Modeling Inter and Intra-Class Relations in the Triplet Loss for Zero-Shot Learning

Yannick Le Cacheux
CEA LIST
yannick.lecacheux@cea.fr

Hervé Le Borgne
CEA LIST
herve.le-borgne@cea.fr

Michel Crucianu
CEDRIC – CNAM
michel.crucianu@cnam.fr

Abstract

Recognizing visual unseen *classes,* i.e. *for which no training data is available, is known as Zero Shot Learning (ZSL). Some of the best performing methods apply the triplet loss to* seen *classes to learn a mapping between visual representations of images and attribute vectors that constitute class prototypes. They nevertheless make several implicit assumptions that limit their performance on real use cases, particularly with fine-grained datasets comprising a large number of classes. We identify three of these assumptions and put forward corresponding novel contributions to address them. Our approach consists in taking into account both inter-class and intra-class relations, respectively by being more permissive with confusions between similar classes, and by penalizing visual samples which are atypical to their class. The approach is tested on four datasets, including the large-scale ImageNet, and exhibits performances significantly above recent methods, even generative methods based on more restrictive hypotheses.*

1. Introduction

The task of zero-shot recognition, also referred to as *zero-shot learning* (ZSL) [1, 18, 20, 26], consists in classifying samples belonging to *unseen classes*, for which no training sample is available. Instead, the only training samples available are from different classes, called the *seen classes*; for each such class, a "semantic" representation is also provided along with the training samples. At testing time, the semantic representations of unseen classes can be used to make predictions. Although ZSL can be applied in many different contexts, it often refers to tasks where samples are of visual nature and the semantic representations, also called *class prototypes*, consist in vectors of attributes. For example, if one considers images of animals, attributes may be the number of legs or wings, the presence of fur or stripes.

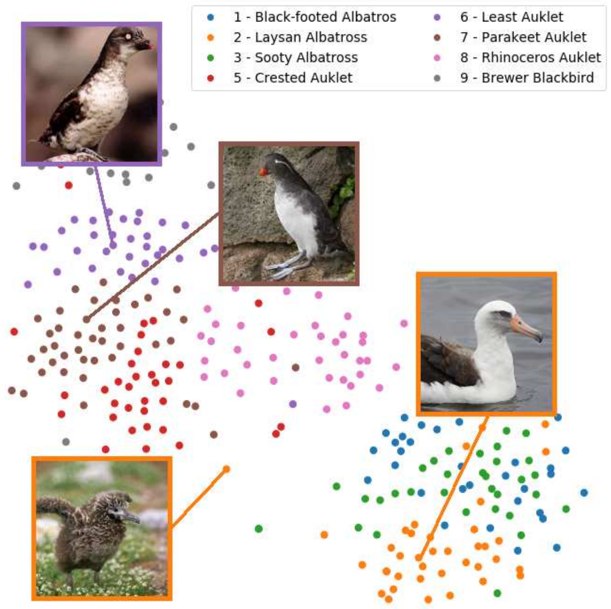

Figure 1. t-SNE [23] visualization of 300 samples from the first 8 training classes from CUB [37]. Classes 6 and 7 look quite similar and appear closer than classes 6 and 2, which are more dissimilar; the nestling from class 2 is far from the other samples in the same class. We propose to explicitly take these inter-class and intra-class relations into account. *Best viewed in color.*

In their seminal work, Lampert *et al.* [18] used a combination of simple classifiers to estimate the probabilities of attributes and classes given visual features. Other approaches [31, 32] use least-square regression models to predict one modality from the other. Triplet loss methods are closer to the final classification goal. They consider that each visual sample should be "much" more similar to the prototype corresponding to its class than to all the others. How "much" more similar is specified by a given fixed margin, included in an adequate penalty as described in more details in Sec. 3.1. DeViSE [12] makes the most direct use of this idea by simply summing the penalty over all training samples and all candidate classes; SJE [3] only keeps the largest such penalty among all classes; ALE [2] adds weights to put more emphasis on the top candidate classes for each visual sample. Although these latter methods have led to promising results for ZSL, they fail to consider several key aspects of the problem. We argue this is due to several implicit assumptions that we identify and propose to fix by introducing corresponding novel contributions.

Assumption that classes are equally different. Relations between classes are typically ignored, *i.e.* there is no difference between any two incorrect class assignments in the triplet loss. However, many datasets encompass groups of very similar classes, particularly fine-grained datasets comprising many classes. One may argue that when building the similarity-based decision model, a confusion between two nearly indistinguishable classes should not be penalized as much as a confusion between two grossly different classes. Figure 1 illustrates such a case: we can see that two samples from classes 6 and 7 are much more difficult to tell apart than two samples from classes 6 and 2. A better differentiation of incorrect classes could enable to learn a more robust mapping between modalities. With this purpose we put forward a *flexible semantic margin* that takes into account the first and second order statistics of the class prototypes to introduce in the triplet loss a margin that reflects the actual dissimilarity between classes (Sec. 3.2).

Assumption of meaningful margin. In many triplet loss methods, the models are trained to separate seen classes with a fixed margin. Although not strictly necessary in the formulation of the objective—it should be sufficient for a visual sample to be at least slightly more compatible with its class prototype than with other prototypes—this margin is supposed to act as a regularizer and reduce overfitting on the training set. However, the compatibility between a visual sample and a prototype being often computed with a dot product, it is not bounded and can be made arbitrarily large by increasing the norm of the projected visual samples. Consequently, the constraint imposed by the margin is reduced to the point where it becomes negligible. While this may be desirable in some cases, by arbitrarily reducing regularization it also negatively affects the overall performance of the resulting model. We introduce a *partial normalization* that allows to learn a proper trade-off between the use of raw visual embeddings, which offers more flexibility to the model, and the unit normalized version, which forces the most restrictive use of the margin (Sec. 3.3).

Assumption of class homogeneity. All samples from seen classes are usually considered equally representative when building the model; yet, they may differ vastly within each class. In particular, some samples may not exhibit attributes usually shared by most members of their class. For example, although tigers are usually orange and striped, there exist white and albino tigers (and such examples can be found in the AwA2 dataset [19, 39]). Or, as illustrated in Fig. 1, a few images of a bird species may represent chicks (nestlings), whose appearance differs greatly from the adult specimens. The presence of outliers in the training set has a negative impact on the learned model, and this impact is stronger for similarity-based models like those employed for ZSL. To address this issue, we propose a *relevance weighting* scheme that quantifies the representativeness of each sample of the training classes (Sec. 3.4).

The core of our contribution is thus to take into account both inter-class and intra-class relations, respectively by being more permissive with confusions between similar classes, and by penalizing visual samples which are not representative of their class. Integrated in a simple triplet loss-based approach, our method also ensures that the constraint imposed by the margin is profitably enforced. Each contribution brings an advantage in itself as shown in the ablation study (Sec. 5). Through extensive evaluation, we show that the full proposal enables to reach a level of performance superior to the current state-of-the-art, in particular on fine-grained datasets with a large number of classes (Sec. 4). In addition, it does not require to change the underlying hypothesis of the ZSL task: the only data that needs to be available at training time consists of the visual samples of the *seen* classes and the corresponding class prototypes. We discuss limitations of our approach in Sec. 6 and draw some directions for further work.

2. Related work

Zero-Shot Learning. In addition to those mentioned in the introduction, many ZSL methods have been proposed [28, 25, 38, 33, 42]. Closely related to our work is the one of Annadani *et al.* [4] who consider how close classes are in the attribute space. This information is explicitly included in the objective function to learn a mapping from the attribute space to the visual space. Our approach to address this aspect is more general since it can be applied to a larger set of existing methods, including those based on the triplet loss. Similarly, Changpinyo *et al.* [7, 8] include an $\ell 2$ distance to measure class similarity in the structured loss variant of their synthesized classifiers (SYNC$_{\text{struct}}$). However, in addition to using higher order statistics to model class (dis)similarity and to adjust its mean and variance as hyperparameters, our overall approach is quite different as we learn a straightforward, linear mapping between modalities and do not make use of *phantom classes*. We also further incorporate these class dissimilarities in a broader framework, which enables our flexible margin to be fully leveraged during the training phase and addresses other limiting aspects of existing ZSL methods. Changpinyo *et al.* [9] proposed to learn to predict visual exemplars from attributes to use these predictions as additional semantic information. This method can be applied to most ZSL model and is thus complementary to our approach.

Generalized Zero-Shot Learning. In the early years of ZSL, only samples from unseen classes were included in the testing dataset. As emphasized by Chao *et al.* [10], it is more realistic to also include unseen samples from the seen classes since a user may want to recognize both unseen and seen classes. Known as Generalized Zero Shot Learning (GZSL), this setting usually leads to a strong bias towards recognizing *seen* classes, used to learn the model, thus reducing the performance of most existing ZSL approaches at the time. This was put into evidence by Xian *et al.* [41], who conducted an extensive evaluation of recent ZSL methods with a common protocol and reported synthetic results using the harmonic mean of the performances obtained on seen and unseen classes. To address this performance gap, Le Cacheux *et al.* [21] proposed a process to select some generic hyper-parameters of several ZSL methods that leads to a significant performance boost in a GZSL setting.

Generative methods. A recent line of research proposed to learn a conditional generator using the seen classes, then generate artificial training samples for the unseen classes [36, 6, 35, 40]. Discriminative models can then be trained based on real samples from the seen classes and artificial samples from the unseen classes. Verma *et al.* [36] proposed to model each class-conditional distribution as an exponential family. Bucher *et al.* tested different generative models of visual features and obtained the best results with a Generative Moment Matching Network [6]. In the same vein, Xian *et al.* [40] used a Generative Adversarial Network to synthesize CNN features conditioned on class-level semantic information. A slightly different approach was adopted by Verma *et al.* [35] who developed a model based on a conditional variational autoencoder and thus generate images on which features can be extracted to learn discriminative models for the unseen classes. These approaches are quite different from ours, since we do not generate features nor images of unseen classes. Once again, these contributions could nevertheless be combined to ours. However, we do not include here such a combination for two reasons. First, each method has a significant number of specific hyperparameters to set that make it quite sensitive to further processing. Second, the setting in which we evaluate our work is slightly less restrictive than the one implied by generative methods. Indeed, upon addition of even a single novel unseen class, generative methods must first generate artificial positive samples for this class, then train a discriminative classifier (SVM, softmax...) for this class and retrain all the classifiers for the previous (seen and unseen) classes. While a classical (G)ZSL system can immediately manage a new unseen class (and thus consider all of them incrementally), the generative approaches [6, 35, 40] need a fully-defined ZSL problem and have to (re)learn the discriminative models each time unseen classes are added. However, since these methods report some of the best results on the fine grained ZSL benchmarks, we compare our approach to them according to the same evaluation protocols.

Transductive setting. Transductive ZSL methods [13, 17, 30, 34] assume that unlabeled samples from unseen classes are available during training. This naturally leads to improved performance. We do not adopt such a restrictive hypothesis in this article and consider that *no* information regarding unseen classes is available at training time.

3. Proposed approach

3.1. Standard triplet loss

Let $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_N)^\top \in \mathbb{R}^{N \times D}$ represent N D-dimensional visual feature vectors, $\mathbf{y} = (y_1, \dots, y_N)^\top \in \{1, \dots, C\}^N$ the corresponding labels assigning them to one of C classes, and $\mathbf{S} = (\mathbf{s}_1, \dots, \mathbf{s}_C)^\top \in \mathbb{R}^{C \times K}$ the class prototypes. A ZSL training set consists of $\{\mathbf{X}, \mathbf{y}, \mathbf{S}\}$.

With a similarity function f providing a compatibility score $f(\mathbf{x}, \mathbf{s})$ between visual sample $\mathbf{x}$ and class prototype $\mathbf{s}$, the standard triplet loss aims to enforce the constraint that for any $\mathbf{x}_n$,

$$f(\mathbf{x}_n, \mathbf{s}_{y_n}) \geq f(\mathbf{x}_n, \mathbf{s}_c) + M, \forall c \neq y_n \tag{1}$$

where $\mathbf{s}_{y_n}$ is the corresponding class prototype, $\mathbf{s}_c$ a different class prototype and M a given margin.

The enforcement of this constraint for a triplet $(\mathbf{x}_n, \mathbf{s}_{y_n}, \mathbf{s}_c)$ takes the form of the following penalty:

$$[M + f(\mathbf{x}_n, \mathbf{s}_c) - f(\mathbf{x}_n, \mathbf{s}_{y_n})]_+ \tag{2}$$

where $[\cdot]_+$ denotes the function $\max\{0, \cdot\}$.

To use this triplet loss, the most straightforward approach is simply to sum it over all possible triplets in the training set as in [12]:

$$\frac{1}{N \cdot C} \sum_{n=1}^{N} \sum_{\substack{c=1 \\ c \neq y_n}}^{C} [M + f(\mathbf{x}_n, \mathbf{s}_c) - f(\mathbf{x}_n, \mathbf{s}_{y_n})]_+ \tag{3}$$

At testing time, the prediction $\hat{y}(\mathbf{x})$ for a visual sample $\mathbf{x}$ is the class among candidate classes $\mathcal{C}^{\text{test}}$ whose prototype maximizes the learned compatibility function f:

$$\hat{y}(\mathbf{x}) = \underset{c \in \mathcal{C}^{\text{test}}}{\operatorname{argmax}} f(\mathbf{x}, \mathbf{s}_c) \tag{4}$$

3.2. Flexible semantic margin

To take into account the distinction between similar and dissimilar classes, we replace the fixed margin M in Eq. (2) by a function $M(c, c')$ measuring the dissimilarity between classes c and c'. Since ZSL attributes tend to be correlated [15], such a function should take these correlations

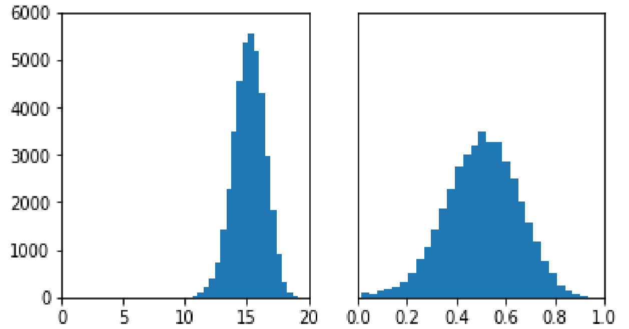

Figure 2. *Left*: histogram of the raw semantic distances in $\mathbf{D}$ from CUB. *Right*: histogram of the rescaled semantic distances in $\widetilde{\mathbf{D}}$ with $\mu_{\widetilde{\mathbf{D}}} = 0.5$ and $\sigma_{\widetilde{\mathbf{D}}} = 0.15$.

into account (as evidenced in Sec. 5). From the set of prototypes of the seen classes we first compute $\mathbf{\Sigma}^{-1}$, the inverse of the covariance matrix of the attributes. Because the number of prototypes is usually small compared to the dimension of the covariance matrix, we use the Ledoit-Wolf method [22] to obtain a more robust estimation of $\mathbf{\Sigma}^{-1}$. This is crucial, as a naive computation of $\mathbf{\Sigma}^{-1}$ would lead to poor results in the following. We then compute the matrix $\mathbf{D}$ composed of the Mahalanobis distances between all pairs of seen class prototypes $(\mathbf{s}_i, \mathbf{s}_j)$:

$$\mathbf{D}_{i,j} = [(\mathbf{s}_i - \mathbf{s}_j)^\top \mathbf{\Sigma}^{-1} (\mathbf{s}_i - \mathbf{s}_j)]^{\frac{1}{2}} \tag{5}$$

Since the semantic space is high dimensional, the distances composing $\mathbf{D}$ typically have low variance $\sigma^2_{\mathbf{D}}$. This is not desirable, since semantic distances that are too close to an average value do not support the intended goal of the introduction of a variable margin. Furthermore, as illustrated in Fig. 2, the initial mean value $\mu_{\mathbf{D}}$ of distances in $\mathbf{D}$ is arbitrary and can be quite large. We therefore rescale the elements of $\mathbf{D}$ to approximately have given mean $\mu_{\widetilde{\mathbf{D}}}$ and standard deviation $\sigma_{\widetilde{\mathbf{D}}}$ while keeping their values positive:

$$\widetilde{\mathbf{D}}_{i,j} = \left[\frac{\mathbf{D}_{i,j} - \mu_{\mathbf{D}}}{\sigma_{\mathbf{D}}} \sigma_{\widetilde{\mathbf{D}}} + \mu_{\widetilde{\mathbf{D}}} \right]_+ \tag{6}$$

The values we use for our flexible semantic margin are $M(c, c') = \widetilde{\mathbf{D}}_{c,c'}$. $\mu_{\widetilde{\mathbf{D}}}$ and $\sigma_{\widetilde{\mathbf{D}}}$ are considered to be hyper-parameters of the model. Note that setting $\sigma_{\widetilde{\mathbf{D}}} = 0$ is equivalent to using a fixed margin $M = \mu_{\widetilde{\mathbf{D}}}$.

3.3. Partial normalization

In Eq. (2), we intuitively expect that a larger value of M should constrain the model to increase the difference $f(\mathbf{x}_n, \mathbf{s}_c) - f(\mathbf{x}_n, \mathbf{s}_{y_n})$, and thus to better differentiate classes c and y_n. However, in most triplet loss methods, the compatibility function f is a dot product between the respective projections of the visual features $\boldsymbol{\theta}(\mathbf{x})$ and class prototypes $\boldsymbol{\phi}(\mathbf{s})$:

$$f(\mathbf{x}, \mathbf{s}) = \boldsymbol{\theta}(\mathbf{x})^\top \boldsymbol{\phi}(\mathbf{s}) \tag{7}$$

with $\boldsymbol{\theta}$ being a linear transformation $\boldsymbol{\theta}(\mathbf{x}) = \mathbf{W}\mathbf{x}$ and $\boldsymbol{\phi}$ the identity in [2, 3, 12]. In this latter case, since $\mathbf{s}$ is usually unit-normalized, so is $\boldsymbol{\phi}(\mathbf{s})$; the value of $f(\mathbf{x}, \mathbf{s})$ only depends—proportionally—on $\|\boldsymbol{\theta}(\mathbf{x})\|_2$ and $\cos(\alpha)$, where α is the angle between $\boldsymbol{\theta}(\mathbf{x})$ and $\boldsymbol{\phi}(\mathbf{s})$. While $\cos(\alpha)$ is obviously bounded, this is not usually the case for $\|\boldsymbol{\theta}(\mathbf{x})\|_2$.

The fact that $\|\boldsymbol{\theta}(\mathbf{x})\|_2$ can grow arbitrarily large means that an increased difference $f(\mathbf{x}_n, \mathbf{s}_c) - f(\mathbf{x}_n, \mathbf{s}_{y_n})$ can be achieved simply by scaling the similarities $f(\mathbf{x}, \mathbf{s})$ accordingly through $\|\boldsymbol{\theta}(\mathbf{x})\|_2$, leading to no practical gain. In practice, we observed that $\|\boldsymbol{\theta}(\mathbf{x})\|_2$ does indeed increase with M: the blue line (corresponding to $\gamma = 0$ as explained below) in Fig. 3 shows how $\|\boldsymbol{\theta}(\mathbf{x})\|_2$ grows from 1.2 with $M = 0.2$ to 6.6 with $M = 2$ on the CUB [37] dataset. This makes the value of M of little relevance and thus reduces the regularization provided by the margin.

We observed that simply regularizing $\boldsymbol{\theta}$ is not generally effective in preventing this effect: we found that, depending on the weight of this regularization, it is either unable to prevent a large $\|\boldsymbol{\theta}(\mathbf{x})\|_2$, or too restrictive and thus limiting the learning ability of the model; no trade-off between these two objectives was able to offer a satisfactory compromise. Fully normalizing $\boldsymbol{\theta}(\mathbf{x})$ before computing $f(\mathbf{x}, \mathbf{s})$ is not always optimal either: we found that completely removing the constraint to produce projections $\boldsymbol{\theta}(\mathbf{x})$ with consistent norms led to severe overfitting in some cases.

We therefore introduce a *partial normalization* function $\mathbf{\Psi}$ parameterized by a scalar $\gamma \in [0, 1]$ and applicable to any vector $\mathbf{v}$:

$$\mathbf{\Psi}_\gamma(\mathbf{v}) = \frac{1}{\gamma(\|\mathbf{v}\|_2 - 1) + 1} \cdot \mathbf{v} \tag{8}$$

defined in such a way that $\gamma = 0$ means that no transformation is applied—the initial norm of $\mathbf{v}$ is preserved—and

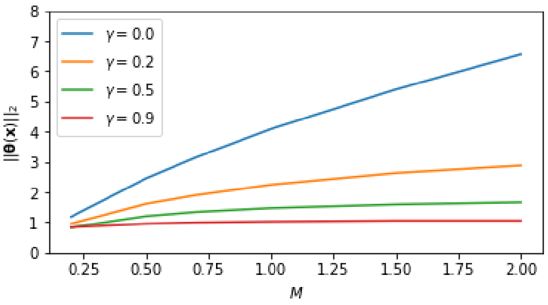

Figure 3. Average norm of projected visual features $\|\boldsymbol{\theta}(\mathbf{x})\|_2$ with respect to M on CUB, without ($\gamma = 0$) and with ($\gamma > 0$) partial normalization. Partial normalization helps prevent $\|\boldsymbol{\theta}(\mathbf{x})\|_2$ from growing with M.

$\gamma = 1$ means that $\Psi_\gamma(\mathbf{v})$ has unit norm.

This partial normalization is applied to the initial $\boldsymbol{\theta}(\mathbf{x})$. As the norm of $\boldsymbol{\theta}(\mathbf{x})$ can still be increased to compensate for Ψ_γ (provided $\gamma \neq 1$), it needs to be combined with a regularization on $\boldsymbol{\theta}$. The combination of both these elements helps preventing $\|\boldsymbol{\theta}(\mathbf{x})\|_2$ from growing arbitrarily, while still offering enough flexibility to avoid restricting the learning abilities of the model. It thus enables the margin M to achieve its intended goal, which is all the more important in our case as M also embodies the class similarities.

Fig 3 shows how partial normalization helps prevent $\|\boldsymbol{\theta}(\mathbf{x})\|_2$ from growing with M.

3.4. Relevance weighting

We make use of intra-class relations by explicitly taking into account the fact that some samples may not be representative of their class: we therefore assign a weight v_n to each training sample $\mathbf{x}_n$ to quantify its representativeness.

For each class c, let $\mathbf{X}^c = (\mathbf{x}_1^c, \dots, \mathbf{x}_{N_c}^c)^\top$ be the matrix whose N_c rows correspond to the visual samples from c, and $\mathbf{y}^c$ the associated class labels. We compute the vector $\mathbf{u}^c$ of distances between all $\mathbf{x}_n^c$ and the center of the class:

$$u_n^c = \left\| \mathbf{x}_n^c - \frac{1}{N_c} \sum_i \mathbf{x}_i^c \right\|_2 \tag{9}$$

Provided the visual features employed are appropriate for these distances to be meaningful, this gives us a first indication of how different an image is from the other images in the same class. It may also be possible to use higher order statistics to evaluate the intra-class relations; but since the visual space is typically very high-dimensional with few samples per class, they are impractical to robustly estimate, and such an approach did not lead to measurable gains.

To allow distances computed in Eq. (9) to result in weights on the same scale regardless of the initial within-class variance, we apply a cumulative distribution function on each element of $\mathbf{u}^c$ to obtain the weights $\mathbf{v}^c$:

$$v_n^c = 1 - \Phi\left(\frac{u_n^c - \mu_c}{\sigma_c}\right) \tag{10}$$

where μ_c and σ_c are the respective mean and standard deviation of the distances in $\mathbf{u}^c$, and $\Phi(\cdot)$ is the cumulative distribution function for the standard normal distribution.

For each visual sample $\mathbf{x}_n$, we weight its contribution to the loss with the corresponding v_n, so that representative samples have more importance.

3.5. Final model

The partially normalized projection of $\mathbf{x}_n$ is denoted by $\widehat{\mathbf{x}_n} = \boldsymbol{\Psi}_\gamma(\boldsymbol{\theta}(\mathbf{x}_n))$, and similarly $\widehat{\mathbf{s}_c} = \boldsymbol{\Psi}_\gamma(\boldsymbol{\phi}(\mathbf{s}_c))$. We choose to always fully normalize the projection of $\mathbf{s}_c$, such that $\widehat{\mathbf{s}_c} = \boldsymbol{\Psi}_1(\boldsymbol{\phi}(\mathbf{s}_c)) = \boldsymbol{\phi}(\mathbf{s}_c)/\|\boldsymbol{\phi}(\mathbf{s}_c)\|_2$.

The projections $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ are regularized by $\Omega[\boldsymbol{\theta}, \boldsymbol{\phi}]$. In order to enforce sparsity and reduce the number of hyper-parameters, Ω is defined as the sum of the average $\ell 1$ norms of the parameters $p_1, \dots, p_P$ of $\boldsymbol{\theta}$ and $q_1, \dots, q_Q$ of $\boldsymbol{\phi}$:

$$\Omega[\boldsymbol{\theta}, \boldsymbol{\phi}] = \frac{1}{P} \sum_{i=1}^{P} |p_i| + \frac{1}{Q} \sum_{i=1}^{Q} |q_i| \tag{11}$$

For a triplet $(\mathbf{x}_n, \mathbf{s}_{y_n}, \mathbf{s}_c)$, $c \neq y_n$, the triplet loss now takes the form:

$$l(\mathbf{x}_n, \mathbf{s}_{y_n}, \mathbf{s}_c) = [M(y_n, c) + \widehat{\mathbf{x}_n}^\top \widehat{\mathbf{s}_c} - \widehat{\mathbf{x}_n}^\top \widehat{\mathbf{s}_{y_n}}]_+ \tag{12}$$

This loss is summed over all triplets $(\mathbf{x}_n, \mathbf{s}_{y_n}, \mathbf{s}_c)$, each triplet being weighted by v_n, the representativeness of $\mathbf{x}_n$. The resulting final loss function is

$$\frac{1}{N \cdot C} \sum_{n=1}^{N} \left(v_n \sum_{\substack{c=1 \\ c \neq y_n}}^{C} l(\mathbf{x}_n, \mathbf{s}_{y_n}, \mathbf{s}_c) \right) + \lambda \Omega[\boldsymbol{\theta}, \boldsymbol{\phi}] \tag{13}$$

where λ is a regularization hyper-parameter.

Two settings of the model can be considered. The first consists in a linear projection of the visual features $\mathbf{x}$ onto the semantic space similar to [12], such that $\boldsymbol{\theta}(\mathbf{x}) = \mathbf{W}_{\boldsymbol{\theta}} \cdot \mathbf{x}$ and $\boldsymbol{\phi}$ is the identity. In the second setting, we make a linear projection of both $\mathbf{x}$ and $\mathbf{s}$ onto a common space with the same dimension as the semantic space, such that $\boldsymbol{\theta}(\mathbf{x}) = \mathbf{W}_{\boldsymbol{\theta}} \cdot \mathbf{x}$ and $\boldsymbol{\phi}(\mathbf{s}) = \mathbf{W}_{\boldsymbol{\phi}} \cdot \mathbf{s}$. Whether both $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ or only $\boldsymbol{\phi}$ is learned is considered to be a hyper-parameter.

The demo code for the final model is available at https://github.com/yannick-lc/iccv2019-triplet-loss.

4. Evaluation

4.1. Experimental setting

Datasets. We evaluate our approach on three standard ZSL datasets: Caltech UCSD Birds 200-2011 (**CUB**) [37], SUN Attribute (**SUN**) [27] and Animals with Attributes 2 (**AwA2**) [39]. CUB and SUN are fine-grained datasets consisting of 200 bird species for the former and 717 scenes for the latter. AwA2 is a more coarse-grained dataset of 50 animal species, that replaces the Animals with Attributes (AwA) [19] dataset whose images are no longer available.

Splits. ZSL test splits which include classes used to train a visual feature extractor, typically classes present in ImageNet [11], can induce a huge bias in the evaluation of the method. We therefore use the "Proposed Splits" of [41] to divide our datasets into training and testing splits and, as suggested in [41], we randomly remove 20% of the images from seen classes to be used as unseen samples from seen classes during testing in the GZSL setting.

We select the hyper-parameters using 3-fold cross-validation on the training sets for CUB and SUN. For AwA2 all 10 unseen testing classes should not be in ImageNet. But among the 40 training classes, only 8 are not in ImageNet, so randomly selected cross-validation folds would contain few such classes. This may introduce significant differences between hyper-parameter values that are optimal for cross-validation folds (as they would mostly contain ImageNet classes) and those optimal for truly unseen classes. We therefore decided to use a single validation split containing all 8 classes that are not in ImageNet.

Visual features. We employ a pre-trained ResNet-101 [14] network as deep visual feature extractor in order to have results comparable with the rest of the state-of-the-art, and in particular with [41] and [21]. Keeping the activation weights of the last pooling layer gives us a 2048-dimensional visual feature representation. Because we need a robust representation to compute distances between visual samples, we apply 10-crop on the original images, *i.e.* each 256 × 256 image is cropped into ten 224 × 224 images: one in each corner and one in the center for both the original image and its y-axis symmetry. The visual features of the resulting images are then averaged to obtain a 2048-dimensional vector. The visual vectors are finally normalized so that each visual feature has unit $\ell 2$ norm.

Attributes. We employ the standard attributes provided with each ZSL dataset, which we normalize to obtain class prototypes having unit $\ell 2$ norm.

Large scale setting. We also conduct experiments on the large scale **ImageNet** [11] dataset. We use the same splits (1K classes for training, up to 20K for testing), attributes (Word2Vec [24] trained on Wikipedia) and features (2048-dimensional vectors extracted with ResNet-101) as in [41].

Optimization. We train the models for 50 epochs using the ADAM optimizer [16], with parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$ and a learning rate of 0.001.

Metrics and evaluation setting. In order to be comparable with other recent publications, we employ the same metric as [41] for ZSL, per class accuracy, which is defined as ${}^{100}/_{|\mathcal{C}^{\text{test}}|} \cdot \sum_{c \in \mathcal{C}^{\text{test}}} \left({}^{1}/_{N_c} \sum_{m=1}^{N^c} \mathbb{1}[\hat{y}(\mathbf{x}_m^c) = y_n^c] \right)$ where $\mathbb{1}[\cdot]$ is the indicator function.

For GZSL, we also use the metrics from [41]. $A_{\mathcal{U} \to \mathcal{U}+\mathcal{S}}$ denotes the accuracy on samples from unseen classes when candidate classes consist of all classes, seen and unseen. $A_{\mathcal{S} \to \mathcal{U}+\mathcal{S}}$ is similarly defined as the accuracy on test (held-out) samples from seen classes. The final GZSL score is the harmonic mean H of $A_{\mathcal{U} \to \mathcal{U}+\mathcal{S}}$ and $A_{\mathcal{S} \to \mathcal{U}+\mathcal{S}}$: $H = 2 \cdot A_{\mathcal{U} \to \mathcal{U}+\mathcal{S}} \cdot A_{\mathcal{S} \to \mathcal{U}+\mathcal{S}} / (A_{\mathcal{U} \to \mathcal{U}+\mathcal{S}} + A_{\mathcal{S} \to \mathcal{U}+\mathcal{S}})$.

We report results averaged over 10 runs with different random initializations of our model parameters.

Method	**CUB**	**SUN**	**AwA2**	Avg
ALE* [2]	54.9	58.1	62.5	58.5
DEVISE* [12]	52.0	56.5	59.7	56.0
SJE* [3]	53.9	53.9	61.9	56.6
ESZSL* [31]	53.9	54.5	58.6	55.6
SYNC$_{\text{o-vs-o}}$* [7]	55.6	56.3	46.6	52.8
PSR [4]	56.0	61.4	63.8	60.4
Ours	*63.8*	*63.5*	67.9	**65.1**
Generative models†				
GFZSL*† [36]	49.3	60.6	63.8	57.9
SE-ZSL† [35]	59.6	63.4	69.2	64.0
Xian† [40]	57.3	60.8	68.2	62.1
Bucher† [6]	59.4	60.1	*69.9*	63.1

Table 1. Accuracy for ZSL, on the "proposed split" of [41]. Results reported in [39] are marked with *. The generative models, marked with †, rely on stronger hypotheses as explained in Sec. 2. Results of Bucher *et al.* [6] are those with the GMMN generative model; those of Xian *et al.* [40] are with Softmax+F-CLSWGAN. Our results are averaged over 10 runs.

Hierarchy	**2-hop**	**3-hop**	**All**
CONSE* [25]	7.63	2.18	0.95
ESZSL* [31]	6.35	1.51	0.62
SYNC$_{\text{o-vs-o}}$* [7]	9.26	2.29	0.96
Ours	**9.81**	**2.52**	**1.09**

Table 2. ZSL results on the large scale ImageNet dataset. Results marked with * correspond to top 3 results from [39].

4.2. Selection of hyper-parameters

We employ the following protocol to determine the hyper-parameters using the validation splits: for a given setting (whether both $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ or only $\boldsymbol{\theta}$ are learned, Sec. 3.5), we first set $\sigma_{\tilde{\mathbf{D}}}$ and γ to 0, and $\mu_{\tilde{\mathbf{D}}}$ to 1, so that the setting is approximately the one from DEVISE. We determine the best λ using the validation set(s). We then divide this value by a factor of 10 in order to not over-constrain the model, and use this new value when jointly selecting γ and $\mu_{\tilde{\mathbf{D}}}$. We select $\sigma_{\tilde{\mathbf{D}}}$ while keeping the other hyper-parameters fixed. Finally, we explore values in the neighborhood of the selected quadruplet $(\mu_{\tilde{\mathbf{D}}}, \sigma_{\tilde{\mathbf{D}}}, \gamma, \lambda)$. We retain the setting of the model producing the best results on the validation set.

For CUB and SUN, the 3-fold cross-validation works as usual. For AwA2 and ImageNet, since there is only one validation set, we perform each evaluation of a set of hyper-parameters with 3 different initializations in order to improve the robustness of the estimation.

4.3. Standard Zero-Shot Learning results

Table 1 reports results in a standard ZSL setting, where test samples belong to unseen classes and candidate classes consist of unseen classes only ($A_{\mathcal{U} \to \mathcal{U}}$). We compare our

Method	**CUB**			**SUN**			**AwA2**			$\bar{H}$
	$A_{\mathcal{U}\to\mathcal{U}+\mathcal{S}}$	$A_{\mathcal{S}\to\mathcal{U}+\mathcal{S}}$	H	$A_{\mathcal{U}\to\mathcal{U}+\mathcal{S}}$	$A_{\mathcal{S}\to\mathcal{U}+\mathcal{S}}$	H	$A_{\mathcal{U}\to\mathcal{U}+\mathcal{S}}$	$A_{\mathcal{S}\to\mathcal{U}+\mathcal{S}}$	H	
Non generative approaches, without calibration										
ALE* [2]	23.7	62.8	34.4	21.8	33.1	26.3	14.0	81.8	23.9	28.2
DeViSE* [12]	23.8	53.0	32.8	16.9	27.4	20.9	17.1	74.7	27.8	27.2
SJE* [3]	23.5	59.2	33.6	14.7	30.5	19.8	8.0	73.9	14.4	22.6
ESZSL* [31]	12.6	63.8	21.0	11.0	27.9	15.8	5.9	77.8	11.0	15.9
SynC$_{\text{o-vs-o}}$* [7]	11.5	70.9	19.8	7.9	43.3	13.4	10.0	90.5	18.0	17.0
PSR [4]	24.6	54.3	33.9	20.8	37.2	26.7	20.7	73.8	32.3	31.0
Ours	30.4	65.8	41.2	22.0	34.1	26.7	17.6	87.0	28.9	**32.3**
Non generative approaches, with calibration										
ALE** [2]	-	-	49.2	-	-	34.9	-	-	56.9	47.0
DeViSE** [12]	-	-	42.4	-	-	32.5	-	-	55.0	43.3
SJE** [3]	-	-	46.7	-	-	36.8	-	-	59.4	47.6
ESZSL** [31]	-	-	38.7	-	-	11.8	-	-	54.4	35.0
SynC$_{\text{struct}}$** [7]	-	-	48.9	-	-	27.9	-	-	62.6	46.5
Ours	55.8	52.3	*53.0*	47.9	30.4	36.8	48.5	83.2	61.3	**50.4**
Generative approaches†										
GFZSL*† [36]	0.0	45.7	0.0	0.0	39.6	0.0	2.5	80.1	4.8	1.6
SE-GZSL† [35]	41.5	53.3	46.7	40.9	30.5	34.9	58.3	68.1	*62.8*	48.1
Xian† [40]	43.7	57.7	49.7	42.6	36.6	*39.4*	57.9	61.4	59.6	**49.6**
Bucher† [6]	49.1	55.9	52.3	39.7	37.7	38.7	46.3	77.3	57.3	49.4

Table 3. Accuracy for GZSL on the “proposed split” of [41]. $A_{\mathcal{U}\to\mathcal{U}}$ and $A_{\mathcal{S}\to\mathcal{U}}$ are the top-1 per class accuracy on *unseen* and *seen* classes respectively. H is the harmonic mean and $\bar{H}$ the average over the three datasets. Results reported in [39] are marked with * and those in [21] with **, the others are reported in the given papers. The generative models, marked with †, rely on stronger hypotheses as explained in Sec. 2. Results of Bucher *et al.* [6] are those with the GMMN generative model; those of Xian *et al.* [40] are those with Softmax+F-CLSWGAN. Our results are averaged over 10 runs.

results to those of several state-of-the-art approaches. We only report the best and most relevant results from [39] to avoid overloading the table. We also included recent state-of-the-art ZSL methods based on generative models in our comparison, although they rely on stronger hypotheses than our approach, as explained in Sec. 2. For this reason, we clearly distinguish generative from non generative methods. We chose to not report any transductive ZSL results as they rely on significantly different and even stronger hypotheses.

Our approach outperforms all the non-generative methods on the three datasets. Surprisingly, it also outperforms the generative approaches on two datasets, as well as by more than 1 point for the average. Three of the generative models obtain better scores than ours on AwA2 only. Indeed, this last dataset is not fine-grained with a large number of classes, so the proposed approach is less relevant.

We also evaluate our approach on the large-scale ImageNet dataset (Table 2), where it likewise outperforms the rest of the state-of-the-art reported in [39].

4.4. Generalized Zero-Shot Learning results

Standard Generalized Zero-Shot Learning. Most recent ZSL methods are also evaluated on GZSL. In the standard evaluation setting, $A_{\mathcal{U}\to\mathcal{U}+\mathcal{S}}$ and $A_{\mathcal{S}\to\mathcal{U}+\mathcal{S}}$ are directly measured, and the final GZSL score is their harmonic mean H. We report our results in this setting, as well as other state-of-the art results in the section *Non generative approaches, without calibration* of Table 3. As already mentioned, there is usually a strong imbalance in favor of the seen classes, penalizing the final score. As a result, the best GZSL models are typically those with the best $A_{\mathcal{U}\to\mathcal{U}+\mathcal{S}}$.

Calibrated Generalized Zero-Shot Learning. As argued in [21], it is more relevant to re-balance or *calibrate* $A_{\mathcal{U}\to\mathcal{U}+\mathcal{S}}$ and $A_{\mathcal{S}\to\mathcal{U}+\mathcal{S}}$ by slightly penalizing seen classes for the benefit of unseen classes during the prediction step, as this more accurately reflects what is expected in a real use case. We share this view and therefore report results after calibration in the *Non generative approaches, with calibration* section of Table 3. The optimal calibration was obtained using training / validation / testing splits specific to GZSL following the protocol described in [21], and we report the score of all models whose calibrated harmonic mean H is given there ([21] does not provide the associated $A_{\mathcal{U}\to\mathcal{U}+\mathcal{S}}$ and $A_{\mathcal{S}\to\mathcal{U}+\mathcal{S}}$).

Generative models. Since generative models typically do not suffer from the imbalance between $A_{\mathcal{U}\to\mathcal{U}+\mathcal{S}}$, $A_{\mathcal{S}\to\mathcal{U}+\mathcal{S}}$ (except GFZSL which overfits on seen classes), we simply report their results as such, without calibration.

Analysis. On average, our approach outperforms other non generative approaches in the standard GZSL setting, *i.e.* without calibration. It also obtains the best score on CUB and SUN. Unsurprisingly, GZSL scores are much higher when calibration is employed; although this is the case for most methods, our approach still obtains the best average score in this setting. The final score on CUB is even higher than the best reported score for a generative approach, even though our approach is arguably simpler and less restrictive as far as the underlying task hypotheses are concerned.

5. Ablation study

In order to evaluate the impact of each component of our proposed approach, we perform an ablation study. We successively deactivate the flexible semantic margin (FSM) by setting $\sigma_{\tilde{\mathbf{D}}}$ to 0 in Eq. (6); the partial normalization (PN) by setting γ to 0 in Eq. (8); and the relevance weighting (RW) scheme by setting all the weights v_n to 1 in Eq. (13). Active hyper-parameters are re-evaluated on the validation set accordingly. We also evaluate the impact of taking attribute correlations into account to estimate class similarities by setting $\mathbf{\Sigma}^{-1}$ to the identity matrix $\mathbf{I}_K$ in Eq. (5), so that semantic distances correspond to euclidean distances.

Table 4 shows the results on the CUB dataset. Even though for CUB the best performing setting of the model is when both $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ (Sec. 3.5) are learned, as determined on the validation sets, we also include some results for the other setting, *i.e.* when only $\boldsymbol{\theta}$ is learned. We denote the respective settings by $\boldsymbol{\theta} + \boldsymbol{\phi}$ and $\boldsymbol{\theta}$. Partial normalization has the largest impact in both cases, but a flexible semantic margin and relevance weighting also significantly increase the final score. They work particularly well together, as their combined impact is better than the sum of their marginal impacts. The bottom line in Table 4 corresponds to a method close to DeViSE [12], and has comparable results.

6. Discussion

The relevance weighting we employ relates the impact of each training sample to its representativeness of the class. A typical case was illustrated here by chicks that are visually different from adult birds and are scarcely present in the bird classes. By reducing the importance of samples representing chicks of a given species, one improves the ability of the system to recognize the adults of this species. However, this also makes the chicks of the species harder to recognize. Chicks of some species are treated as outliers because they are atypical and significantly under-represented, *not* because they don't belong to that species. Beyond the case of bird or animal species that dominate in the ZSL benchmarks of the literature, this may be an issue for some practical cases. A way to circumvent the problem is to define sub-classes and process each of them separately, provided that each is sufficiently well represented in the training set. On the *seen* classes, sub-classes can be found by clustering the visual feature vectors. The problem remains for the *unseen* classes, unless a transductive setting is adopted.

Setting	FSM	(MD)	PN	RW	**Score**
$\boldsymbol{\theta} + \boldsymbol{\phi}$	✓		✓	✓	**63.8**
	✓	-	✓	✓	61.7
	-		✓	✓	61.8
	✓		-	✓	57.6
	✓		✓	-	61.7
	-		-	✓	61.0
	-		-	-	56.6
$\boldsymbol{\theta}$	✓		✓	✓	**61.3**
	-		✓	✓	61.1
	✓		-	✓	60.0
	✓		✓	-	55.3
	-		-	-	55.0

Table 4. Ablation study on the CUB dataset, with settings $\boldsymbol{\theta} + \boldsymbol{\phi}$ and $\boldsymbol{\theta}$ (Sec. 3.5). FSM stands for the *flexible semantic margin*, Eq. (6). MD stands for the Mahalanobis distance, Eq. (5). PN stands for *partial normalization*, Eq. (8). RW stands for *relevance weighting*, Eq. (10). Results averaged over 10 runs.

The performance exhibited by our approach is comparable to that of generative methods, while we rely on less restrictive hypotheses than generative methods—since they need to have information about all unseen classes to generate the samples required to learn the discriminative models. An alternative with the generative approach is to use incremental learning systems [29, 5] but it usually leads to a significant drop in performance. Hence, our proposal has a practical interest for real systems that aim to recognize unseen classes whose number increases regularly.

7. Conclusion

Before the introduction of the generative approach for ZSL, most of the best performing methods for ZSL were based on the triplet loss. Several assumptions made by these methods limited their ability to reach optimal performance on real use cases, particularly on fine-grained datasets comprising a large number of classes. In this paper, we identified three of these assumptions and put forward novel contributions to solve the problems they raised. We thus proposed to model in a triplet loss approach the intra and inter-class relations of ZSL datasets, by accounting for the actual dissimilarities of the classes and quantifying the representativeness of each training sample with regard to its label, while ensuring that the constraint imposed by the margin is profitably enforced. This approach significantly improves performance with respect to non generative methods and even outperforms generative methods, while making less restrictive hypotheses.

References

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

[2] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *Pattern Analysis and Machine Intelligence*, 38(7):1425–1438, 2016. 1, 4, 6, 7

[3] Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *Computer Vision and Pattern Recognition*, pages 2927–2936. IEEE, 2015. 1, 4, 6, 7

[4] Yashas Annadani and Soma Biswas. Preserving semantic relations for zero-shot learning. In *Computer Vision and Pattern Recognition*, pages 7603–7612, 2018. 2, 6, 7

[5] Eden Belouadah and Adrian Popescu. Deesil: Deep-shallow incremental learning. In *Computer Vision – ECCV 2018 Workshops*, pages 151–157, 01 2018. 8

[6] Maxime Bucher, Stéphane Herbin, and Frédéric Jurie. Zero-shot classification by generating artificial visual features. In *RFIAP*, 2018. 3, 6, 7

[7] Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Synthesized classifiers for zero-shot learning. In *Computer Vision and Pattern Recognition*, pages 5327–5336. IEEE, 2016. 2, 6, 7

[8] Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Classifier and exemplar synthesis for zero-shot learning. *arXiv preprint arXiv:1812.06423*, 2018. 2

[9] Soravit Changpinyo, Wei-Lun Chao, and Fei Sha. Predicting visual exemplars of unseen classes for zero-shot learning. In *Computer Vision and Pattern Recognition*, pages 3476–3485, 2017. 2

[10] Wei-Lun Chao, Soravit Changpinyo, Boqing Gong, and Fei Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In *European Conference on Computer Vision*, pages 52–68. Springer, 2016. 3

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition*, pages 248–255. IEEE, 2009. 5, 6

[12] Andrea Frome, Greg S. Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Tomas Mikolov, et al. Devise: A deep visual-semantic embedding model. In *Advances in Neural Information Processing Systems*, pages 2121–2129, 2013. 1, 3, 4, 5, 6, 7, 8

[13] Yanwei Fu, Timothy M. Hospedales, Tao Xiang, and Shaogang Gong. Transductive multi-view zero-shot learning. *Pattern Analysis and Machine Intelligence*, 37(11):2332–2345, 2015. 3

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition*, pages 770–778. IEEE, 2016. 6

[15] Dinesh Jayaraman, Fei Sha, and Kristen Grauman. Decorrelating semantic visual attributes by resisting the urge to share. In *Computer Vision and Pattern Recognition*, pages 1629–1636, 2014. 3

[16] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, 2014. 6

[17] Elyor Kodirov, Tao Xiang, Zhenyong Fu, and Shaogang Gong. Unsupervised domain adaptation for zero-shot learning. In *International Conference on Computer Vision*, pages 2452–2460, 2015. 3

[18] Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *Computer Vision and Pattern Recognition*, pages 951–958. IEEE, 2009. 1

[19] Christoph H. Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *Pattern Analysis and Machine Intelligence*, 36(3):453–465, 2014. 2, 5

[20] Hugo Larochelle, Dumitru Erhan, and Yoshua Bengio. Zero-data learning of new tasks. In *AAAI Conference on Artificial Intelligence*, 2008. 1

[21] Yannick Le Cacheux, Hervé Le Borgne, and Michel Crucianu. From classical to generalized zero-shot learning: A simple adaptation process. In *International Conference on Multimedia Modeling*, pages 465–477. Springer, 2019. 3, 6, 7

[22] Olivier Ledoit and Michael Wolf. A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88(2):365–411, 2004. 4

[23] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008. 1

[24] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013. 6

[25] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg Corrado, and Jeffrey Dean. Zero-shot learning by convex combination of semantic embeddings. In *International Conference on Learning Representations*, 2014. 2, 6

[26] Mark Palatucci, Dean Pomerleau, Geoffrey E. Hinton, and Tom M. Mitchell. Zero-shot learning with semantic output codes. In *Advances in Neural Information Processing Systems*, pages 1410–1418, 2009. 1

[27] Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *Computer Vision and Pattern Recognition*, pages 2751–2758. IEEE, 2012. 5

[28] Shafin Rahman, Salman Khan, and Fatih Porikli. A unified approach for conventional zero-shot, generalized zero-shot, and few-shot learning. *IEEE Transactions on Image Processing*, 27(11):5652–5667, 2018. 2

[29] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, and Christoph H. Lampert. icarl: Incremental classifier and representation learning. *Computer Vision and Pattern Recognition*, pages 5533–5542, 2017. 8

[30] Marcus Rohrbach, Sandra Ebert, and Bernt Schiele. Transfer learning in a transductive setting. In *Advances in Neural Information Processing Systems*, pages 46–54, 2013. 3

[31] Bernardino Romera-Paredes and Philip Torr. An embarrassingly simple approach to zero-shot learning. In *International Conference on Machine Learning*, pages 2152–2161, 2015. 1, 6, 7

[32] Yutaro Shigeto, Ikumi Suzuki, Kazuo Hara, Masashi Shimbo, and Yuji Matsumoto. Ridge regression, hubness, and zero-shot learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 135–151. Springer, 2015. 1

[33] Richard Socher, Milind Ganjoo, Christopher D. Manning, and Andrew Ng. Zero-shot learning through cross-modal transfer. In *Advances in Neural Information Processing Systems*, pages 935–943, 2013. 2

[34] Jie Song, Chengchao Shen, Yezhou Yang, Yang Liu, and Mingli Song. Transductive unbiased embedding for zero-shot learning. In *Computer Vision and Pattern Recognition*, pages 1024–1033, 2018. 3

[35] Vinay Kumar Verma, Gundeep Arora, Ashish Mishra, and Piyush Rai. Generalized zero-shot learning via synthesized examples. In *Computer Vision and Pattern Recognition*, 2018. 3, 6, 7

[36] Vinay Kumar Verma and Piyush Rai. A simple exponential family framework for zero-shot learning. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 792–808. Springer, 2017. 3, 6, 7

[37] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD Birds-200-2011 dataset. 2011. 1, 4, 5

[38] Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. Latent embeddings for zero-shot classification. In *Computer Vision and Pattern Recognition*, pages 69–77, 2016. 2

[39] Yongqin Xian, Christoph H. Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *Pattern Analysis and Machine Intelligence*, 2018. 2, 5, 6, 7

[40] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *Computer Vision and Pattern Recognition*, 2018. 3, 6, 7

[41] Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning – the good, the bad and the ugly. In *Computer Vision and Pattern Recognition*, pages 3077–3086. IEEE, 2017. 3, 5, 6, 7

[42] Ziming Zhang and Venkatesh Saligrama. Zero-shot learning via semantic similarity embedding. In *International Conference on Computer Vision*, pages 4166–4174, 2015. 2