Trading-off Information Modalities in Zero-shot Classification

Jorge Sánchez
CIEM-CONICET
Av. Medina Allende s/n,
X5000HUA Córdoba, Argentina
jorge.sanchez@unc.edu.ar

Matías Molina
CONICET and Universidad Nacional de Córdoba
Av. Medina Allende s/n,
X5000HUA Córdoba, Argentina
matias.molina@unc.edu.ar

Abstract

Zero-shot classification is the task of learning predictors for classes not seen during training. A practical way to deal with the lack of annotations for the target categories is to encode not only the inputs (images) but also the outputs (object classes) into a suitable representation space. We can use these representations to measure the degree at which images and categories agree by fitting a compatibility measure using the information available during training. One way to define such a measure is by a two step process in which we first project the elements of either space (visual or semantic) onto the other and then compute a similarity score in the target space. Although projections onto the visual space has shown better general performance, little attention has been paid to the degree at which the visual and semantic information contribute to the final predictions. In this paper, we build on this observation and propose two different formulations that allow us to explicitly trade-off the relative importance of the visual and semantic spaces for classification in a zero-shot setting. Our formulations are based on redefinition of the similarity scoring and loss function used to learn the projections. Experiments on six different datasets show that our approach lead to improve performance compared to similar methods. Moreover, combined with synthetic features, our approach competes favorably with the state of the art on both the standard and generalized settings.

1. Introduction

In the past decade we have witnessed a tremendous growth on the capabilities of learning systems to encode complex high-level information from data. Along with the development of novel algorithms and powerful computing machinery, a key factor for this growth has been the availability of increasingly large amounts of manually annotated data that can be used for training, with ImageNet [7] being the paradigmatic example on the field of image classification. Although gathering large quantities of data is in many cases possible, the effort required to annotate it may be too big to make it practical, specially in cases of uncommon and fine-grained visual concepts. This has lead to a great interest in the development of predictive models that can be trained from a few examples [30, 38, 26]. Zero-shot learning (ZSL) is a extreme case in which for some of the target concepts no training samples are provided [18, 40]. For instance, in the zero-shot classification (ZSC) problem we are given labeled samples from a known set of categories and we are asked to learn a model that is able to make predictions about object classes not seen during training, in which case we need to resort on additional sources of information that allow us to overcome the lack of annotations for such classes. Although different sources of side information has been explored in the past (word embeddings [10, 44], class hierarchies [1], textual descriptors [19], etc.), visual attributes [18] remain as the most effective output encoding method in the literature.

It is important to note that, differently from the fully supervised learning setting, in the zero-shot scenario all knowledge about the target categories is not encoded explicitly (via class-level labels) but indirectly trough a representation space that is different from the one chosen to encode the images. The interplay between these two "views" of the same abstract concepts need to be coordinated in order to be useful. However, different problems might require different trade-offs in terms of the information that these two view may provide, *i.e.* visual cues might be relevant in discriminating fine-grained details while semantic relations might help extrapolate to a different set of visually similar objects. Training a model under the zero-shot setting is not only about learning input-output relations, but uncovering semantic relations as seen by the representations chosen to encode both images and concepts. Here, instead of learning a partitioning of the input space based on class-membership relations, we can learn a scoring function that measures the degree at which image- and class-level representations "agree" on a given abstract concept. This has become the dominant approach to tackle the zero-shot classification in the literature. A particularly relevant family of methods correspond to those that seek (explicitly or not) to learn a projection for the representations in one space (*e.g.* visual) to the other (*e.g.* semantic) so that they can be easily compared by means of a suitable metric on the target space. In this case, learning the model accounts to optimizing both the projection and similarity computation under any constraint imposed by the end task.

In this work, we build on this idea and consider a simple model based on bilinear projections and cosine similarities on the visual and semantic spaces. Since our aim it to study the interplay between the visual and semantic modalities (as induced by the model) and not to learn better single-modality representations, we assume both representations are given and fixed during training. We propose two different formulations that allow us to parametrically trade-off the relative importance of each space in dealing with the zero-shot problem. One of the formulations is based on the redefinition of the scoring mechanism while the other on the definition of an upper bound over a suitable loss function. We run extensive experimental evaluations on six different datasets. Despite their simplicity, our models are shown to compete favorably with more elaborated formulations.

2. Related Work

Zero-shot recognition relies on the semantic knowledge encoded into the representations chosen for the outputs. Among different alternatives [10, 44, 1, 19], attribute descriptors remain the most effective [3, 40, 36]. Early works on attribute based classification divided the problem in two different stages: attribute prediction and class label assignment. The DAP and IAP methods of [18] are among the first methods in the literature based on this idea. For instance, DAP first learns a set of attribute predictors based on the available training information and then use these models to classify previously unseen categories according to a maximum-a-posteriori (MAP) rule. It has been observed [11] that two-stage approaches suffer from the domain-shift problem, which arises from the decoupling of the problem into the intermediate task of learning attribute predictors and the end task of predicting class labels. Instead, more recent work seek to mitigate this problem by directly learning a mapping between the visual feature space and the semantic space. Here, while some methods rely on simple bilinear forms and loss formulations [10, 3, 28, 17], others exploit more complex projections or elaborate on more complex learning objectives [33, 4, 43, 8, 20].

DeViSE [10] propose a loss inspired by the Ranking SVM [16] and learns a linear mapping from the visual to the semantic embedding space. SJE [3] and ALE [2] learn a bilinear compatibility by optimizing a ranking loss that ensures that more importance is given to the top of the list. While SJE optimizes a loss inspired from the Structure SVM [32], ALE uses on a weighted approximation to the ranking objective [39]. ESZSL [28] proposes a simple closed-form solution by considering the Euclidean error of the projections induced by the model from the visual to the semantic space and back. SAE [17] further explores this idea and propose an autoencoder learned so as to minimize the projection and reconstruction errors on the semantic and visual feature space. Using a *hubness* argument, *i.e.* the presence of universal neighbors or *hubs* in high-dimensional vector spaces, Zhang *et al.* [44] propose a deep embedding model that project attribute descriptors onto the visual representation space. Experimental results show the advantages of using the visual space as the embedding space for (nearest-neighbor) classification. A similar observation was made by Jiang *et al.* [14] and Wang *et al.* [37] regarding the usefulness of the visual space under a the zero-shot learning setting.

GFZSL [33] proposes a generative framework based on the estimation of class-conditional distributions from attribute descriptors. At test time, attributes from unseen classes are used to regress the parameters of these class-conditional densities. Given a test sample, it is assigned to the class with the maximum posterior probability. PSR [4] use the similarity between attributes as a proxy for the semantic similarity of the corresponding categories. The authors propose to learn a mapping from the semantic to the visual space such that these proximity relations are preserved. The model is based on encoder-decoder architectures and a triplet-based loss formulation. ZSKL [43] apply the concept of kernel alignment [6] and propose different non-linear compatibility functions between visual and semantic features. MLSE [8] encodes the semantic representations into a latent space through an adaptive graph formulation and a rather involved optimization procedure.

More recently, with the emergence of deep generative modeling [12, 22], some works propose to learn to synthetize samples from arbitrary classes based on the aligned samples available for training [41, 29]. These family of approaches deviate from the more restrictive ZSL setting as they assume the availability of semantic descriptors for the test classes during training. They correspond to the *class-transductive instance-inductive* characterization of [36]. Assuming that target class information is available during training has also been explored in non-generative settings. For instance, [15] propose a contrastive learning formulation, observing large improvements on the generalized ZSL setting.

For a broader overview of these and other methods we refer the reader to [40, 36].

3. Preliminaries

In zero-shot classification (ZSC) we are given a training set $\mathcal{D}^{tr} = \{(x_i, y_i)\}_{i=1}^{N}$ of image-label pairs, $x_i \in \mathcal{X}$, $y_i \in \mathcal{Y}^{tr} \subset \mathcal{Y}$, sampled from a known set of visual categories. The goal is to learn a mapping $f : \mathcal{X} \to \mathcal{Y}$ from $\mathcal{D}^{tr}$ that can be used to classify samples over a different set $\mathcal{Y}^{ts} \subset \mathcal{Y}$. If the sets $\mathcal{Y}^{tr}$ and $\mathcal{Y}^{ts}$ are disjoint, the problem is known as standard zero-shot classification; otherwise, it is known as generalized zero-shot classification. The particular case of $\mathcal{Y}^{tr} = \mathcal{Y}^{ts}$ corresponds to the standard supervised classification problem. To overcome the lack of annotations for the target categories, we assume each class $y \in \mathcal{Y}$ can be encoded by $z_y \in \mathcal{Z}$ in an output representation space that allow us to encode some of the abstract semantic relations that can be found among the elements of $\mathcal{Y}$. Examples of such representations are visual attributes [18], word embeddings [10, 44], class hierarchies [1] and textual descriptors [19]. Let $z_y = z(y) \in \mathcal{Z}$ denote the representation of class y. The problem can be reformulated as that of learning a mapping (scoring function) $F : \mathcal{X} \times \mathcal{Z} \to \mathbb{R}$ that measures the degree at which the input and output signatures *agree* on a concept (visual-semantic compatibility score) and then use this model to rank test images and target categories accordingly. Parameters of this model are learned solely using samples from $\mathcal{D}^{tr}$. Once the model is trained, it can be used to classify an input sample x over an arbitrary set $\tilde{\mathcal{Y}}$ as follows:

$$\hat{y} = \underset{y \in \tilde{\mathcal{Y}}}{\operatorname{argmax}} F(x, z_y; W), \tag{1}$$

where $\tilde{\mathcal{Y}} = \mathcal{Y}^{ts}$ or $\tilde{\mathcal{Y}} = \mathcal{Y}^{tr} \cup \mathcal{Y}^{ts}$ for the standard or generalized settings, respectively, and W denotes model parameters. If both the input and output spaces are encoded by means of vectorial representations, a common yet effective approach to define F is as a simple bilinear form, as follows:

$$F(x, z_y; W) = x^T W z_y \tag{2}$$

where $x \in \mathbb{R}^D$, $z_y \in \mathbb{R}^E$ and $W \in \mathbb{R}^{D \times E}$. For instance, x may correspond to a feature vector extracted from a pretrained deep network and z_y to a vector representation for class y. Using $\langle \cdot, \cdot \rangle$ to denote the dot-product operator, Eq. (2) can be written as:

$$F(x, z_y; W) = \langle x, W z_y \rangle = \langle W^T x, z_y \rangle. \tag{3}$$

This form is particularly attractive since it allow us to see the compatibility score induced by Eq. (2) as a two-step symmetric process in which we first project one of the representations onto the space of the other and then compute a similarity score on the target space.

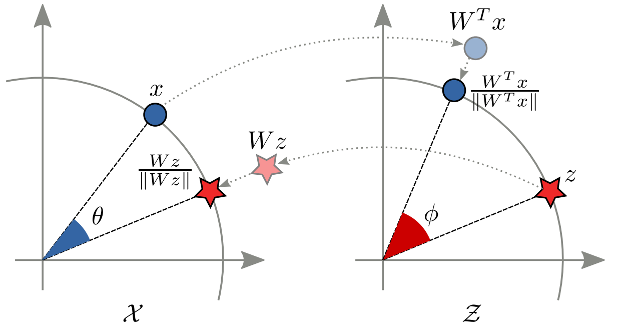

Figure 1. Suppose that $\|x\| = \|z\| = 1$. Replacing the dot-product by the cosine similarity in either side of Eq. (3) can be seen as linear embedding (by W^T or W) followed by a projection onto the unit sphere on the target space. The visual-semantic compatibility between x and y is the dot-product between one of the signatures and the projection of the other, *i.e.* $\cos\theta$ and $\cos\phi$.

4. Projection Symmetries in ZSC

From the discussion above, we see that the symmetry of Eq. (3) stems from the linearity of the projection by W and the use of a simple dot-product for similarity computation. Replacing either of them by a non-linear operation brakes this symmetry and makes the similarities defined on the visual and semantic spaces to behave differently. In what follows, we consider replacing the dot-product by a cosine similarity and propose two different formulations that allow us to trade off the relative importance of the semantic and visual modalities in a zero-shot scenario. Figure 1 provides an illustration of the projections and similarity notions induced by this process.

4.1. Weighted similarity formulation

We first explore a simple convex combination of the similarities in the visual and semantic domains, as follows:

$$F_\beta(x, z_y; W) = \beta \langle\!\langle x, W z_y \rangle\!\rangle + (1 - \beta) \langle\!\langle W^T x, z_y \rangle\!\rangle, \tag{4}$$

with $\langle\!\langle \cdot, \cdot \rangle\!\rangle$ denoting the cosine similarity operator, *i.e.* dot-product between L_2-normalized vectors, and $0 \leq \beta \leq 1$ is a trade-off parameter. Note that, although the projection by W is a linear operation, the similarity function is not since the normalization is over $W z_y$ and $W^T x$, respectively. Eq. (4) defines a similarity score which is a combination of two measures defined independently on the visual and semantic spaces, with their relative strength balanced by β. From Eq. (4) we define the following softmax posterior:

$$p(y|x) = \frac{\exp\left(\tau F_\beta(x, z_y; W)\right)}{\sum_{y'} \exp\left(\tau F_\beta(x, z_{y'}; W)\right)}, \tag{5}$$

with τ a temperature scaling parameter [13]. Instead of treating τ as a hyperparameter to be tuned, we learn it from data together with the projection matrix W. The rationale behind this choice is to improve convergence by allowing the model to expand the range of the exponential arguments beyond the $[-1,+1]$ interval induced by the cosine metric [35]. From Eq. (5) we can derive the following loss:

$$\mathcal{L}_{F_\beta}(W;\mathcal{D}^{tr}) = \sum_{x,y} \ell_{F_\beta}(x,y;W) = \sum_{x,y} -\log p(y|x)$$
$$= \sum_{x,y} -\tau F_\beta(x,z_y;W) + \underset{y'}{\text{LSE}}(\tau F_\beta(x,z_{y'};W)) \quad (6)$$

with LSE the log-sum-exp operator and the sums run over training pairs $(x,y) \in \mathcal{D}^{tr}$. Predictions are performed as in Eq. (1) but using F_β instead of F.

4.2. Weighted loss formulation

In the previous section we defined a scoring function as a weighted combination of the similarities computed on the visual and semantic spaces (Eq. (4)) and used it to model a conditional density over the set of known categories (Eq. (5)) in a softmax manner. Instead, we could model a conditional distribution on each space independently and aggregate them via an adequate pooling mechanism. Let us begin by defining:

$$p_{\mathcal{X}}(y|x) = \frac{\exp\left(\mu\langle\!\langle x, Wz_y\rangle\!\rangle\right)}{\sum_{y'}\exp\left(\mu\langle\!\langle x, Wz_{y'}\rangle\!\rangle\right)}, \quad (7)$$

$$p_{\mathcal{Z}}(y|x) = \frac{\exp\left(\nu\langle\!\langle W^T x, z_y\rangle\!\rangle\right)}{\sum_{y'}\exp\left(\nu\langle\!\langle W^T x, z_{y'}\rangle\!\rangle\right)} \quad (8)$$

where $\mathcal{X}$ and $\mathcal{Z}$ denote the space used for similarity computation and $\mu,\nu \in \mathbb{R}_+$ trainable scaling parameters. Let us now consider as scoring function the maximum of the posteriors given by Eq. (7)–(8), *i.e.*

$$G(x,y;W) = \max\left(p_{\mathcal{X}}(y|x), p_{\mathcal{Z}}(y|x)\right). \quad (9)$$

Following a similar reasoning as in the previous section, we can define the following optimization objective[1]:

$$\mathcal{L}_G(W;\mathcal{D}^{tr}) = -\sum_{(x,y)\in\mathcal{D}^{tr}} \log G(x,y;W). \quad (10)$$

However, in doing so, we have lost the ability to trade-off the relative importance of the visual and semantic spaces as we did in Eq. (4). We can recover such flexibility by noting that, for any $0 \le \beta \le 1$, the following upper bound holds[2]:

$$-\log G \le -\left[\beta \log p_{\mathcal{X}} + (1-\beta)\log p_{\mathcal{Z}}\right]. \quad (11)$$

Therefore, instead of directly optimizing Eq. (10), we can minimize the following upper bound on $\mathcal{L}_G$:

$$\mathcal{L}_{G,\beta}(W;\mathcal{D}^{tr}) =$$
$$-\beta\sum_{x,y}\log p_{\mathcal{X}}(y|x) - (1-\beta)\sum_{x,y}\log p_{\mathcal{Z}}(y|x). \quad (12)$$

This loss can be seen as the convex combination of the losses that would have been derived from $p_{\mathcal{X}}$ and $p_{\mathcal{Z}}$ by following the same reasoning that led to Eq. (6). For any $0 \le \beta \le 1$ it holds also that $\mathcal{L}_G(W;\mathcal{D}^{tr}) \le \mathcal{L}_{G,\beta}(W;\mathcal{D}^{tr})$.

4.3. Relation between $\mathcal{L}_{F_\beta}$ and $\mathcal{L}_{G,\beta}$

Let us consider $\mu = \nu = \tau$ in Eq. (7)–(8). Using the definition of F_β given by Eq. (4) we can write the per-sample losses in Eq. (6) and (12) as follows:

$$\ell_{F_\beta} = -\tau F_\beta + \text{LSE}(\beta\tau F_1 + (1-\beta)\tau F_0)$$
$$\ell_{G,\beta} = -\tau F_\beta + \beta\text{LSE}(\tau F_1) + (1-\beta)\text{LSE}(\tau F_0).$$

From the above, it can be seen that both objectives promote that images and attributes of the same class are pulled together (F_β), while those from different classes are pushed away. The difference stem on the way the later is promoted. If we think on LSE as an approximation to the maximum, we see that while ℓ_{F_β} penalizes the difference between F_β and the maximum among all classes, $\ell_{G,\beta}$ penalizes the difference between F_β and the weighted average of the maxima on each space. Under the assumption of equal temperatures and from the convexity of the LSE function we have $\ell_{F_\beta} \le \ell_{G,\beta}$.

5. Experiments

We follow the protocol proposed in Xian *et al.* [40] in all our experiments. Details are provided next.

Datasets. We report experiments on the following datasets: Caltech UCSD Birds 200-2011 (CUB) [34], Animals with Attributes 1 and 2 (AWA1 & AWA2) [40], attribute Pascal & Yahoo (APY) [9], SUN attributes (SUN) [25] and Oxford flowers (FLO) [23]. CUB is a fine-grained datasets containing 11788 images of 200 different bird species each of which is annotated with 312 attributes. AWA1 contains 30475 images of 50 animal species described by 85-dimensional attribute vectors. AWA2 is an updated version of the same dataset consisting of 37322 images from the same classes. We decided to include both AWA1 and AWA2 in order to compare with other works that report results on either of both. APY is a coarse grained dataset depicting 32 classes described by 64 attributes. SUN is a fine-grained dataset containing 14340 images from 717 different visual scenes. Each scene is annotated with 102 different attributes. Finally, FLO is a fine-grained dataset

[1] Eq. (6) follows from adopting maximum-likelihood criteria after Eq. (5). Since Eq. (9) is not a valid pdf, it does not allows for the same formal derivation. However, the loss given by Eq. (10) appears as a sensible choice.

[2] For any $p,q \ge 0$ and $0 \le \alpha \le 1$ holds that $max(p,q) \ge p^\alpha q^{1-\alpha}$. Eq. (11) follows from applying logarithms on both sides and inverting the inequality.

Table 1. Statistics of the datasets used in our experiments.

Number of ...	SUN	FLO	CUB	AWA2	AWA1	APY
seen categories	645	82	150	40	40	20
unseen categories	72	20	50	10	10	12
attributes	102	1024	312	85	85	64
samples	14340	8189	11788	37322	30475	15339
training samples	10320	5631	7057	23527	19832	5932
test seen samples	2580	1403	1764	5882	4958	1483
test unseen samples	1440	1155	2967	7913	5685	7924
granularity	fine	fine	fine	coarse	coarse	coarse

with 8189 images of 102 types of flowers. Attribute vectors in this case correspond to 1024 dimensional embeddings computed from a set of fine-grained visual descriptions for each class [27]. Dataset statistics are shown in Table 1.

Evaluation Metric Zero-shot performance is measured by the average per-class top-1 accuracy (Top-1 Acc.), *i.e.* by computing the average top-1 accuracy per-class and then averaging results across classes. Evaluation on the ZSC setting is performed on $\mathcal{Y}^{ts}$ while for GZSC it is performed on $\mathcal{Y}^{tr} \cup \mathcal{Y}^{ts}$. We follow [40] and report Top-1 Acc. for the train (*tr*) and test (*ts*) sets as well as their harmonic mean (H).

Experimental Setup We follow the experimental setup proposed in Xian *et al*. We use the same class splits ("ps" in [40]) as well as the same visual and attribute representation released by the authors. They correspond to features extracted by a ResNet101 model pretrained on ImageNet and per-class real valued attribute vectors. In the case of FLO, we use the features and class split proposed by [27]. Hyperparameter tuning is performed independently on each dataset by training different models on three different subsets of the "trainval" partition and selecting the value that maximizes the average top-1 accuracy across the different splits. All our models were implemented in pytorch [24]. To train our models we use SGD with momentum and an exponential learning rate decay with a factor of 0.9. We use a batch size of 64 and a learning rate of 10^{-2} in all our experiments. We validate the number of epochs (up to a max. of 100) and parameter β in Eq. (4) and (12). Temperature scaling τ, μ and ν in Eq. (5), (7) and (8) are learned along with W. As a pre-processing step, we apply a L_2-normalization on both the visual and semantic representations. No further transformations of the inputs or outputs are applied. The code is available at `http://github.com/jadrs/zsl`.

5.1. Effect of the trade-off parameter β

In this section we evaluate the effect of the parameter β on the models outlined in Sec. 4.1 and Sec. 4.2. Fig. 2 shows the average Top-1 validation accuracy over the validation splits for the SUN and AWA1 datasets as a function of the trade-off parameter β. WS denotes the weighted similarity formulation relying on the scoring function F_β (Eq. (4)) and loss function $\mathcal{L}_{F_\beta}$ (Eq. (6)). WL to the weighted loss formulation with scoring function G (Eq. (9)) and loss $\mathcal{L}_{G,\beta}$ (Eq. (12)). We report the mean Top-1 Acc. over the validation splits and error bars at ±1 standard deviation. From the figure, it can be seen that different values of β result in different performance metrics. For WSF, setting $\beta = 0$ ($\beta = 1$) is equivalent to using similarity scores defined only over the semantic (visual) space. Different choices for this parameter offer different trade-offs among these two regimes. For the SUN dataset, if we consider the WS formulation we observe a peak between $\beta = 0.6$ and 0.7, *i.e.* at a slightly higher weight to the visual side. For WL, the interpretation is more subtle because of the mismatch between the loss ($\mathcal{L}_{G,\beta}$) and scoring function (G) definitions. Nevertheless, we observe a similar trend w.r.t. β. In the

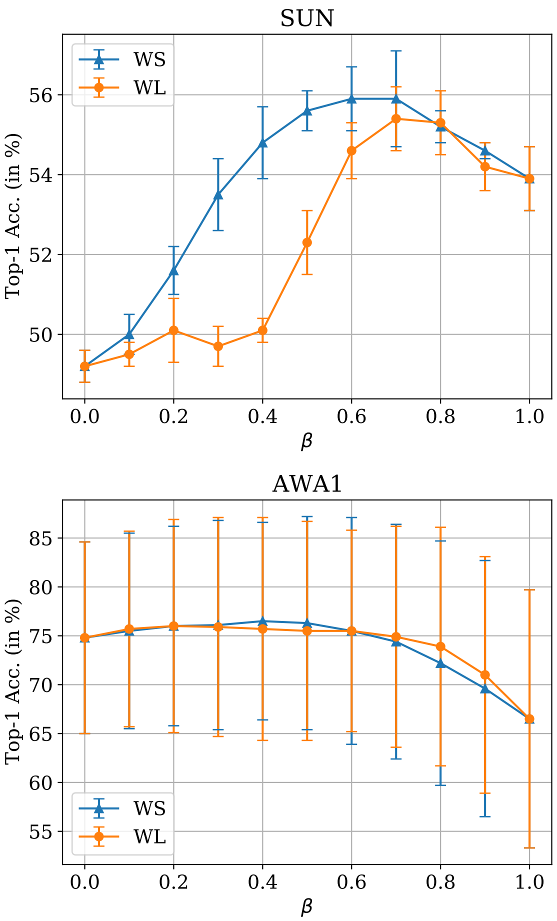

Figure 2. Average top-1 accuracy on the validation set of the AWA1 (left) and SUN datasets (right) as a function of the trade-off parameter β. Errorbars at ±1 standard deviation are shown. WSF and WL denote weighted similarity and loss formulations, respectively.

Table 2. Values of β found by cross validation.

SUN	FLO	CUB	AWA2	AWA1	APY
0.7	0.8	0.6	0.5	0.5	0.8

case of the AWA1 dataset, WS and WL behave similarly for different values of the parameter. Besides the higher variance in the results, we observe a peak around $\beta = 0.5$. We also observe a higher degradation of performance for higher values of β. This is interesting since it reveals that, beyond the chosen formulation, the optimal balance between visual and semantic information is characteristic of each particular dataset (and the chosen representation spaces). Table 2 show the values of the parameter β chosen for each dataset using the validation procedure outlined above. Interestingly, we observe a preference towards the visual term. This is consistent with the observations made by [44, 14] regarding the better performance observed by models based on projections onto the visual space.

5.2. Zero-Shot Classification Experiments

In this section we compare our approach with others methods from the literature. We consider the following groups of methods: 1) those based on simple bilinear forms as the one considered in this paper, *i.e*. DeViSE [10], SJE [3], ESZSL [28]and ALE [2]; 2) methods based on more elaborated formulations, as GFZSL [33], PSR [4], ZSKL[3] [43], and MLSE [8]; and 3) models for which class embeddings are assumed to be available during training, *i.e*. conditional feature generation network of of [41] (FGN+softmax) and the contrastive learning approach of [15] (TCN). Although this deviates from the base ZSL setting, we decided to include them in our experiments since they can be used transparently in conjunction with a wide range of models. For a fairer comparison, our results are based on our own implementation of the feature generation network (FGN) of [41] (marked as “rep.” in the table). We do not include in the evaluation some recent works (*e.g*. [21, 5, 20, 31, 42]) as they either use different set of features than those proposed by Xian *et al*. or do not conform to the same training and evaluation procedure. Evaluation of the impact of different combinations of visual and semantic representations as well as the learning of better encoding mechanisms is beyond the scope of this paper. Besides WS and WL, we also include a baseline model consisting of a simple bilinear form over L_2-normalized signatures and a temperature Eq. (6). We denote this system as “bilinear”. Table 3 shows the average per-class top-1 accuracy for these methods. Those marked by an asterisk ($\star$) are reproduced from [40].

Table 3. ZSC performance as the average per-class top-1 accuracy over unseen test classes as proposed by Xian *et al*. Results marked with a star ($\star$) are reproduced from [40, 41].

Method	SUN	FLO	CUB	AWA2	AWA1	APY
DeViSE*	56.5	45.9	52.0	59.7	54.2	39.8
SJE*	53.7	53.4	53.9	61.9	65.6	32.9
ESZSL*	54.5	51.0	53.9	58.6	58.2	38.3
ALE*	58.1	48.5	54.9	62.5	59.9	39.8
GFZSL*	60.6	-	49.3	63.8	68.3	38.4
PSR	61.4	-	56.0	63.8	-	38.4
ZSKL	61.7	-	51.7	70.5	70.1	45.3
MLSE	62.8	-	64.2	67.8	-	46.2
TCN	61.5	-	59.5	71.2	70.3	38.9
FGN+softmax*	60.8	67.2	57.3	-	68.2	-
FGN+softmax (rep.)	60.8	56.7	59.5	65.8	70.1	35.1
Ours (bilinear)	57.6	50.5	43.5	64.7	63.4	38.4
Ours (WS)	63.9	62.4	57.0	64.1	63.9	40.5
Ours (WL)	63.1	62.3	54.4	65.1	67.7	39.0
Ours (FGN+bilinear)	64.2	66.2	60.6	70.7	71.8	43.1
Ours (FGN+WS)	62.8	64.9	62.5	66.0	65.7	38.2
Ours (FGN+WL)	64.4	63.6	62.3	69.1	69.2	39.9

Compared to the first group of methods we observe that, while the bilinear baseline achieves a comparable classification performance, both WS and WL exhibit a significant improvement. If we consider more complex models as those from the second group, WS and WL improve over GFZSL and PSR for all dataset but AWA1. Compared to MLSE, we improve only on the sun dataset but remain behind on the others. Compared to ZSKL, our models show improvements only for the fine grained datasets (SUN and CUB). Notwithstanding the performance gap, the proposed approach is based on a much simpler formulation. For instance, PSR requires different architectures designs for different datasets while MLSE relies on an rather involved optimization procedure.

Finally, we follow [41] and train a conditional feature generator based on the attribute and visual descriptors of seen classes. We then generate 300 synthetic samples for each class from the test set and use them to train a softmax classifier on them. Table 3 show the results reported in [41] as well as our own implementation. We were able to reproduce the results by Xian *et al*. (and in some cases with a slight improvement) for all datasets but FLO, where we observe a large gap in performance. Results obtained with the softmax model on synthetic features perform on par with the models from the second group. If we now train our models using features from the training and synthetic test sets, we observe a clear improvement in all cases. Remarkably, the largest improvement is observed for the simple bilinear baseline, specially on the fine-grained problems. WL appears as to have an edge over WS but the difference is not conclusive. Note however that in this scenario we assume full knowledge of the output attributes. In case where this information is not available at training time, WS and WL offer a good trade-off between model simplicity and overall classification performance. Furthermore, the insights

[3] “Gaussian-Ort” kernel in [43].

gained by exploring bilinear symmetries can be useful in other application domains *e.g.* multimodal problems, metric learning, etc.

5.3. Generalized Zero-Shot Classification Experiments

In this section we analyze results obtained under the generalized setting. In Table 4 we show results for the same datasets as in the previous section as well as for the same groups of methods. The only difference is that, for the FGN+X[4] models, now we generate 2000 samples per class instead of 300 as we did before. Also in this case, we were able to reproduce the results reported Xian *et al.* on all datasets but FLO, where we still observe a large performance gap. Again, we believe this gap could be made smaller by a more aggressive parameter tuning but such evaluation is beyond the scope of our work. TCN improves over FGN+softmax* on all dataset but SUN. Overall, the performance observed by methods from this group exhibit a large improvement compared to those from the first and second, showing the importance of having access to some information of the target classes during training.

From the table, we observe that our models (bilinear, WS and WL) perform on par or better than the models from the first and second group on all datasets except APY. On this dataset both WS and WL are behind the bilinear baseline which in turn show a performance which sits between the models of the first and second group. WS and WL show similar performance overall, improving also over the bilinear baseline. Combined with generated test features, our models show a large performance boost, improving on more than 10 absolute points in the harmonic mean over the models trained on seen samples alone. As noted in [41], is due to a reductions of the number of samples from the test (unseen) categories that are missclassified as belonging to any of the seen classes. This is supported by the large improvement observed in the "ts" column for models FGN-x compared to their simpler counterpart.

6. Conclusions

In this work we developed around the idea that simple zero-shot formulations based on bilinear compatibility functions can be seen as a two-step process consisting on a projection followed by a similarity computation on the target space. This process can be formulated from the visual to the semantic space and *vice versa*. We explored this symmetry by proposing two different formulations, namely: *i)* by redefining the similarity metric as a weighted combination of the similarities on each space, and *ii)* by deriving an upper bound to a suitable loss function that can be expressed as a combination of the losses acting on each space. Experiments on different dataset both in the standard and generalized settings showed promising results. Due to the widespread use of learnable cosine metrics in the literature, we believe this work can be found useful in other application domains.

[4] We denote FGN-bilinear, FGN-WS or FGN-WL generically as FGN-X.

Acknowledgments

This work used computational resources from CCAD-UNC, which is part of SNCAD-MinCyT, Argentina.

References

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

[2] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *IEEE Tr. on Pattern Analysis and Machine Intelligence*, 38(7):1425–1438, 2015.

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

[4] Yashas Annadani and Soma Biswas. Preserving semantic relations for zero-shot learning. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 7603–7612, 2018.

[5] Yannick Le Cacheux, Herve Le Borgne, and Michel Crucianu. Modeling inter and intra-class relations in the triplet loss for zero-shot learning. In *Proc. of the IEEE Intl. Conf. on Computer Vision*, pages 10333–10342, 2019.

[6] Nello Cristianini, John Shawe-Taylor, Andre Elisseeff, and Jaz S Kandola. On kernel-target alignment. In *Advances in Neural Information Processing Systems*, pages 367–373, 2002.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conf. on Computer Vision and pattern recognition*, pages 248–255. Ieee, 2009.

[8] Zhengming Ding and Hongfu Liu. Marginalized latent semantic encoder for zero-shot learning. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 6191–6199, 2019.

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

[10] 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*, pages 2121–2129, 2013.

[11] Yanwei Fu, Timothy M Hospedales, Tao Xiang, and Shaogang Gong. Transductive multi-view zero-shot learning.

Table 4. GZSC performance as the average per-class top-1 accuracy over seen and unseen test classes. Following, Xian *et al*. we report the average per-class top-1 accuracy over train (*tr*) and test (*ts*) classes as well as their harmonic mean (*H*). Results with a star (*) are reproduced from [40, 41].

	SUN			FLO			CUB			AWA2			AWA1			APY		
Method	*ts*	*tr*	*H*	*ts*	*tr*	*H*	*ts*	*tr*	*H*	*ts*	*tr*	*H*	*ts*	*tr*	*H*	*ts*	*tr*	*H*
DeViSE*	16.9	27.4	20.9	9.9	44.2	16.2	23.8	53.0	32.8	17.1	74.7	27.8	13.4	68.7	22.4	4.9	76.9	9.2
SJE*	14.7	30.5	19.8	13.9	47.6	21.5	23.5	59.2	33.6	8.0	73.9	14.4	11.3	74.6	19.6	3.7	55.7	6.9
ESZSL*	11.0	27.9	15.8	11.4	56.8	19.0	12.6	63.8	21.0	5.9	77.8	11.0	6.6	75.6	12.1	2.4	71.1	4.6
ALE*	21.8	33.1	26.3	23.7	62.8	34.4	13.3	61.6	21.9	14.0	81.8	23.9	16.8	76.1	27.5	4.6	73.7	8.7
GFZSL*	0.0	39.6	0.0	-	-	-	0.0	45.7	0.0	2.5	80.1	4.8	1.8	80.3	3.5	0.0	83.3	0.0
PSR	20.8	37.2	26.7	-	-	-	20.7	73.8	32.3	24.6	54.3	33.9	-	-	-	13.5	51.4	21.4
ZSKL	20.1	31.4	24.5	-	-	-	21.6	52.8	30.6	18.9	82.7	30.8	17.9	82.2	29.4	10.5	76.2	18.5
MLSE	20.7	36.4	26.4	-	-	-	22.3	71.6	34.0	23.8	83.2	37.0	-	-	-	12.7	74.3	21.7
TCN	31.2	37.3	34.0	-	-	-	52.6	52.0	52.3	61.2	65.8	63.4	49.4	76.5	60.0	24.1	64.0	35.1
FGN+softmax*	42.6	36.6	39.4	59.0	73.8	65.6	43.7	57.7	49.7	-	-	-	57.9	61.4	59.6	-	-	-
FGN+softmax (rep.)	43.4	34.9	38.7	35.8	66.6	46.5	47.4	48.8	48.1	40.5	76.8	53.0	50.4	69.9	58.6	14.7	71.4	24.4
Ours (bilinear)	18.7	28.4	22.6	17.4	32.8	22.7	20.3	43.6	27.7	10.0	88.5	18.1	12.4	85.3	21.6	6.8	69.0	12.4
Ours (WS)	20.1	39.4	26.6	35.4	72.6	47.6	23.2	65.7	34.3	19.3	88.2	31.7	17.0	89.1	28.6	4.6	84.2	8.8
Ours (WL)	19.9	39.1	26.4	34.8	72.3	47.0	21.5	61.9	31.9	19.5	89.8	32.0	18.0	86.8	29.9	4.8	81.2	9.1
Ours (FGN+bilinear)	48.8	26.8	34.6	58.3	41.3	48.3	53.5	44.5	48.6	57.9	53.5	55.6	65.5	39.5	49.3	25.3	56.0	34.8
Ours (FGN+WS)	49.7	30.3	37.7	58.7	40.8	48.1	47.5	55.1	51.0	38.6	85.3	53.2	44.7	81.5	57.7	11.0	79.1	19.4
Ours (FGN+WL)	46.3	28.7	35.5	56.0	52.7	54.3	49.4	51.8	50.6	45.8	83.4	59.2	52.2	78.6	62.8	17.9	73.3	28.7

IEEE Tr. on Pattern Analysis and Machine Intelligence, 37(11):2332–2345, 2015.

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[13] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *Proc. of the 34th Intl. Conf. on Machine Learning-Volume 70*, pages 1321–1330. JMLR. org, 2017.

[14] Huajie Jiang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Learning class prototypes via structure alignment for zero-shot recognition. In *Proc. of the European Conf. on Computer Vision*, pages 118–134, 2018.

[15] Huajie Jiang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Transferable contrastive network for generalized zero-shot learning. In *Proc. of the IEEE Intl. Conf. on Computer Vision*, pages 9765–9774, 2019.

[16] Thorsten Joachims. Optimizing search engines using clickthrough data. In *Proc. of the eighth ACM SIGKDD Intl. Conf. on Knowledge discovery and data mining*, pages 133–142, 2002.

[17] Elyor Kodirov, Tao Xiang, and Shaogang Gong. Semantic autoencoder for zero-shot learning. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 3174–3183, 2017.

[18] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE Tr. on Pattern Analysis and Machine Intelligence*, 36(3):453–465, 2013.

[19] Jimmy Lei Ba, Kevin Swersky, Sanja Fidler, et al. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *Proc. of the IEEE Intl. Conf. on Computer Vision*, pages 4247–4255, 2015.

[20] Kai Li, Martin Renqiang Min, and Yun Fu. Rethinking zero-shot learning: A conditional visual classification perspective. In *Proc. of the IEEE Intl. Conf. on Computer Vision*, pages 3583–3592, 2019.

[21] Yang Liu, Jishun Guo, Deng Cai, and Xiaofei He. Attribute attention for semantic disambiguation in zero-shot learning. In *Proc. of the IEEE Intl. Conf. on Computer Vision*, pages 6698–6707, 2019.

[22] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[23] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conf. on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008.

[24] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8026–8037, 2019.

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

[26] Hang Qi, Matthew Brown, and David G Lowe. Low-shot learning with imprinted weights. In *Proc. of the IEEE Conf. on Computer Vision and pattern recognition*, pages 5822–5830, 2018.

[27] Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. Learning deep representations of fine-grained visual descrip-

tions. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 49–58, 2016.

[28] Bernardino Romera-Paredes and Philip Torr. An embarrassingly simple approach to zero-shot learning. In *Intl. Conf. on Machine Learning*, pages 2152–2161, 2015.

[29] Edgar Schonfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. Generalized zero-and few-shot learning via aligned variational autoencoders. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 8247–8255, 2019.

[30] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pages 4077–4087, 2017.

[31] Tristan Sylvain, Linda Petrini, and Devon Hjelm. Locality and compositionality in zero-shot learning. In *Intl. Conf. on Learning Representations*, 2020.

[32] Ioannis Tsochantaridis, Thorsten Joachims, Thomas Hofmann, and Yasemin Altun. Large margin methods for structured and interdependent output variables. *Journal of Machine Learning Research*, 6(Sep):1453–1484, 2005.

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

[34] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[35] Feng Wang, Xiang Xiang, Jian Cheng, and Alan Loddon Yuille. Normface: L2 hypersphere embedding for face verification. In *Proc. of the 25th ACM Intl. Conf. on Multimedia*, pages 1041–1049, 2017.

[36] Wei Wang, Vincent W Zheng, Han Yu, and Chunyan Miao. A survey of zero-shot learning: Settings, methods, and applications. *ACM Tr. on Intelligent Systems and Technology (TIST)*, 10(2):1–37, 2019.

[37] Xinsheng Wang, Shanmin Pang, Jihua Zhu, Zhongyu Li, Zhiqiang Tian, and Yaochen Li. Visual space optimization for zero-shot learning. *arXiv preprint arXiv:1907.00330*, 2019.

[38] Yu-Xiong Wang, Ross Girshick, Martial Hebert, and Bharath Hariharan. Low-shot learning from imaginary data. In *Proc. of the IEEE Conf. on Computer Vision and pattern recognition*, pages 7278–7286, 2018.

[39] Jason Weston, Samy Bengio, and Nicolas Usunier. Wsabie: Scaling up to large vocabulary image annotation. 2011.

[40] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *IEEE Tr. on Pattern Analysis and Machine Intelligence*, 41(9):2251–2265, 2018.

[41] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *Proc. of the IEEE Conf. on Computer Vision and pattern recognition*, pages 5542–5551, 2018.

[42] Wenjia Xu, Yongqin Xian, Jiuniu Wang, Bernt Schiele, and Zeynep Akata. Attribute prototype network for zero-shot learning. In *NeurIPS*, 2020.

[43] Hongguang Zhang and Piotr Koniusz. Zero-shot kernel learning. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 7670–7679, 2018.

[44] Li Zhang, Tao Xiang, and Shaogang Gong. Learning a deep embedding model for zero-shot learning. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 2021–2030, 2017.