodness of semantics of the learned concepts, and its impact on classification. Qualitative results for interventions are provided in Figure 6 and the Appendix (§A3).

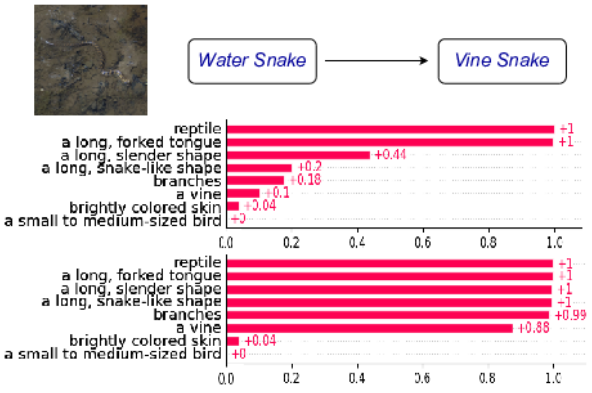

Figure 6: Manual interventions on concepts: We identify concepts that are incorrectly labeled, and modify them based on the image semantics, this results in correct classification.

Alignment between Similar Concepts. Ideally, if two concepts are similar in nature (and thus have similar text embeddings), their multimodal embeddings should maintain this similarity irrespective of whether or not the concepts co-occur in a given input. To verify if such inter-concept semantics are preserved, we compute pairwise concept embedding similarities for INet100 and the pairwise multimodal embedding similarities averaged over test samples. The elementwise squared errors between these matrices have a mean of 0.019, indicating that two similar concepts would likely have similar multimodal embeddings. This shows that our approach preserves relative semantics of concepts.

Evaluation with Other Variants of Attention. Recent attempts (Katharopoulos et al. 2020; Vyas, Katharopoulos, and Fleuret 2020; Shen et al. 2021; Wang et al. 2020; Kitaev, Kaiser, and Levskaya 2019) have been made to improve the computational efficiency of transformers. To study the ability of our framework in incorporating new advancements in the transformer architecture, we evaluated a variant of our method that implemented **MuCIL** with Linear Attention. We use transformer blocks featuring linear attention proposed in (Katharopoulos et al. 2020) in our multimodal encoder. We evaluate this variant of our model across benchmark datasets for 5 experiences with 2000 exemplars. The results in Table 5 show that even with linear attention, it achieves comparable performance to vanilla attention on all three benchmarks. This could help make our method even more efficient, while retaining its performance. We believe this could be an interesting direction of future work.

	CIFAR-100		CUB		ImageNet-100	
Method	FAA	AF	FAA	AF	FAA	AF
MuCIL	**0.70**	**0.30**	**0.81**	0.06	**0.80**	0.09
MuCIL -L	0.69	0.31	**0.82**	**0.05**	**0.80**	**0.08**

Table 5: Performance comparison of MuCIL vs. MuCIL-Linear over 5 experiences with 2000 exemplars, across different datasets.

5 Conclusions and Future Work

In this work, we study a new paradigm of continual learning for interpretable models where both new classes and new concepts are introduced across experiences. We show that existing works suffer from degradation in concept-class relationships in such settings. We propose a method that adapts the transformer architecture in vision-language encoders to include concept embeddings, which are anchored to natural language concepts. Through a set of carefully designed loss terms, our approach can not only classify reliably in a CL setting, but can also specify the human-understandable concepts used for the classification. We evaluate our method on three benchmark datasets and introduce new metrics to study the efficacy of concepts in our framework. Our qualitative and quantitative results show the significant promise of the proposed method. Beyond being a method for concept-based CL, we believe that our efforts can further open up the direction of inherently interpretable CL models in the community. Analysis of CL methods that do not directly replay past labels and the refinement of our architecture for better CL would be interesting directions of future work.

Acknowledgments

This work was partly supported by the Reliance Postgraduate Scholarship, Prime Minister's Research Fellowship (PMRF) program and funding support from Adobe. We are grateful to the anonymous reviewers for their valuable feedback, which improved the presentation of the paper.

References

Alvarez-Melis, D.; and Jaakkola, T. S. 2018. Towards Robust Interpretability with Self-Explaining Neural Networks. arXiv:1806.07538.

Barker, M.; Collins, K. M.; Dvijotham, K.; Weller, A.; and Bhatt, U. 2023. Selective Concept Models: Permitting Stakeholder Customisation at Test-Time. arXiv:2306.08424.

Benitez, R.; et al. 2023. Ante-Hoc Generation of Task-Agnostic Interpretation Maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3763–3768.

Buzzega, P.; Boschini, M.; Porrello, A.; Abati, D.; and Calderara, S. 2020. Dark experience for general continual learning: a strong, simple baseline. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781713829546.

Buzzega, P.; Boschini, M.; Porrello, A.; and Calderara, S. 2021. Rethinking Experience Replay: a Bag of Tricks for Continual Learning. In *2020 25th International Conference on Pattern Recognition (ICPR)*, 2180–2187. Los Alamitos, CA, USA: IEEE Computer Society.

Chattopadhay, A.; Sarkar, A.; Howlader, P.; and Balasubramanian, V. N. 2018. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, 839–847. IEEE.

Chattopadhyay, A.; Slocum, S.; Haeffele, B. D.; Vidal, R.; and Geman, D. 2022. Interpretable by design: Learning predictors by composing interpretable queries. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 45(6): 7430–7443.

Chaudhry, A.; Ranzato, M.; Rohrbach, M.; and Elhoseiny, M. 2019. Efficient Lifelong Learning with A-GEM. In *International Conference on Learning Representations*.

Chen, L.; Chen, J.; Hajimirsadeghi, H.; and Mori, G. 2020. Adapting grad-cam for embedding networks. In *proceedings of the IEEE/CVF winter conference on applications of computer vision (CVPR)*, 2794–2803.

Chen, Z.; Bei, Y.; and Rudin, C. 2020. Concept whitening for interpretable image recognition. *Nature Machine Intelligence*, 2(12): 772–782.

Collins, K. M.; Barker, M.; Espinosa Zarlenga, M.; Raman, N.; Bhatt, U.; Jamnik, M.; Sucholutsky, I.; Weller, A.; and Dvijotham, K. 2023. Human Uncertainty in Concept-Based AI Systems. In *Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society*, 869–889.

Dabkowski, P.; and Gal, Y. 2017. Real time image saliency for black box classifiers. *Advances in Neural Information Processing Systems (NeurIPS)*, 30.

Fong, R. C.; and Vedaldi, A. 2017. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE international conference on computer vision*, 3429–3437.

Hadsell, R.; Rao, D.; Rusu, A. A.; and Pascanu, R. 2020. Embracing change: Continual learning in deep neural networks. *Trends in cognitive sciences*, 24(12): 1028–1040.

Jethani, N.; Sudarshan, M.; Covert, I. C.; Lee, S.-I.; and Ranganath, R. 2021. Fastshap: Real-time shapley value estimation. In *International Conference on Learning Representations (ICLR)*.

Katharopoulos, A.; Vyas, A.; Pappas, N.; and Fleuret, F. 2020. Transformers are rnns: Fast autoregressive transformers with linear attention. 5156–5165. PMLR.

Kazhdan, D.; Dimanov, B.; Jamnik, M.; Liò, P.; and Weller, A. 2020. Now you see me (CME): concept-based model extraction. *arXiv preprint arXiv:2010.13233*.

Kim, I.; Kim, J.; Choi, J.; and Kim, H. J. 2023. Concept Bottleneck with Visual Concept Filtering for Explainable Medical Image Classification. *arXiv preprint arXiv:2308.11920*.

Kitaev, N.; Kaiser, L.; and Levskaya, A. 2019. Reformer: The Efficient Transformer. In *International Conference on Learning Representations (ICLR)*.

Koh, P. W.; Nguyen, T.; Tang, Y. S.; Mussmann, S.; Pierson, E.; Kim, B.; and Liang, P. 2020. Concept bottleneck models. 5338–5348. PMLR.

Mai, Z.; Li, R.; Kim, H.; and Sanner, S. 2021. Supervised Contrastive Replay: Revisiting the Nearest Class Mean Classifier in Online Class-Incremental Continual Learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 3589–3599.

Marconato, E.; Bontempo, G.; Teso, S.; Ficarra, E.; Calderara, S.; and Passerini, A. 2022. Catastrophic forgetting in continual concept bottleneck models. 539–547. Springer.

Margeloiu, A.; Ashman, M.; Bhatt, U.; Chen, Y.; Jamnik, M.; and Weller, A. 2021. Do concept bottleneck models learn as intended? *arXiv preprint arXiv:2105.04289*.

Montavon, G.; Binder, A.; Lapuschkin, S.; Samek, W.; and Müller, K. 2019. Explainable AI: interpreting, explaining and visualizing deep learning. *Spring er LNCS*, 11700.

Nauta, M.; Trienes, J.; Pathak, S.; Nguyen, E.; Peters, M.; Schmitt, Y.; Schlötterer, J.; van Keulen, M.; and Seifert, C. 2023. From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable ai. *ACM Computing Surveys*, 55(13s): 1–42.

Oikarinen, T.; Das, S.; Nguyen, L. M.; and Weng, T.-W. 2023. Label-free Concept Bottleneck Models.

Petsiuk, V.; Das, A.; and Saenko, K. 2018. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. 8748–8763. PMLR.

Rebuffi, S.-A.; Kolesnikov, A.; Sperl, G.; and Lampert, C. H. 2017. iCaRL: Incremental Classifier and Representation Learning. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 5533–5542.

Rigotti, M.; Miksovic, C.; Giurgiu, I.; Gschwind, T.; and Scotton, P. 2021. Attention-based interpretability with concept transformers. In *International Conference on Learning Representations (ICLR)*.

Rudin, C. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5): 206–215.

Rymarczyk, D.; van de Weijer, J.; Zieliński, B.; and Twardowski, B. 2023. ICICLE: Interpretable Class Incremental Continual Learning. *arXiv preprint arXiv:2303.07811*.

Sattarzadeh, S.; Sudhakar, M.; Plataniotis, K. N.; Jang, J.; Jeong, Y.; and Kim, H. 2021. Integrated grad-cam: Sensitivity-aware visual explanation of deep convolutional networks via integrated gradient-based scoring. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1775–1779. IEEE.

Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, 618–626.

Shen, Z.; Zhang, M.; Zhao, H.; Yi, S.; and Li, H. 2021. Efficient attention: Attention with linear complexities. 3531–3539.

Shin, S.; Jo, Y.; Ahn, S.; and Lee, N. 2023. A closer look at the intervention procedure of concept bottleneck models. *arXiv preprint arXiv:2302.14260*.

Sokol, K.; and Flach, P. 2021. Explainability is in the mind of the beholder: Establishing the foundations of explainable artificial intelligence. *arXiv preprint arXiv:2112.14466*.

Steinmann, D.; Stammer, W.; Friedrich, F.; and Kersting, K. 2023. Learning to Intervene on Concept Bottlenecks. arXiv:2308.13453.

Sundararajan, M.; and Najmi, A. 2020. The many Shapley values for model explanation. In *International Conference on Machine Learning (ICML)*, 9269–9278. PMLR.

Vilone, G.; and Longo, L. 2021. Notions of explainability and evaluation approaches for explainable artificial intelligence. *Information Fusion*, 76: 89–106.

Vyas, A.; Katharopoulos, A.; and Fleuret, F. 2020. Fast transformers with clustered attention. *Advances in Neural Information Processing Systems (NeurIPS)*, 33: 21665–21674.

Wang, L.; Zhang, X.; Su, H.; and Zhu, J. 2023. A comprehensive survey of continual learning: Theory, method and application. *arXiv preprint arXiv:2302.00487*.

Wang, R.; Wang, X.; and Inouye, D. I. 2020. Shapley Explanation Networks. In *International Conference on Learning Representations (ICLR)*.

Wang, S.; Li, B. Z.; Khabsa, M.; Fang, H.; and Ma, H. 2020. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*.

Yan, A.; Wang, Y.; Zhong, Y.; Dong, C.; He, Z.; Lu, Y.; Wang, W.; Shang, J.; and McAuley, J. 2023. Learning Concise and Descriptive Attributes for Visual Recognition. *arXiv preprint arXiv:2308.03685*.

Yang, Y.; Panagopoulou, A.; Zhou, S.; Jin, D.; Callison-Burch, C.; and Yatskar, M. 2023. 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*, 19187–19197.

Yvinec, E.; Dapogny, A.; Cord, M.; and Bailly, K. 2022. Singe: Sparsity via integrated gradients estimation of neuron relevance. *Advances in Neural Information Processing Systems (NeurIPS)*, 35: 35392–35403.