Choose Your Neuron: Incorporating Domain Knowledge through Neuron-Importance

Ramprasaath R. Selvaraju[1*] **Prithvijit Chattopadhyay**[1*] **Mohamed Elhoseiny**[2]
Tilak Sharma[3] **Dhruv Batra**[1,2] **Devi Parikh**[1,2] **Stefan Lee**[1]
[1]Georgia Institute of Technology [2]Facebook AI Research [3]Facebook
{ramprs,prithvijit3,dbatra,parikh,steflee}@gatech.edu
{elhoseiny,tilaksharma,dbatra,parikh}@fb.com

Abstract

Individual neurons in convolutional neural networks supervised for image-level classification tasks have been shown to implicitly learn semantically meaningful concepts ranging from simple textures and shapes to whole or partial objects – forming a "dictionary" of concepts acquired through the learning process. In this work we introduce a simple, efficient zero-shot learning approach based on this observation. Our approach, which we call Neuron Importance-Aware Weight Transfer (NIWT), learns to map domain knowledge about novel classes onto this dictionary of learned concepts and then optimizes for network parameters that can effectively combine these concepts – essentially learning classifiers by discovering and composing learned semantic concepts in deep networks. In addition to demonstrating improvements on the generalized zero-shot learning benchmark, we show that by having an additional component which requires grounding neuron-level concepts in human-interpretable semantics, we can also interpret the decisions made by the learned classifiers at a fine-grained level of neurons. Our code is available at https://github.com/ramprs/neuron-importance-zsl.

1 Introduction

While deep neural networks have pushed the boundaries of several tasks in the past few years, one major caveat associated with this model class is the inability to generalize well from a few examples like humans can. To close this gap, the task of learning deep classifiers for unseen classes from external domain knowledge alone – termed zero-shot learning (ZSL) – has been the topic of increased interest within the community [16, 15, 10, 18, 25, 31, 27, 2, 11, 3, 21, 5, 13].

As humans, much of the way we acquire and transfer knowledge about novel concepts is in reference to or via composition of concepts which are already known. For instance, upon hearing that *"A Red Bellied Woodpecker is a small, round bird with a white breast, red crown, and spotted wings."*, we can compose our understanding of colors and birds to imagine how we might distinguish such an animal from other birds. While individual neurons in deep networks have been shown to learn localized, semantic concepts, applying a similar compositional learning strategy for deep neural networks has proven challenging. In addition, such intermediate concepts captured by units within a network lack referable groundings – *i.e.* even if a network contains units sensitive to *"white breast"* and *"red crown"*, there is no explicit mapping of these neurons to the relevant language name or description. Prior work in interpretability has adopted crowd-sourcing "neuron names" to discover these groundings [4]. However, this annotation process is model dependent which makes it expensive and impractical. Moreover, even if given perfect "neuron names", it is an open question how to leverage this neuron-level descriptive supervision to train novel classifiers.

[*] The first two authors contributed equally. Published as a conference paper at ECCV 2018.

Preprint. Work in progress.

Many existing zero-shot learning approaches make use of deep features to learn joint embeddings with class descriptions [28, 1, 3, 5, 19, 8, 9, 7]. These higher-level features collapse many underlying concepts in the pursuit of class discrimination; consequentially, accessing lower-level concepts and recombining them in new ways to represent novel classes is difficult. Mapping class descriptions to lower-level activations directly on the other hand is complicated by the high intra-class variance of activations due to both spatial and visual differences within instances of a class. We address these challenges by explicitly grounding class descriptions (including attributes and free-form text) to the *importance* of lower-layer neurons to final network decisions [22].

In our approach, which we call Neuron Importance-based Weight Transfer (NIWT), we learn a mapping between class-specific domain knowledge and the importances of individual neurons within a deep network – using images (to compute neuron-importance) and corresponding domain knowledge representation(s) of training classes. We then use this learned mapping to predict neuron importances from knowledge about unseen classes and optimize classification weights such that the resulting network aligns with the predicted importances. In other words, based on domain-knowledge of the unseen categories, we can predict which low-level neurons should matter in the final classification decision. We can then learn network weights such that the neurons predicted to matter actually do contribute to the final decision. In this way, we connect the description of a previous unseen category to weights of a classifier that can predict this category at test time – all without having seen a single image from this category. Moreover, having an additional component which requires grounding neuron-level concepts in human-interpretable semantics, we can also interpret the decisions made by the learned classifiers at a fine-grained level of neurons.

Generalized Zero-Shot Learning. We focus on the challenging generalized zero-shot (GZSL) learning setting as the test-bed. More concretely, the problem setup is explained as follows. Consider a dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^N$ comprised of example input-output pairs from a set of *seen classes* $\mathcal{S} = \{1, \dots, s\}$ and *unseen classes* $\mathcal{U} = \{s{+}1, \dots, s{+}u\}$. For convenience, we use the subscripts $\mathcal{S}$ and $\mathcal{U}$ to indicate subsets corresponding to seen and unseen classes respectively, *e.g.* $\mathcal{D}_S = \{(x_i, y_i) \mid y_i \in \mathcal{S}\}$. Further, assume there exists domain knowledge $\mathcal{K} = \{k_1, ..., k_{s+u}\}$ corresponding to each class (*e.g.* class level attributes or natural language descriptions). Concisely, the goal of generalized zero-shot learning is then to learn a mapping $f : \mathcal{X} \rightarrow \mathcal{S} \cup \mathcal{U}$ from the input space $\mathcal{X}$ to the combined set of seen and unseen class labels using only the domain knowledge $\mathcal{K}$ and instances $\mathcal{D}_{\mathcal{S}}$ belonging to the seen classes. At inference, GZSL is made more challenging by dropping the unrealistic assumption that test instances are known a priori to be from unseen classes (unlike standard ZSL) – as such methods are evaluated based on performance on both seen as well as unseen classes.

Contributions. Concretely, we make the following contributions in this work:

- A zero-shot learning approach that involves grounding class descriptions into neuron(s) within a deep network and then optimizing unseen classifier weights to effectively combine the grounded concepts. In addition to being able to handle arbitrary forms of domain knowledge (captions and attributes), our approach also demonstrates improvements on the generalized zero-shot learning benchmark on CUB and AWA2.
- Our method is capable of explaining its zero-shot predictions with human-interpretable semantics from attributes. We show how inverse mappings from neuron importance to domain knowledge can also be learned to provide interpretable visual and textual explanations for the decisions made by newly learned classifiers for seen and unseen classes.

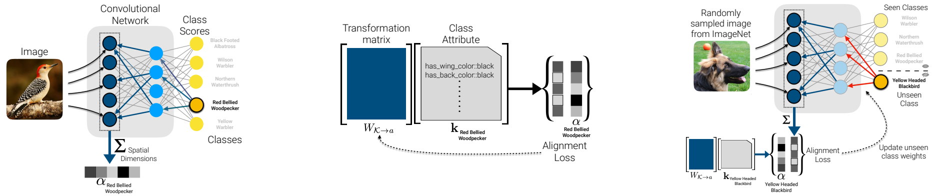
2 Neuron Importance-Aware Weight Transfer (NIWT)

At a high level, we map free-form domain knowledge to neurons within a deep network and then learn classifiers based on novel class descriptions with respect to the prior groundings. Concretely, this consists of three steps: (1) estimating the importance of individual neuron(s) at a fixed layer w.r.t. the decisions made by the network for the seen classes (see Figure 1a), (2) learning a mapping between domain knowledge and these neuron-importances (see Figure 1b), and (3) optimizing classifier weights with respect to predicted neuron-importances for unseen classes (see Figure 1c).

2.1 Class-dependent Neuron Importance

Unlike the salient concepts captured by class-descriptions about the content of images – for example, describing the coloration and shape of a bird's head – visually discriminative concepts captured by deep classifiers lack human-interpretable groundings. We identify neurons in a network corresponding to discriminative concepts before aligning them with domain knowledge in Section 2.2.

Consider a deep neural network $\texttt{NET}_{\mathcal{S}}(\cdot)$ trained for classification which predicts scores $\{o_c \mid c \in \mathcal{S}\}$ for seen classes $\mathcal{S}$. We note that while other measures of neuron importance have been proposed [29, 14] in various contexts; we interpret measure of a neuron n's importance to the final score



(a) (b) (c)

Figure 1: Our Neuron Importance-Aware Weight Transfer (NIWT) approach can be broken down in to three stages. a) class-specific neuron importances are extracted for seen classes at a fixed layer, b) a linear transform is learned to project free-form domain knowledge to these extracted importances, and c) weights for new classifiers are optimized such that neuron importances match those predicted by this mapping for unseen classes.

o_c is simply the gradient of o_c with respect to the neuron's activation a^n (where n indexes the channel dimension). For networks containing convolutional units, we follow [22] and simply compute importance as the mean gradient (along spatial dimensions), writing the neuron importance α_c^n as

$$\alpha_c^n = \overbrace{\frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W}}^{\text{global average pooling}} \underbrace{\frac{\partial o_c}{\partial a_{ij}^n}}_{\text{gradients via backprop}} \tag{1}$$

where $a_{i,j}^n$ is the activation of neuron n at spatial position i, j. For a given input, the importance of every neuron in the network can be computed for a given class via a single backward pass followed by a global average pooling operation for convolutional units. In practice, we focus on α's from single layers in the network in our experiments. Some notable properties that we observe and eventually leverage by expressing neuron-importance in the above manner are:

1. **Intra-class Consistency.** We find gradient-based importance scores to be quite consistent across images of the same class despite the visual variation between instances, and likewise to correlate poorly across classes.
2. **End-to-End Differentiability.** This measure of neuron-importance is fully differentiable with respect to model parameters which we use to learn novel classifiers with gradient methods.

2.2 Mapping Domain Knowledge to Neurons

Without loss of generality, consider a single layer L within $\texttt{NET}_{\mathcal{S}}(\cdot)$. Given an instance $(x_i, y_i) \in \mathcal{D}_{\mathcal{S}}$, let $\mathbf{a}_c = \{\alpha_c^n \mid n \in L\}$ be a vector of importances computed for neurons in L with respect to class c when x_i is passed through the network. After computing the importance vector $\mathbf{a}_{y_i}$ for each seen class instance (x_i, y_i) and pairing it with the domain knowledge representation k_{y_i} of the corresponding class, we learn a simple (one-to-many) linear mapping $W_{\mathcal{K}\to a}$ from k_{y_i} to $\mathbf{a}_{y_i}$ – aligning interpretable semantics with individual neurons. As importances are gradient based, we penalize errors in the predicted importances based on cosine distance – emphasizing alignment over magnitude. We minimize the cosine distance loss

$$\mathcal{L}(W_{\mathcal{K}\to a}, \mathbf{a}_{y_i}, \mathbf{k}_{y_i}) = 1 - \frac{(W_{\mathcal{K}\to a} \cdot \mathbf{k}_{y_i}) \cdot \mathbf{a}_{y_i}}{\|W_{\mathcal{K}\to a} \cdot \mathbf{k}_{y_i}\| \, \|\mathbf{a}_{y_i}\|}, \tag{2}$$

via gradient descent to estimate $W_{\mathcal{K}\to a}$. We stop training when average rank-correlation of predicted and true importance vectors stabilizes for a set of held out validation classes from $\mathcal{S}$.

2.3 Neuron Importance to Classifier Weights

Here we use predicted importances to learn classifiers for the unseen classes. We modify $\texttt{NET}_{\mathcal{S}}$ to extend the output space to include unseen classes ($\texttt{NET}_{\mathcal{S}\cup\mathcal{U}}$) – expanding the final fully-connected layer to include additional neurons with weight vectors $\mathbf{w}^1, \ldots, \mathbf{w}^u$ for the unseen classes – initialized from a multivariate normal distribution with parameters estimated from the seen class weights.

Utilizing the learned mapping $W_{\mathcal{K}\to A}$ and given access to unseen class domain knowledge $\mathcal{K}_{\mathcal{U}} = \{\mathbf{k}_c \mid c \in \mathcal{U}\}$, we predict neuron importances for the unseen classes as $\mathbf{a}_c = W_{\mathcal{K}\to a}\mathbf{k}_c$. Further, given an input, the neuron importances for the unseen classes estimated from $\texttt{NET}_{\mathcal{S}\cup\mathcal{U}}$ – $\hat{\mathbf{a}}^c$ – are functions of the corresponding weight parameters $\mathbf{w}_c$. Intuitively, the set $A_{\mathcal{U}} = \{\mathbf{a}_1, ..., \mathbf{a}_u\}$ could serve as a source of *noisy* ground truth with which the importance vectors predicted with respect to the network parameters ($\hat{\mathbf{a}}^c$) should align. We supervise $\hat{\mathbf{a}}^c$ with $\mathbf{a}_c$ through the cosine-distance loss and optimize for all $\{\mathbf{w}_c \mid c \in \mathcal{U}\}$ via gradient-descent. To further ensure that the scale of optimal set of weights for the unseen classes does not deviate significantly from the seen class – to avoid eventual bias in the logits across $\mathcal{S}$ and $\mathcal{U}$, we add we introduce a L_2 regularization term which constrains the learned unseen weights to be a similar scale as the mean of seen weights $\overline{\mathbf{w}}_{\mathcal{S}}$. We write the final objective as

$$\mathcal{L}(\mathbf{w}_c, \hat{\mathbf{a}}_c, \mathbf{a}_c) = \mathcal{L}(\mathbf{w}_c, \mathbf{a}_c) = 1 - \frac{\hat{\mathbf{a}}_c \cdot \mathbf{a}_c}{\|\hat{\mathbf{a}}_c\|\;\|\mathbf{a}_c\|} + \lambda\|\mathbf{w}_c - \overline{\mathbf{w}}_{\mathcal{S}}\|,^{\dagger} \tag{3}$$

where λ controls the strength of this regularization. In practice, we find training to be robust to a wide-range of λ values. As observed importances $\hat{\mathbf{a}}^c$ are themselves computed from network gradients, updating weights based on this loss requires computing a Hessian-vector product; however, this is relatively efficient as the number of weights for each unseen class is small and independent of those for other classes. For ablations characterizing the behavior of the optimization process with respect to different components, refer to [23].

Training Images. Note that to perform the above optimization, we need to pass images through the network to compute importance vectors. Recall that importances are only weakly correlated with image features and since they can be computed for any of the unseen classes irrespective of the input image class – we find simply inputing images with natural statistics to be sufficient (pairing random images from ImageNet [6] to form random tuples $(\hat{\mathbf{a}}_c, \mathbf{k}_c)$).

3 Experiments

			AWA2 [28]			CUB [26]		
		Method	$\mathtt{Acc}_{\mathcal{U}}$	$\mathtt{Acc}_{\mathcal{S}}$	$\mathtt{H}$	$\mathtt{Acc}_{\mathcal{U}}$	$\mathtt{Acc}_{\mathcal{S}}$	$\mathtt{H}$
ResNet101 [12]	FT	ALE [2][1]	22.7	**75.1**	34.9	24.1	**60.8**	**34.5**
		Deep Embed. [30][1]	21.5	59.6	31.6	**24.7**	57.4	**34.5**
		NIWT-Attributes	**42.3**	38.8	**40.5**	20.7	41.8	27.7
		NIWT-Caption		N/A		22.1	25.7	23.8
VGG16 [24]	FT	ALE [2][1]	16.9	**91.5**	28.5	25.3	**62.6**	36.0
		Deep Embed. [30][1]	26.6	83.3	38.2	27.0	49.7	35.0
		NIWT-Attributes	**35.3**	75.5	**48.1**	**31.5**	44.9	**37.0**
		NIWT-Caption		N/A		15.9	46.5	23.6

Table 1: Generalized Zero-Shot Learning performances on the proposed splits [28] for AWA2 and CUB. We report class-normalized accuracies on seen and unseen classes and harmonic mean.

Datasets and Metrics. We conduct experiments on two datasets – Animals with Attributes 2 (AWA2) [28] and Caltech-UCSD Birds 200 (CUB) [26]. Both datasets provides access to continuous class-level attributes. In addition, CUB also has 10 human captions [20] associated with each image. For both datasets, we use the GZSL splits proposed in [28] which ensure that no unseen class occurs within the ImageNet [6] dataset (used for pre-training networks). We evaluate our approach using class-normalized accuracy computed over both seen and unseen classes – breaking the results down into unseen accuracy $\mathtt{Acc}_{\mathcal{U}}$, seen accuracy $\mathtt{Acc}_{\mathcal{S}}$, and the harmonic mean between them $\mathtt{H}$.

Models. We experiment with ResNet101 [12] and VGG16 [24] models pretrained on ImageNet [6] and fine-tuned on the seen classes. Refer to [23] for more details and ablations.

NIWT Settings. To learn the mapping $W_{\mathcal{K}\to a}$ we hold out five seen classes and stop optimization when rank correlation between observed and predicted importances is highest. For attribute vectors, we use the class level attributes directly and for captions on CUB we use average word2vec embeddings[17] for each class. When optimizing for weights given importances, we stop when the loss fails to improve by 1% over 40 iterations. We choose values of λ, learning rate and the batch size by grid search on $\mathtt{H}$ for a disjoint set of validation classes sampled from the seen classes.

Baselines. We compare NIWT with two well-performing zero-shot learning approaches – ALE [2] and Deep Embed. [30]. While the former relies on learning compatibility functions for class labels and visual features the latter leverages deep networks, jointly aligning domain knowledge with deep features end-to-end. For the mentioned baselines, we utilize code provided by the authors and report results by directly tuning hyper-parameters on the test-set to convey an upper-bound of performance.

Results. Our results are summarized in Table 1. Some notable trends are,

1. **NIWT shows improvements on the generalized zero-shot learning benchmark.** For both datasets, NIWT-Attributes based on VGG establishes a new state of the art for harmonic mean (48.1% for AWA2 and 37.0% for CUB). For AWA2, this corresponds to a $\sim 10\%$ improvement over prior state-of-the-art which is based on deep feature embeddings.
2. **NIWT effectively grounds both attributes and free-form language.** We see strong performance both for attributes and captions across both networks (37.0% and 23.6% $\mathtt{H}$ for VGG

[†] Note that while optimizing this objective only the unseen class weights $\{\mathbf{w}_c \mid c \in \mathcal{U}\}$ are updated keeping everything else fixed. Computing $\hat{\mathbf{a}}_c$ via backpropagation renders $\hat{\mathbf{a}}_c = f(\mathbf{w}_c)$.

and 27.7% and 23.8% H for ResNet). Note that we use relatively simple, class-averaged representations for captioning which may contribute to the lower absolute performance.

Importance to Weight Input Images. We show performance with differing input images during weight optimization (random noise, ImageNet, and seen class images) in Table. 2. As expected, performance improves as input images more closely resemble the unseen classes; however, we note that learning occurs even with random noise images.

Sampling Mode	$\mathtt{Acc}_{\mathcal{U}}$	$\mathtt{Acc}_{\mathcal{S}}$	H
Random Normal	23.9	41.0	30.2
ImageNet	31.5	44.9	37.0
Seen-Classes	36.4	40.0	38.1

Table 2: Results by sampling images from different sets for NIWT-Attributes on VGG-CUB.

This observation suggests a way to adapt NIWT to the few-shot or continual learning setup. Specifically, instead of running the optimization process on random tuples $(\hat{\mathbf{a}}_c, \mathbf{k}_c)$, we could compute $\hat{\mathbf{a}}_c$ by passing available images of the *class* to be learnt. Scarcity of labeled samples for the novel class in the few-shot learning setup could be compensated for by the additional $\mathbf{a}_c$ supervision from the already learned $W_{\mathcal{K}\to a}$ mapping.

4 Explaining NIWT

Recall that NIWT involves an explicit learning component that requires us to ground the salient concepts for a class of interest in the important neurons. Here we explore how a similar grounding framework would allow us to expose the decision making process of the network for a given instance at a fine-grained level of neurons – where in addition to grounding the important neurons with respect to a prediction in human-interpretable semantics, we can also express the relative *visual* focus across said concepts. Fig. 2 demonstrates explanations for decisions made by the learned classifiers.

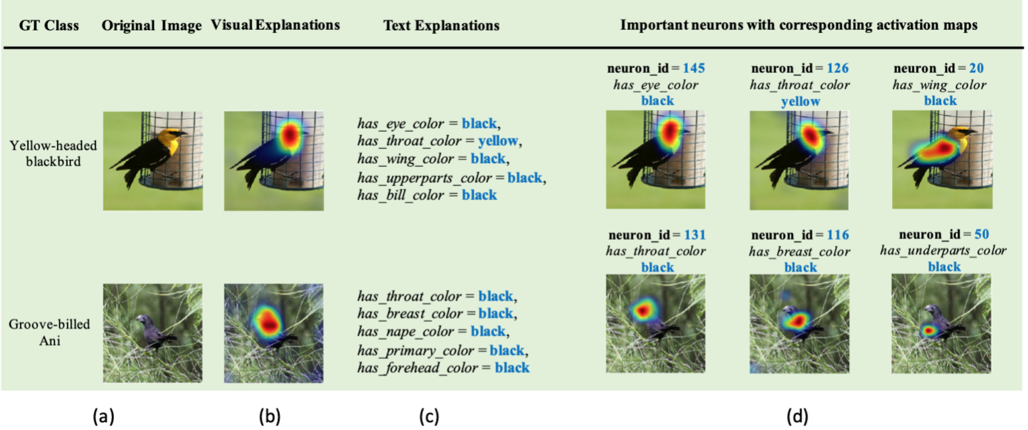


Figure 2: Explanations corresponding to the decisions made by the learned classifier for instances of the unseen classes on CUB.(a) the ground truth class and image, (b) visual explanations for the GT category, (c) textual explanations obtained using the inverse mapping $W_{a\to\mathcal{K}}$, (d) most important neurons for this decision, associated names and activation maps. Refer to [23] for more such examples.

Visual Explanations. Since learning classifiers for the novel classes via NIWT preserves the end-to-end differentiable nature of the network as a whole, any gradient-based interpretability technique (or otherwise) is applicable to provide support for decisions made at inference. We use Grad-CAM [22] on instances of unseen classes to provide explanations for the novel classifier learned via NIWT. Quantitative results on CUB – characterized by the mean fraction of Grad-CAM activation present inside the bounding box of the object of interest – indicate that the learned classifier is indeed capable of focusing on the relevant regions (0.80 ± 0.008 for seen and 0.79 ± 0.005 for unseen classes).

Textual Explanations. In our setup, we frame textual explanations as the problem of retrieving relevant attributes given the neurons important for a decision made by the network. We instantiate this as learning an inverse mapping $W_{a\to\mathcal{K}}$ – from importance scores $\mathbf{a}_c$ to associated domain-knowledge $\mathbf{k}_c$ – in a manner similar to Sec. 2.2. At inference, for a decision made by the learned classifier, we retrive the top-5 scoring attributes under $W_{a\to\mathcal{K}}$. Intuitively, a high scoring $\mathbf{k}_c$ retrieved via $W_{a\to\mathcal{K}}$ from a certain $\mathbf{a}_c$ emphasizes the relevance of that attribute for the corresponding class c. Quantitatively, we evaluate the fidelity of the retrieved explanations as the percentage of associated ground truth attributes for an instance in the retrieved top-k ones – 83.9% for CUB. Qualitatively, the retrieved explanations correlate well with the associated visual explanations as described above.

Neuron Names and Focus. Treating neuron-names as referable groundings of concepts captured by a deep convolutional network – we characterize the same as the top-1 textual explanation retrieved for a single-neuron under $W_{a \rightarrow \mathcal{K}}$. We instantiate this by feeding a one-hot encoded vector corresponding to the important neurons one at a time to $W_{a \rightarrow \mathcal{K}}$ and retrieving the top-scoring $\mathbf{k}_c$. In contrast to prior work, this one-shot process circumvents issues surrounding the collection of expensive annotations or performing any additional optimization on top of the same. In addition, observing the activation map of the associated neurons allows us to thereby characterize the *focus* of the neuron of interest.

References

[1] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for attribute-based classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 819–826, 2013. 2

[2] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *IEEE transactions on pattern analysis and machine intelligence*, 38(7):1425–1438, 2016. 1, 4

[3] Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2927–2936, 2015. 1, 2

[4] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Computer Vision and Pattern Recognition*, 2017. 1

[5] Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Synthesized classifiers for zero-shot learning. In *CVPR*, 2016. 1, 2

[6] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*, 2009. 4

[7] Mohamed Elhoseiny, Ahmed Elgammal, and Babak Saleh. Write a classifier: Predicting visual classifiers from unstructured text. *Ieee T Pattern Anal*, PP(99):1–1, 2017. 2

[8] Mohamed Elhoseiny, Babak Saleh, and Ahmed Elgammal. Write a classifier: Zero-shot learning using purely textual descriptions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2584–2591, 2013. 2

[9] Mohamed Elhoseiny, Yizhe Zhu, Han Zhang, and Ahmed Elgammal. Link the head to the "beak": Zero shot learning from noisy text description at part precision. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2

[10] Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1778–1785. IEEE, 2009. 1

[11] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *Advances in neural information processing systems*, 2013. 1

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4

[13] Elyor Kodirov, Tao Xiang, and Shaogang Gong. Semantic autoencoder for zero-shot learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1

[14] Sandeep Konam, Ian Quah, Stephanie Rosenthal, and Manuela Veloso. Understanding convolutional networks with apple : Automatic patch pattern labeling for explanation. *First AAAI/ACM Conference on AI, Ethics, and Society*, 2018. 2

[15] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3):453–465, 2014. 1

[16] Hugo Larochelle, Dumitru Erhan, and Yoshua Bengio. Zero-data learning of new tasks. In *AAAI*, volume 1, page 3, 2008. 1

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013. 4

[18] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. In *ICLR*, 2014. 1

[19] Ruizhi Qiao, Lingqiao Liu, Chunhua Shen, and Anton van den Hengel. Less is more: zero-shot learning from online textual documents with noise suppression. In *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on, year=2016*. 2

[20] Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. Learning deep representations of fine-grained visual descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 49–58, 2016. 4

[21] Bernardino Romera-Paredes and PHS Torr. An embarrassingly simple approach to zero-shot learning. In *Proceedings of The 32nd International Conference on Machine Learning*, pages 2152–2161, 2015. 1

[22] R. R. Selvaraju, A. Das, R. Vedantam, M. Cogswell, D. Parikh, and D. Batra. Grad-CAM: Why did you say that? Visual Explanations from Deep Networks via Gradient-based Localization. *ICCV*, 2017. 2, 3, 5

[23] Ramprasaath R Selvaraju, Prithvijit Chattopadhyay, Mohamed Elhoseiny, Tilak Sharma, Dhruv Batra, Devi Parikh, and Stefan Lee. Choose your neuron: Incorporating domain knowledge through neuron-importance. *arXiv preprint arXiv:1808.02861*, 2018. 4, 5

[24] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *ICLR*, 2015. 4

[25] Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng. Zero-shot learning through cross-modal transfer. In *Advances in neural information processing systems*, 2013. 1

[26] 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. 4

[27] Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. Latent embeddings for zero-shot classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 1

[28] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning - the good, the bad and the ugly. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2, 4

[29] Ruichi Yu, Ang Li, Chun-Fu Chen, Jui-Hsin Lai, Vlad I. Morariu, Xintong Han, Mingfei Gao, Ching-Yung Lin, and Larry S. Davis. NISP: pruning networks using neuron importance score propagation. *CVPR*, 2018. 2

[30] Li Zhang, Tao Xiang, and Shaogang Gong. Learning a deep embedding model for zero-shot learning. In *CVPR*, 2017. 4

[31] Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via semantic similarity embedding. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015. 1