Multimedia Tools and Applications
https://doi.org/10.1007/s11042-023-14877-1

Learning semantic ambiguities for zero-shot learning

Celina Hanouti[1] **· Hervé Le Borgne**[2]

Received: 7 February 2022 / Revised: 30 May 2022 / Accepted: 6 February 2023

Abstract

Zero-shot learning (ZSL) aims at recognizing classes for which no visual sample is available at training time. To address this issue, one can rely on a semantic description of each class. A typical ZSL model learns a mapping between the visual samples of seen classes and the corresponding semantic descriptions, in order to do the same on unseen classes at test time. State of the art approaches rely on generative models that synthesize visual features from the prototype of a class, such that a classifier can then be learned in a supervised manner. However, these approaches are usually biased towards seen classes whose visual instances are the only one that can be matched to a given class prototype. We propose a regularization method that can be applied to any conditional generative-based ZSL method, by leveraging only the semantic class prototypes. It learns to synthesize discriminative features for possible semantic description that are not available at training time, that is the unseen ones. The approach is evaluated for ZSL and GZSL on four datasets commonly used in the literature, either in inductive or transductive settings, with results on-par or above state of the art approaches. The code is available at https://github.com/hanouticelina/lsa-zsl.

Keywords Zero-shot learning · Generative latent space · Semantic ambiguity

1 Introduction

Being able to classify, detect or segment objects into images with as less annotated data as possible is one of the most important problem addressed to implement practical application in computer vision. A radical framework is proposed by the zero-shot learning (ZSL), in which not a single visual example is used during learning, but where it is possible to rely on external data from another modality. Typically, the latter are semantic attributes or textual descriptions that can be represented by vectors. Hence, the task consists in learning a

✉ Hervé Le Borgne
herve.le-borgne@cea.fr

Celina Hanouti
celina.hanouti@wefox.com

[1] wefox, Berlin, Germany

[2] Université Paris-Saclay, CEA, List, Palaiseau, F-91120, France

Published online: 31 March 2023

mapping between the image space and the semantic space using images from seen classes, available at training time only. In the original form of ZSL, the images of the test set belong to unseen classes, for which no sample is available at training time. A more realistic "generalized" setting (GZSL) [30] proposes nevertheless to recognize both seen and unseen classes at test time. Another classical distinction is made between the inductive and the transductive setting, the latter allowing to use test images (without their annotation) at training time, similarly to semi-supervised learning.

Recent approaches to ZSL use generative models to produce visual samples from unseen classes based on their semantic descriptions [3, 4, 27, 29]. With such synthetic samples, we thus have a classical supervised learning setting for the unseen classes as well. One of the most performing approaches in this vein is f-VAEGAN-D2 [31] that shares the weights of the decoder of a variational autoencoder (VAE) with those of the generator of a generative adversarial network (GAN). Since it is trained in combination with a conditional encoder (and either a conditional or a non-conditional discriminator), it is able to benefit from unlabeled unseen visual samples (transductive setting) to synthesize discriminative image features of unseen classes. It also obtains very good performance in inductive setting as well. Narayan et al. [16] proposed TF-VAEGAN that add a decoder which reconstructs the semantic prototypes and a feedback loop from this module to the GAN generator to refine the generated features during both the training and feature synthesis stages. The motivation to add a decoder is that it can provide a complementary information along with the generator, since the latter maps a single prototype to many possible visual instance while the decoder does the opposite. Both pieces of information can be used at test time to create features that are used to learn ZSL and GZSL classifiers.

Beyond the usefulness of the feedback loop, one can see TF-VAEGAN as similar to f-VAEGAN-D2 with an additional loss that regularizes the generator. However, this loss essentially addresses a reconstruction task, while ZSL consists first and foremost in discriminating classes. We thus argue that a loss that regularizes in a way that favor class disambiguation would be more relevant. The loss we propose (Section 3) can be integrated to any generative model for ZSL. The idea is to train the generator to learn some ambiguous semantic prototypes built by mixing real available ones, and recognize the corresponding ambiguous classes. This idea may seem similar to the one proposed by [6] who applied mixup [33] to the ZSL task, but it has a crucial difference. In fact, when they apply mixup, Chou et al. matches the virtual prototype to a corresponding virtual visual feature. Our approach focuses on recognizing a virtual class, thus its label only. Hence, the regularization forces the generator to synthesize discriminative features from unknown class prototypes, some of them being potentially close to some prototypes of the unseen classes (see Section 4.4.2 for a detailed discussion and evaluation with regards to the availability of the seen/unseen prototypes during training). The generator nevertheless learns without visual samples from seen classes and is thus not constrained by particular images that are not relevant. Indeed, mixing two particular images does not usually result into a meaningful image, while mixing semantic description or attributes may make sense (Fig. 1). Therefore, restricting mixup to the semantic space corresponds better to what is expected in the (G)ZSL task. Our approach is computationally less expensive and leads to better results than Chou et al. in practice (Section 4). Last, the loss they defined is mainly useful in an inductive setting, while the approach we propose can be even more useful in a transductive one.

Our main contribution consists in a regularization loss, that can be applied to any conditional generative-based ZSL model. Integrated to f-VAEGAN-D2 or TFVAEGAN, it

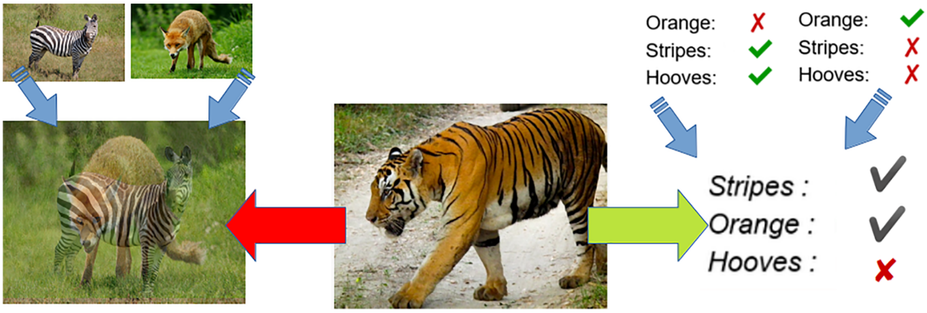

Fig. 1 An image of *tiger* differs from a mix of an image of *zebra* and *fox*. At the opposite, learning to discriminate a mix of semantic attributes (or description) of a fox and a zebra makes the proposed model able to better identify a tiger at test time

improves significantly their performance on different benchmarks, either in inductive or transductive setting.

2 State of the art

Early approaches in ZSL relied on attribute prediction [8], ridge regression [21] or triplet-loss [7, 9, 10]. We refer to [12] for a more detailed overview of these approaches as we will focus on generative approaches in the following.

In order to address the biased prediction towards seen classes, generative approaches synthesize visual features of unseen classes from their semantic features with generative models like Variational Autoencoders (VAEs) or Generative adversarial networks (GANs). [29] combines a conditional Wasserstein GAN [2] with a categorization network to generate more discriminative features. Bucher et al. [4] proposed three different conditional GANs to generate features, Generative Moment Matching Network (GMMN), AC-GAN, and Denoising Auto-Encoder. Other works use conditional VAE. Arora et al. [3] integrates an attribute regressor and a feedback mechanism into a VAE-based model to generate more discriminative features and ensure that the generated features are semantically close to the distribution of the real features. Schonfeld et al. [22] proposes to align the visual features and the corresponding semantic embeddings in a shared latent space, using two Variational Autoencoders (VAEs). Recent works take advantage of both GANs and VAEs by combining them with shared decoder and generator. Xian et al. [31] proposes a VAEGAN-based model that leverages the unlabeled instances under the transductive setting via an additional unconditional discriminator. Similar to the idea proposed by Arora et al. [16] augments the f-VAEGAN-D2 method with a semantic embedding decoder and a feedback mechanism to enforce a semantic consistency and improve feature synthesis. In this work, we propose to enrich a conditional VAEGAN-based ZSL method with an auxiliary task as well while focusing on another aspect, more related to the ability to discriminate classes, namely reducing ambiguities among categories. Such a goal can be useful beyond zero-shot learning, for tasks that aim at relating ambiguous visual and semantic information such as multimodal entity linking [1] and retrieval [15, 34], cross-modal retrieval [5, 24, 25] or classification [26].

To alleviate the domain shift problem, transductive ZSL methods are proposed to leverage the unlabeled unseen-class data at training. Xian et al. [30] and Ye and Guo

[32] proposed to use graph-based label propagation while [27] uses an Expectation-Maximization (EM) strategy, where pseudo-labeled unseen class examples were used to update the parameter estimates of unseen class distributions. Generative models were also applied to transductive ZSL. Paul et al. [19] leverages Wasserstein GAN [2] to synthesize the unseen domain distribution via minimizing the marginal difference between the true latent space representation of the unlabeled samples of unseen classes and the synthesized space. f-VAEGAN-D2 and TF-VAEGAN can also be applied to transductive settings.

Since its advent, interpolation-based regularization has been shown to be a surprisingly effective method to improve generalization and robustness on both supervised and semi-supervised settings. Zhang et al. [33] proposed *mixup*, a data augmentation technique for image classification consisting in creating virtual training examples constructed as the convex combinations of pairs of visual data samples and their corresponding labels. This simple approach has shown to be an effective model regularizer that favor linear behavior in-between training examples. Recently, [6] applied *mixup* to zero-shot learning. Similarly to our method, they interpolate both the visual samples and the semantic prototypes. However, unlike our approach, they used mixup as a direct data augmentation approach, while we apply the interpolation in the conditional space of a generative ZSL model and propose a specific regularization loss in the semantic space. More specifically, we train a conditional generative ZSL model to recognize virtual ambiguous classes. The generator synthesizes features from the corresponding ambiguous class prototypes, which are then used to perform the classification task. In practice, the difference between the linear interpolation we propose and the usual mixup setting used by [6] is reflected by the mixing proportion leading to the best performances. Indeed, as an augmentation data approach, mixup usually have better performances with a mixing proportion that must be either close to 0 or 1, making the new virtual samples pretty close to the original ones. In contrast, we obtain the best performances with a mixing proportion close to 0.5, making the new *virtual* classes completely distinct from the real ones. These classes are different from the actual unseen classes used at test time, but allow the generator to be regularized in some 'empty' parts of the semantic space.

3 Method

3.1 Problem setting and notation

Let us consider a set of images $X = \{x_1, ..., x_l, x_{l+1}, ..., x_t\}$ encoded in the image feature space $\mathcal{X} = \mathbb{R}^d$ and two disjoint sets of class labels: a seen class label set $\mathcal{Y}^s$ and the unseen one $\mathcal{Y}^u$. The set of class prototypes is denoted as $C = \{c(y) \| y \in \mathcal{Y}^s \cup \mathcal{Y}^u, c(y) \in \mathcal{C}\}$. Usually, $c(y)$ is a vector of binary attributes, but may be word embeddings when one wants to describe a large set of classes [7, 11, 13]. The first l samples x_s, with $s \leq l$, are labeled samples from seen classes $y^s \in \mathcal{Y}^s$ and the remaining samples x_u, with $l + 1 \leq u \leq t$, are unlabeled data from novel classes, $y^u \in \mathcal{Y}^u$. In the inductive setting, the training set contains only labeled seen classes examples, and the semantic information about both seen and unseen classes. In the transductive setting, the training set contains both labeled (seen classes) and unlabeled (unseen classes) data samples. In fact, there is an ambiguity in the definition of the *transductive* setting in the literature, as there is more than one definition of this setting. Indeed, [12] defines the *class-transductive* setting, in which class prototypes of both seen and unseen classes are available during the training phase, and the *instance-transductive* setting, where both prototypes and unlabeled images from unseen classes are available. The class-transductive setting is sometimes referred as inductive, as the author considers that the unseen prototypes need to be available to generate the unseen visual samples used to learn the classifiers. In this paper, we refer to “transductive” setting when unseen prototypes are used for any other usage than generating unseen visual features.

In zero-shot learning, the goal is to predict the label of images that belong to unseen classes, i.e. $f_{zsl} : \mathcal{X} \rightarrow \mathcal{Y}^u$, while in the GZSL scenario, the task is to predict labels of images that can belong to either seen or unseen classes, i.e. $f_{gzsl} : \mathcal{X} \rightarrow \mathcal{Y}^s \cup \mathcal{Y}^u$, with f a compatibility function that computes the likelihood of an image to belong to a class.

3.2 Learning semantic ambiguities

We consider a generative approach to ZSL, for which a conditional model $G(.)$ is trained to minimize a loss $\mathcal{L}$, using visual samples x^s and prototypes $c(y^s)$ from seen classes y^s. At test time, it is able to generate visual samples x^u from a prototype $c(y^u)$ for unseen classes y^u, which are used to train a classifier in a fully supervised fashion. Applying mixup to this setting such as [6] consists in augmenting the training data with virtual pairs:

$$\begin{aligned} \tilde{c} &= \lambda c(y_i^s) + (1 - \lambda) c(y_j^s) \\ \tilde{x} &= \lambda x_i^s + (1 - \lambda) x_j^s \end{aligned} \tag{1}$$

where λ is a hyperparameter to determine and $((x_i, y_i), (x_j, y_j))$ is a couple of annotated data from seen classes randomly selected. As explained above, we argue that using the visual samples biases the generator towards seen classes. Such a bias was identified for the former ZSL approaches by [30] and led to the definition of generalized zero-shot learning (GZSL), that is the most common and challenging setting in the literature.

We thus adopt a different strategy: focusing on learning the ambiguities in the semantic space only. Indeed, as illustrated in Fig. 1 mixing two particular images does not usually make sense, because of strong inconsistencies at the pixel level. In contrast, mixing semantic information *may* make sense. First, it is at the origin of a large bestiary in fantasy literature, in science fiction and in heroic fantasy. A unicorn is described semantically as a horse with a corn, and no picture of such creature has been taken up to date, although many artist proposed some visual representation of it. However, we do not expect the generator to produce features that would result into a plausible representation but rather features able to emphasis the differences between classes, in order to better distinguish them. Hence, we propose to regularize the model with such a constraint only, while being independent of any particular existing visual representation. In practice, we create ambiguous classes as a linear interpolation of real semantic prototype pairs and their labels:

$$\begin{aligned} \tilde{c} &= \lambda c(y_i) + (1 - \lambda) c(y_j) \\ \tilde{y} &= \lambda y_i + (1 - \lambda) y_j \end{aligned} \tag{2}$$

The hyperparameter λ can be a fixed value or more generally a random variable $\lambda \sim \Lambda$. During the learning phase, the generator G synthesizes feature $\hat{x} \in \mathcal{X}$ from a latent code $z \sim \mathcal{N}(0, 1)$ conditioned by the ambiguous class prototype $\tilde{c} \in \mathcal{C}$. This image is then used as input to a classifier f, leading to the proposed regularization loss as:

$$\mathcal{L}_I = \mathbb{E}_{z,\lambda}[l(f(\hat{x}), \tilde{y})] \tag{3}$$

where l is the cross-entropy between the input $\hat{x} = G([z; \tilde{c}])$ and the target $\tilde{y}$. With its generic formulation, the proposed regularization can be applied to a large number of generative models. In the following, we integrate it to f-VAEGAN-D2 [31] as illustrated in Fig. 2 by adding $\mathcal{L}_I$ to their losses. In that case, the total loss to minimize is $\mathcal{L} = \mathcal{L}_{BCE} + \gamma \mathcal{L}_{WGAN}^s + \mathcal{L}_{WGAN}^u + \mathcal{L}_I$ with γ an hyperparameter (see Section 4.2). It can

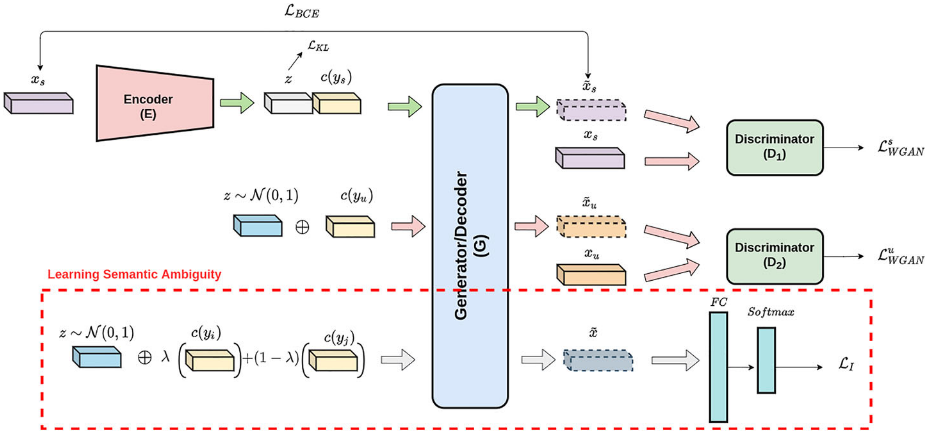

Fig. 2 Our contribution, highlighted by the red dashed rectangle, included to a model similar to f-VAEGAN-D2 [31]. The encoder E takes the real seen features x_s as input and outputs a latent code z, which is then input together with embeddings $c(y^s)$ to the generator G that synthesizes features $\tilde{x}_s$. The generator G synthesizes features of unseen classes from the class prototypes $c(y^u)$ concatenated with random noise z. The two discriminators D_1 and D_2 learn to distinguish between real and synthesized features. We introduce a novel task to train the generator G : First, virtual *ambiguous* classes are constructed as convex combinations of *real* classes, then, the generator G synthesizes a feature $\tilde{x}$ from the corresponding *ambiguous* class prototypes concatenated with random noise. Further, the synthesized features $\tilde{x}$ are used to perform a classification task

nevertheless be added to any generative-based ZSL model, and we show how it performs with TFVAEGAN in Section 4.5.

4 Experimental evaluation

4.1 Datasets and metrics

We evaluate our method on four datasets that are commonly used in the ZSL literature, namely Caltech UCSD Birds 200-2011 (CUB) [28], SUN Attribute dataset [18], Oxford Flowers (FLO) [17] and Animals with Attributes (AWA2) [30]. Their main characteristics are reported in Table 1

We applied the evaluation protocol of [30], relying on the “proposed splits” that insure that none of the test classes appear in ImageNet, since it is used to pre-train the visual

Table 1 Main characteristics of the dataset used

Dataset	Granularity	# images	#classes	Attributes size
CUB [28]	Fine	11,788	200	1024
SUN [18]	Fine	14,340	717	102
FLO [17]	Fine	8,189	102	1024
AwA2 [30]	Coarse	37,322	50	85

The granularity stands for fine-grained or coarse-grained classification feature extractor. The performances are reported in terms of average per-class top-1 accuracy (T1) for ZSL settings, and with the harmonic mean (**H**) of the average per-class top-1 accuracy on seen (s) and unseen (u) classes for GZSL. Unless otherwise specified, we use 2048-dimension 101-ResNet features as visual embeddings for all the datasets. For class semantic prototypes of CUB and FLO, we adopt the 1024-dim sentence embeddings of character-based CNN-RNN model generated from fine-grained visual descriptions [20]. For AWA2, the binary attributes relate to e.g animal species (*"fish, bird, plankton"*), color (*"black, brown, blue"*), behaviour (*"hibernate, timid, slow"*) and other features. For SUN, they rather relate to function/affordances, materials, spatial envelope and surface properties. We compare our method to TF-VAEGAN [16], f-VAEGAN-D2 [31], CLSWGAN [29] and LisGAN [14].

To compare the methods over several benchmarks and estimate their aggregated merit, we adopt the median normalized relative gain (mNRG) [23]. Indeed, such a comparison can be biased if one uses a simple average over different benchmarks. mNRG exhibits several interesting features such as an independence to outlier scores, coherent aggregation or time consistency. Its main drawback is that a reference method has to be chosen, from which the performance of each method is measured, according to a unique aggregated score, possibly negative if the method performs globally worse. In our case, we choose CLSWGAN [29] in inductive setting as reference. For the comparison with fine-tuned features, we use f-VAEGAN-D2 in inductive setting as reference. We compute the mNRG by aggregating the accuracy for ZSL and the harmonic mean H of seen and unseen accuracy for GZSL. By definition, the score of the reference is 0. If $mNRG < 0$ then the method performs globally worse than the references over all datasets.

4.2 Implementation details

The generator G and discriminators D_1 and D_2 are implemented as two-layer fully connected networks with 4096 hidden units. The generator is updated every 5 discriminator iterations [2]. The function f used in equation (3) is implemented as a two-layers fully connected network that takes an input synthesized feature of size $d = 2048$, has a hidden layer of size 4096 and outputs a probability distribution with regards to all classes of interest. We use LeakyReLU activation everywhere, except at the output of G, where a sigmoid non-linearity is applied before the binary cross-entropy loss $\mathcal{L}_{BCE}$. ZSL and GZSL classifiers are implemented as a single layer perceptron of size 2,048, trained for 20 epochs. We use Adam optimizer with a learning rate of 0.0001. Our (PyTorch) code is based on the one of [16] and is available at https://github.com/hanouticelina/lsa-zsl. We determined that the hyperparameters $\gamma = 10$ and the gradient penalty of the WGAN loss $\lambda_{WGAN} = 10$ allowed us to obtain similar performances as those reported in [16, 31], although they are sometimes different to the hyperparameter values reported in these papers. Using the code of [16], it is possible to reproduce their experiments and those of [31] for the inductive setting only. Our code allows to reproduce the experiments under the transductive setting as well.

4.3 State-of-the-art comparison

Tables 2 and 3 show the comparison to the state-of-the-art. For inductive ZSL setting, our model performs globally better than all other methods with the highest mNRG score. It also achieves the best score on CUB and SUN. In the transductive ZSL setting, our approach obtains a mNRG score of 22.8, establishing a new transductive ZSL state-of-the-art on CUB, SUN and AWA2. The comparison to [6] is particular since they report results on three of

Table 2 State-of-the-art comparison accuracy for ZSL on the "proposed split" of [30], both inductive (IN) and transductive (TR) results are shown

		CUB T1	FLO T1	SUN T1	AWA2 T1	mNRG
Zero-shot Learning						
IN	CLSWGAN [29]	57.3	67.2	60.8	68.2	0 [ref]
	LisGAN [14]	58.8	69.6	61.7	70.6	2.0
	f-VAEGAN-D2 (*) [31]	61.0	67.7	64.7	71.1	3.3
	TF-VAEGAN (*) [16]	63.2	**70.4**	64.3	**73.2**	**4.2**
	Ours	**70.7**	69.2	**64.7**	71.9	3.8
TR	ALE-trans [30]	54.5	48.3	55.7	70.7	-4.0
	GFZSL [27]	50.0	85.4	64.0	78.6	6.8
	DSRL [32]	48.7	57.7	56.8	72.8	-6.3
	f-VAEGAN-D2 (*) [31]	74.2	89.1	70.1	89.8	19.3
	TF-VAEGAN (*) [16]	77.2	**92.6**	70.1	92.1	21.9
	Ours	**80.6**	89.3	**71.7**	**92.8**	**22.7**
ZSL with fine-tuned features						
FT-IN	f-VAEGAN-D2 (⋆) [31]	74.1	70.5	**64.5**	69.9	0 [ref]
	TF-VAEGAN (⋆) [16]	72.5	70.6	64.1	68.5	-0.9
	Ours	**83.3**	**72.8**	64.0	**70.4**	**1.4**
FT-TR	f-VAEGAN-D2 (⋆) [31]	82.1	95.6	68.5	89.9	14.0
	TF-VAEGAN (⋆) [16]	85.1	**96.0**	**73.8**	**93.0**	**17.1**
	Ours	**86.1**	95.8	70.0	91.1	16.6

Avg is the average score over the four datasets. Models marked with * were partially re-implemented

the considered datasets only. Without fine-tuning, their results is far above other methods on AWA2 but also far below on CUB and SUN.

Unsurprinsingly, in the GZSL setting, feature generating approaches obtain better results than others. We also note that the accuracy on unseen classes (u) and the one on seen classes (s) are better balanced. Our model outperforms the existing methods for both inductive and transductive GZSL settings. In particular, in the inductive GZSL setting, our model obtains 67.2% on CUB, significantly improving those obtained previously (56.9%). By reducing the bias towards seen classes, we globally achieve better performance on unseen classes. However, the scores on seen classes may slightly decrease, in particular in inductive setting. It is nevertheless compensated by the gain on the unseen classes.

We also conducted some experiments with fine-tuned features, with the same features as those used in [16, 31]. To compute the global score mNRG for this experiment, we used f-VAEGAN-D2 in transductive setting as a baseline. In the ZSL setting, the results of TF-VAEGAN are globally better than ours in the transductive setting, both being significantly above f-VAEGAN. However, in the inductive setting, the results of TF-VAEGAN are below the baseline while ours are still slightly above.

In the GZSL settings, TF-VAEGAN still has a lower mNRG score than the baseline f-VAEGAN-D2 in the inductive setting and quite comparable score in the transductive one.

Table 3 State-of-the-art comparison accuracy for GZSL on the "proposed split" of [30], both inductive (IN) and transductive (TR) results are shown

		CUB			FLO			SUN			AWA2			
		u	s	H	u	s	H	u	s	H	u	s	H	*mNRG*
Generalized Zero-shot Learning														
IN	`CLSWGAN` [29]	43.7	57.7	49.7	59.0	73.8	65.6	42.6	36.6	39.4	57.9	61.4	59.6	0 [ref]
	`LisGAN` [14]	46.5	57.9	51.6	57.7	**83.8**	68.3	42.9	37.8	40.2	52.6	76.3	62.3	2.3 (1.9)
	`f-VAEGAN-D2` (*) [31]	48.5	60.2	53.7	56.8	73.9	64.2	45.1	38.0	41.3	**57.6**	70.6	63.5	2.9 (3.9)
	`TF-VAEGAN` (*) [16]	52.2	62.7	56.9	62.4	83.5	**71.4**	41.3	**39.2**	40.2	52.8	**81.9**	64.2	5.2 (4.6)
	[6]	41.4	49.7	45.2	–	–	–	29.9	40.2	34.3	65.1	78.9	71.3	xx (-4.5)
	Ours	**60.3**	**75.9**	**67.2**	**62.6**	81.5	70.8	**45.2**	39.0	**41.8**	**57.6**	76.4	**65.6**	**5.6** (6)
TR	`ALE-trans` [30]	23.5	45.1	30.9	13.6	61.4	22.2	19.9	22.6	21.2	12.6	73.0	21.5	-28.5
	`GFZSL` [27]	24.9	45.8	32.2	21.8	75.0	33.8	0.0	41.6	0.0	31.7	67.2	43.1	-24.7
	`DSRL` [32]	17.3	39.0	24.0	26.9	64.3	37.9	17.7	25.0	20.7	20.8	74.7	32.6	-26.4
	`f-VAEGAN-D2` (*) [31]	65.6	68.1	66.8	78.7	87.2	82.7	60.6	41.9	49.6	84.8	88.6	86.7	17.1
	`TF-VAEGAN` (*) [16]	69.1	**75.1**	72.0	83.8	91.9	87.6	**62.5**	**46.8**	**53.5**	84.5	**90.2**	87.2	22.2
	Ours	**74.2**	70.5	**72.3**	**85.1**	**92.2**	**88.5**	59.5	46.0	51.8	**86.7**	88.5	**87.6**	**22.8**
GZSL with fine-tuned features														
FT-IN	`f-VAEGAN-D2` [31]	70.4	79.2	74.5	59.0	**93.8**	72.4	**50.6**	37.4	**43.0**	60.1	79.3	68.3	0 [ref]
	`TF-VAEGAN` [16]	64.1	79.7	71.0	**64.6**	90.3	**75.3**	38.2	**43.8**	40.8	57.8	**87.3**	**69.5**	-0.5 (-2.2)
	[6]	69.2	76.4	72.6	-	-	-	50.5	43.1	46.5	69.0	86.5	76.8	xx (3.5)
	Ours	**65.6**	**82.7**	**73.2**	62.0	89.4	73.2	43.9	41.8	42.8	**60.7**	80.1	69.0	**0.25** (-0.2)
FT-TR	`f-VAEGAN-D2` (*) [31]	77.1	82.0	79.5	92.7	97.4	94.9	**62.0**	38.6	47.6	83.9	**95.5**	**89.3**	12.3
	`TF-VAEGAN` (*) [16]	80.1	**80.0**	80.0	82.7	94.3	88.2	60.5	43.6	**50.7**	79.7	89.6	84.4	11.8
	Ours	**81.2**	82.6	**81.9**	**93.0**	**97.7**	**95.3**	47.5	**48.6**	48.0	**85.6**	91.3	88.3	**13.7**

We measure Top-1 accuracy on seen (s) and unseen (u) classes as well as their harmonic mean (H). $\tilde{H}$ is the average over the four datasets. Models marked with * were partially re-implemented. The mNRG in parentheses is computed without the score on FLO

Our approach obtains performances in line with f-VAEGAN in the inductive but significantly outperforms the two other approaches in the transductive setting when compared over the four datasets.

4.4 Ablation study

4.4.1 Influence of the mixing proportion

In this section, we perform an ablation study on four ZSL datasets. We evaluate our model with different values of the random mixing proportion λ (Results are shown in Table 4). Note that when λ is sampled from a distribution, a new value is selected for each minibatch.

We found that the best performances accross all datasets were met when $\lambda = 0.5$ or $\lambda \sim \mathcal{N}(0.5, 0.25)$, i.e. setting equal weights for the two terms of the convex combination. Other settings for λ, such as $\lambda = 0.2$ or $\lambda \sim Uniform(0, 1)$, deteriorates the performances ($\sim$ 1% worse for CUB, SUN and FLO and $\sim$ 4% worse for AWA2). Even poorer performances were found when setting $\lambda \sim Beta(0.3, 0.3)$ ($\sim$ 1% worse for CUB and SUN and $\sim$ 8% worse for FLO and AWA2).

For image classification [33], the random mixing proportion is sampled from the Beta distribution with a small value of α, as it assumes that the examples in the neighborhood of each data sample share the same class. Indeed, given a small $\alpha = 0.3$, beta distribution samples more values closer to either 0 and 1, making the mixing result closer to either one of the two examples. However, in our method, we construct *ambiguous* semantic prototypes with the corresponding *ambiguous* classes being completely distinct from the real ones. Therefore, sampling λ from a Beta distribution, with $\alpha < 1$, is not a reasonable choice.

4.4.2 Influence of the subset to learn virtual prototypes

According to the nominal protocol of the proposed method, new “frontier prototypes” are learned by combining prototypes of both seen and unseen categories in the transductive setting. The usage of seen prototypes in (2) allows to regularize the conditional latent space, such that further used unseen prototypes result in more discriminative features. If unseen prototypes are used, the space is better regularized in their neighborhood. It is particularly interesting if some unseen prototypes are not contained in the convex envelop of the seen classes. Let nevertheless note that we conducted experiments with negative λ without getting noticeable improvements. To evaluate the respective contribution of seen and unseen prototypes in our model, we compared the performances on CUB, while using different subsets of prototypes in (2). For fair comparison, the results reported in Table 5 were all obtained with f-VAEGAN using unlabeled images at training time, such that the ‘s+u’ results are the same as the transductive setting in Tables 2 and 3.

In the ZSL setting, we obtain the same results whether we use all prototypes or unseen prototypes only, this makes sense since the test images are from the unseen classes only and there is no point in modelling the ambiguities with seen classes. It is also interesting to note that f-VAEGAN-D2 has a score of 74.2 in the transductive setting, while one can have have a score of 79.1 with the regularization learned with seen prototypes only. It shows that most of the improvement is due to the global regularization of the conditional latent space, rather than to a local one in the neighborhood of the prototypes used at test time.

In the GZSL setting, the results are better when the regularization is learned with unseen prototypes only rather than seen ones, but the usage of both is still above. Looking at the results on the seen and unseen classes specifically, one can see that the results are obviously

Table 4 Transductive ZSL and GZSL results with different values of the λ hyperparameter

	Zero-shot Learning				Generalized Zero-shot Learning											
	CUB	FLO	SUN	AWA2	CUB			FLO			SUN			AWA2		
	T1	T1	T1	T1	u	s	H	u	s	H	u	s	H	u	s	H
$\lambda \sim$ Beta(0.3, 0.3)	**80.9**	72.1	71.3	87.7	74.4	70.0	72.1	67.3	89.7	76.9	60.6	44.9	51.6	79.6	86.3	82.8
$\lambda \sim$ Uniform(0, 1)	80.8	86.3	**72.2**	87.0	75.0	69.0	71.9	83.1	91.1	86.9	60.1	45.5	51.8	79.1	87.5	83.1
$\lambda \sim \mathcal{N}(0.5, 0.25)$	80.7	**89.7**	71.4	89.2	74.3	69.8	72.0	86.3	91.0	**88.5**	60.6	45.6	**52.1**	85.9	88.7	87.2
$\lambda = 0.5$	80.6	89.3	71.7	**92.8**	74.2	70.5	**72.3**	85.1	92.2	**88.5**	59.5	46.0	51.8	86.7	88.5	**87.6**
$\lambda = 0.2$	**80.9**	86.9	71.7	88.6	74.2	69.8	71.9	83.3	92.1	87.5	59.3	45.3	51.3	81.2	83.6	82.4

Table 5 Performances on CUB according to the prototype training subset used to learn ambiguous prototypes (*s* =seen, *u*=unseen)

	ZSL	GZSL		
	T1	u	s	H
s+u	80.6	**74.2**	70.5	**72.3**
s	79.1	69.4	**70.9**	70.1
u	**80.7**	72.6	70.0	71.2

For fair comparison, unlabeled images are used at training time in each case

better for the classes that are regularized with (3). The comparison to the results obtained by f-VAEGAN-D2 without the regularization in Table 3 (u=65.6 s=68.1 H=66.8) shows that the regularization is beneficial in any case.

4.5 Integration to TF-VAEGAN

Our contribution is generic and can be used in other conditional generative-based ZSL architectures. Therefore, we evaluate the generalization capabilities of our proposed method, by integrating our contribution in the TF-VAEGAN [16] framework.

We first learn the model end-to-end, adding the proposed regularization. Table 6 shows the comparison on CUB, between the original TF-VAEGAN model and the one learned with the proposed regularization (3). Our contribution improves the performance of the vanilla TF-VAEGAN for both ZSL and GZSL tasks, either in inductive or transductive settings, by 1.5 to 3 points. Interestingly, in GZSL, one can note that the improvement is mainly due to an increase of the scores on unseen classes, while the ones on seen classes is almost similar to the vanilla TF-VAEGAN. It thus tends to show that our approach reduces the bias towards seen classes in the generalized context.

We conducted an additional experiment consisting in fine-tuning the generator learned by TF-VAEGAN with our method. To prevent the generator from losing the previously learned information, which is the marginal feature distribution, the discriminators D_1 and D_2 are trained from scratch. We again observe an improvement of the performance for ZSL and GZSL, both in inductive and transductive settings. The scores are nevertheless intermediate

Table 6 Comparison between the vanilla `TF-VAEGAN` and that augmented with our loss (3), on CUB dataset for both inductive and transductive ZSL/GZSL settings, either when the model is learned from scratch or fine-tuned (ft)

		ZSL	GZSL		
		T1	u	s	H
IN	Vanilla	63.2	52.2	62.7	56.9
	+ ours (scratch)	64.9	57.4	62.6	59.8
	+ ours (ft)	64.3	54.1	62.7	58.0
TR	Vanilla	77.2	69.1	75.1	72.0
	+ ours (scratch)	79.0	72.3	75.3	73.7
	+ ours (ft)	78.5	71.6	74.9	73.2

between those obtained by the original model and those obtained previously by learning from scratch. In both cases, the GZSL experiments show that most of the score improvement is due to a better recognition of the unseen classes, while the performances on the seen classes are similar to (or slightly below) the original model.

5 Conclusion

We propose a novel approach to train a conditional generative-based model for zero-shot learning. The approach improves the discriminative capacity of the synthesized features by training the generator to recognize virtual *ambiguous* classes. We construct the corresponding *ambiguous* class prototypes as convex combinations of the *real* class prototypes and then we train the generator to recognize these virtual classes. This simple procedure allows the generator to learn the transitions between categories and thus, to better distinguish them. Our approach can be integrated to any conditional generative model. Experiments on four benchmark datasets show the effectiveness of our approach across zero-shot and generalized zero-shot learning. In most cases, the improvement is due to a better recognition of unseen classes, while the score on seen classes are maintained, which means that our approach reduces the bias towards seen classes in GZSL. However, most of the time, the score on seen classes remains higher than the one on unseen classes, showing the bias still remains to some extent.

The method is limited to create ambiguous classes from a *couple* of real classes by a linear interpolation. To push further our approach, one could explore non-linear interpolation for constructing ambiguous classes, or considering more than two *real* classes to construct an *ambiguous* one. Note that the experiment we conducted on (linear) extrapolation did not bring interesting results. Beyond this contribution to zero-shot learning, our approach can also be beneficial to other tasks that aims at relating ambiguous visual and semantic information such as multimodal entity linking and retrieval, cross-modal retrieval or classification and more generally those in which a latent space is used for learning data features.

Funding this work relied on the use of the FactoryIA cluster, financially supported by the Ile-de-France Regional Council. HLB is partially funded by CPS4EU project funded from the H2020-ECSEL-2018-IA call – Grant Agreement: 826276 and the ANR-19-CE23-0028 MEERQAT project. CH was funded by CEA for her master internship.

Data Availability The datasets analysed during the current study are available from the corresponding author on reasonable request. The code, features and finetuned features are available online.

Declarations

Conflict of Interests The authors declare that they have no conflict of interest.

References

1. Adjali O, Besancon R, Ferret O et al (2020) Multimodal entity linking for tweets. In: European conference on information retrieval, Lisbon, Portugal
2. Arjovsky M, Chintala S, Bottou L (2017) Wasserstein generative adversarial networks. In: Proceedings of the 34th international conference on machine learning. JMLR.org, ICML'17, vol 70, pp 214–223
3. Arora G, Verma VK, Mishra A et al (2017) Generalized zero-shot learning via synthesized examples. CoRR: 1712.03878
4. Bucher M, Herbin S, Jurie F (2017) Generating Visual Representations for Zero-Shot Classification. In: International conference on computer vision (ICCV) workshops: TASK-CV: transferring and adapting source knowledge in computer vision, Venise, Italy
5. Chami I, Tamaazousti Y, Le Borgne H (2017) Amecon: abstract meta-concept features for text-illustration. In: ACM international conference on multimedia retrieval. ICMR, Bucharest
6. Chou YY, Lin HT (2021) Adaptive and generative zero-shot learning. In: International conference on learning representations
7. Frome A, Corrado GS, Shlens J et al (2013) Devise: a deep visual-semantic embedding model. In: Advances in neural information processing systems, pp 2121–2129
8. Lampert CH, Nickisch H, Harmeling S (2009) Learning to detect unseen object classes by between-class attribute transfer. In: Computer vision and pattern recognition. IEEE, pp 951–958
9. Le Cacheux Y, Le Borgne H (2019a) From classical to generalized zero-shot learning: a simple adaptation process. In: International conference on multimedia modeling. Springer, pp 465–477
10. Le Cacheux Y, Le Borgne H, Crucianu M (2019b) Modeling inter and intra-class relations in the triplet loss for zero-shot learning. In: International conference on computer vision, pp 10,333–10,342
11. Le Cacheux Y, Le Borgne H, Crucianu M (2020a) Using sentences as semantic embeddings for large scale zero-shot learning. In: ECCV 2020 workshop: transferring and adapting source knowledge in computer vision, Springer
12. Le Cacheux Y, Le Borgne H, Crucianu M (2021) Zero-shot Learning with Deep Neural Networks for Object Recognition. In: Benois Pineau J, Zemmari A (eds) Multi-faceted deep learning. Springer, chap 6, pp 273–288
13. Le Cacheux Y, Popescu A, Le Borgne H (2020b) Webly supervised semantic embeddings for large scale zero-shot learning. In: Asian conference on computer vision
14. Li J, Jing M, Lu K et al (2019) Leveraging the invariant side of generative zero-shot learning. In: IEEE computer vision and pattern recognition (CVPR)
15. Myoupo D, Popescu A, Le Borgne H et al (2010) Multimodal image retrieval over a large database. In: Peters C, Caputo B, Gonzalo J et al (eds) Proceedings of the 10th international conference on Cross-language evaluation forum: multimedia experiments, Lecture notes in computer science. Springer Berlin / Heidelberg, pp 177–184, Berlin
16. Narayan S, Gupta A, Khan FS et al (2020) Latent embedding feedback and discriminative features for zero-shot classification. In: ECCV
17. Nilsback ME, Zisserman A (2008) Automated flower classification over a large number of classes. In: Indian conference on computer vision graphics and image processing
18. Patterson G, Hays J (2012) Sun attribute database: discovering, annotating and recognizing scene attributes. In: Computer vision and pattern recognition
19. Paul A, Krishnan NC, Munjal P (2019) Semantically aligned bias reducing zero shot learning. In: Proceedings of the ieee conference on computer vision and pattern recognition, pp 7056–7065
20. Reed S, Akata Z, Schiele B et al (2016) Learning deep representations of fine-grained visual descriptions. In: Computer vision and pattern recognition
21. Romera-Paredes B, Torr P (2015) An embarrassingly simple approach to zero-shot learning. In: International conference on machine learning
22. Schonfeld E, Ebrahimi S, Sinha S et al (2019) Generalized zero-shot learning via aligned variational autoencoders. In: Computer vision and pattern recognition (CVPR) workshops
23. Tamaazousti Y, Le Borgne H, Hudelot C et al (2020) Learning more universal representations for transfer-learning. IEEE T Pattern Anal Mach Intell 42(9):2212–2224
24. Tran TQN, Le Borgne H, Crucianu M (2015) Combining generic and specific information for cross-modal retrieval. In: In: Proc. ACM international conference on multimedia retrieval (ICMR)
25. Tran TQN, Le Borgne H, Crucianu M (2016a) Aggregating image and text quantized correlated components. In: Computer vision and pattern recognition, Las Vegas

26. Tran TQN, Le Borgne H, Crucianu M (2016b) Cross-modal classification by completing unimodal representations. In: ACM multimedia 2016 workshop: vision and language integration meets multimedia fusion, Amsterdam, The Netherlands
27. Verma VK, Rai P (2017) A simple exponential family framework for zero-shot learning. In: Machine learning and knowledge discovery in databases
28. Wah C, Branson S, Welinder P et al (2011) The Caltech-UCSD Birds-200-2011 Dataset. Tech Rep, CNS-TR-2011-001, California Institute of Technology
29. Xian Y, Lorenz T, Schiele B et al (2018) Feature generating networks for zero-shot learning. In: 2018 IEEE conference on computer vision and pattern recognition, computer vision foundation / IEEE computer society. CVPR 2018, Salt Lake City, pp 5542–5551
30. Xian Y, Schiele B, Akata Z (2017) Zero-shot learning - the good the bad and the ugly. In: Computer vision and pattern recognition
31. Xian Y, Sharma S, Schiele B et al (2019) F-vaegan-d2: a feature generating framework for any-shot learning. In: Computer vision and pattern recognition, pp 10,267–10,276
32. Ye M, Guo Y (2017) Zero-shot classification with discriminative semantic representation learning. In: Computer vision and pattern recognition, pp 5103–5111
33. Zhang H, Cissė M, Dauphin YN et al (2018) Mixup: beyond empirical risk minimization. In: International conference on learning representations
34. Znaidia A, Shabou A, Popescu A et al (2012) Multimodal feature generation framework for semantic image classification. In: ACM international conference on multimedia retrieval (ICMR 2012)

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.