Learning to Identify Seen, Unseen and Unknown in the Open World: A Practical Setting for Zero-Shot Learning

Sethupathy Parameswaran[1], Yuan Fang[1], Chandan Gautam[2], Savitha Ramasamy[2], Xiaoli Li[2,3]
[1]Singapore Management University [2]Institute for Infocomm Research, A*STAR
[3]A*STAR Centre for Frontier AI Research
{sethupathyp, yfang}@smu.edu.sg, {gautamc, ramasamysa, xlli}@i2r.a-star.edu.sg

Abstract

As vision-language models advance, addressing the Zero-Shot Learning (ZSL) problem in the open world becomes increasingly crucial. Specifically, a robust model must handle three types of samples during inference: seen *classes with visual and semantic information provided in training,* unseen *classes with only the semantic information in training, and* unknown *samples with no prior information from training. Existing methods either handle seen and unseen classes together (ZSL) or seen and unknown classes (known as Open-Set Recognition, OSR). However, none addresses the simultaneous handling of all three, which we term Open-Set Zero-Shot Learning (OZSL). To address this problem, we propose a two-stage approach for OZSL that recognizes seen, unseen, and unknown samples. The first stage classifies samples as either seen or not, while the second stage distinguishes unseen from unknown. Furthermore, we introduce a cross-stage knowledge transfer mechanism that leverages semantic relationships between seen and unseen classes to enhance learning in the second stage. Extensive experiments demonstrate the efficacy of the proposed approach compared to naïvely combining existing ZSL and OSR methods. The code is available at* https://github.com/smufang/OZSL.

1. Introduction

Deep learning has shown great promise in various computer vision tasks. However, they require a large amount of labeled data for training, which can be time consuming or expensive to acquire. On the other hand, humans can easily identify a novel unseen object only based on a textual description of that object, by leveraging their visual experience of related objects in the past. For example, if one has seen a motorcycle, they will be able to identify a bicycle based on a simple textual description: "It looks similar to a motorcycle but is lighter and has no engine." As shown in Fig. 1(b), this kind of learning ability is called zero-shot learning (ZSL) [14,22,45], which aims to identify objects in both *seen* and *unseen* classes. Specifically, at training time, both visual features and semantic (text or attribute-based) descriptions are available for seen classes, while only semantic information is given for unseen classes without any visual input.

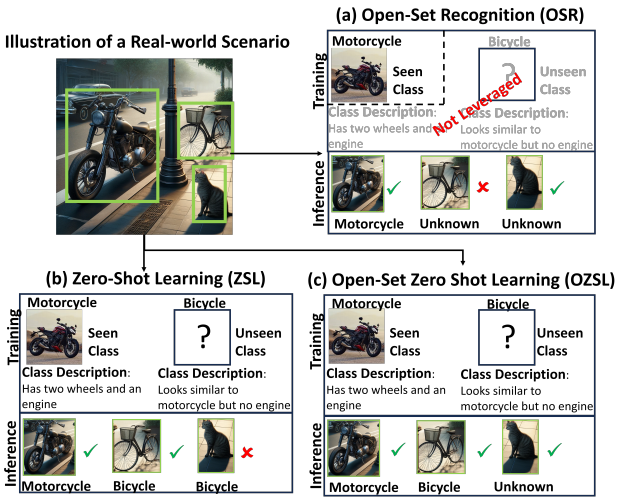


Figure 1. Illustration[1] of different classification settings. (a) OSR methods can flag unknown samples but cannot recognize unseen class samples, where class descriptions for unseen classes are not utilized. (b) ZSL methods can identify unseen class samples but cannot flag unknown samples. (c) Our OZSL approach aims to identify seen and unseen class samples, and at the same time flag unknown samples.

As there is no guarantee that the model in a real-world deployment will only encounter classes that it has been trained for or given semantic descriptions, despite recent advances in ZSL methods [10, 52], an important practical problem still remains: How to flag samples from an *unknown* class along with classifying samples from seen and

[1]See Appendix E for the source of the image in the illustration.

unseen classes in an open-world scenario? Specifically, we do not know anything about unknown classes during training: Neither visual nor semantic features are given. At inference time, samples from an unknown class can be considered as out-of-distribution (OOD) samples, as they are encountered by the trained model unintentionally during inference in open-world applications such as autonomous driving [18, 55]. For example, as illustrated in Fig. 1, if a model is trained on motorcycles using labeled images, it can be generalized to identify a related class such as bicycle through ZSL with only the class description (assuming images for this class is difficult to obtain). However, in the open world, it can encounter a stray cat or virtually any other class, which the model must flag as unknown for conservative handling instead of forcing it to be a predefined seen or unseen class. It is interesting to note that intuitively, *unseen classes* refer to those "we know we have not seen", which can be predefined by providing class descriptions. In contrast, *unknown classes* represent what "we don't know that we haven't seen", which can encompass anything (i.e., open-set). Attempting to define an exhaustive list of all possible classes would be infeasible and critically flawed. Hence, it is crucial for the model to learn to flag samples it does not recognize as *unknown* in open-set scenarios.

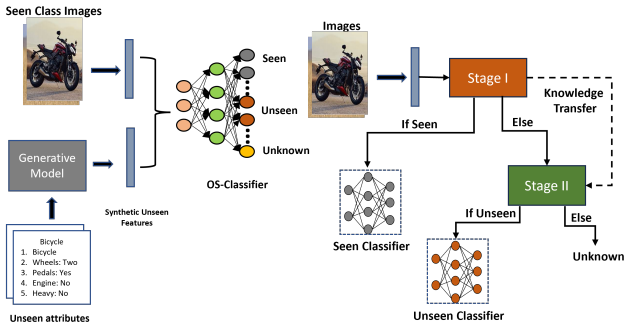
On one hand, existing ZSL methods misclassify unknown samples as one of the seen/unseen classes during inference. On the other hand, open-set recognition (OSR) [9, 38] is designed to recognize unknown samples using only the in-distribution (ID) data during training. However, as illustrated in Fig. 1(a), these OSR methods are not capable of handling unseen classes, and misclassify them as unknown. Therefore, in this paper, we aim to address a novel problem of *Open-set Zero-Shot Learning* (OZSL), which calls for the capability of both ZSL and OSR. To be more specific, our goal is to recognize seen, unseen, and unknown classes under a single framework as shown in Fig. 1(c). The key difference between existing OSR methods and the proposed OZSL setting is we have visual samples only for the seen classes among the ID classes, while only having semantic information for the remaining ID but unseen classes.

One natural idea is to detect all three types of seen, unseen and unknown classes simultaneously in one go. For instance, we may use a simple threshold on the ZSL predictions to determine whether the sample is unknown, thereby reducing the problem to the standard ZSL setting. Alternatively, we can employ a generative model to produce synthetic visual samples for the unseen classes based on their semantic information, reducing the problem to the standard OSR setting. However, trying to classify all three types in one go can be less robust. On one hand, both unseen and unknown classes lack visual features in training, making their separation difficult. On the other hand, seen and unseen classes often present significant overlap in their semantic space [11,32] (e.g., motorcycle and bicycle) in order to align the visual and semantic spaces. These two sources of misclassification can be difficult to be dealt with simultaneously. Hence, trying to classify all three types in one go tends to reduce the overall performance.

To overcome this challenge, we propose a two-stage approach: In Stage I, we train a model to separate seen classes from the rest, namely, unseen and unknown, while classifying a seen sample to its respective seen class; in Stage II, we train a model to separate unseen and unknown classes, while classifying an unseen sample to its respective unseen class. In particular, in Stage II, we can employ synthetic unseen class data as a substitute for the missing visual features, to mimic the seen classes in Stage I. Hence, the two stages share a similar goal: Both aim to separate samples with visual features (real visual input in Stage I or synthetic features in Stage II) and those without. While such a two-stage approach is intuitive, the two stages are decoupled, despite their similar goal.

Hence, a second challenge lies in how to enable knowledge transfer from Stage I to Stage II, to leverage the inherent semantic relatedness between seen and unseen samples, which can help improve the overall performance. That is, since semantic information of seen and unseen classes are related, we leverage the seen classes learned in Stage I to help the representation learning of unseen classes in Stage II. Furthermore, the transfer of knowledge from Stage I to Stage II is advantageous as Stage I is trained by using real visual features from seen classes, while Stage II is trained by using synthetic visual features, which tends to be less robust than Stage I. To enable the cross-stage knowledge transfer, we propose two strategies: (i) *Weight initialization* strategy, where we initialize the Stage II model with the trained weights of the Stage I model; (ii) *Distribution retainment* strategy, where we propose a distribution retainment loss to ensure the seen class distribution learned in Stage I is maintained in Stage II.

We summarize the contributions of this work. (i) To the best of our knowledge, this is the first work introducing a novel problem setting called open-set zero-shot learning (OZSL), a more practical setup for many applications in the open world. (ii) We demonstrate that naïvely combining ZSL with OSR methods does not lead to good performance. Subsequently, we propose a two-stage method for the OZSL problem, introducing cross-stage knowledge transfer through the novel strategies of weight initialization and distribution retainment to improve the OZSL performance. (iii) We conduct extensive experiments to demonstrate the viability of our approach, which outperforms a series of state-of-the-art ZSL and OSR methods as well as their combinations.



(a) One-stage approach (b) Two-stage approach

Figure 2. Illustration of one-stage and two-stage approaches for open-set zero-shot learning.

2. Related Work

Our work is mainly related to open-set recognition and zero-shot learning, which are briefly reviewed here.

Open-Set Recognition (OSR). OSR addresses the problem of separating known samples, which the model is trained to recognize, from unknown samples which the model encounters unintentionally during inference. OSR is essential for making a model robust in safety-critical applications. In literature, many solutions have been proposed to address the OSR problem. Hendrycks, Dan, and Kevin Gimpel, *et al*. [23] have proposed a baseline method based on the predicted maximum softmax probability (MSP). OpenMAX [6] uses a Weibull distribution to distinguish known samples from unknowns, while Lee, Kimin, *et al*. [28] use the Mahalanobis distance as a measure to distinguish them. Recently, Wang, Haoqi, *et al*. [51] have introduced virtual logit matching which combines information from the feature space and logits space to better identify known samples from unknown samples. Beyond these non-generative models, Lee, Kimin, *et al*. [29] have proposed a generative OSR method using generative adversarial networks, which generates synthetic unknown samples to train the model for distinguishing the known and unknown samples.

Zero-Shot Learning (ZSL). ZSL addresses the problem of identifying unseen classes using only seen class samples and unseen class semantic information, which can be broadly classified into three categories.

Non-generative Methods: Non-Generative methods [4, 7, 24, 45, 58] generally perform ZSL in one of the following three ways. (i) Visual to semantic mapping [1, 19, 21, 27, 47]: Visual data are projected into lower-dimensional space. As it shrinks the variance of the data, it might lead to the hubness problem [16] . (ii) Semantic to visual mapping [22, 33, 46, 61]: It resolves the hubness issue and relies heavily on the attribute features, which are mapped into the visual feature space to be used as class prototypes. (iii) Map visual and semantic features into a common subspace [3, 8, 20, 30, 42]: It benefits from both of the above approaches and learns a common embedding for ZSL.

Generative Methods: Another common approach to the ZSL problem involves generating synthetic samples for unseen classes using a generative model. Most existing generative approaches fall into one of the following categories: variational autoencoders [12, 36, 43, 49], generative adversarial networks [17, 31, 56], and normalization flow [13, 44]. All these methods are conditioned on unseen class attributes to generate synthetic samples for unseen classes. Then, by substituting the synthetic samples as the training data for unseen classes, the ZSL problem can be reduced to a standard classification problem.

Domain Separation Methods: These methods [5, 11, 25, 35] first separate seen classes from unseen classes and perform separate classifications on the two categories (seen and unseen). This avoids overlapping decision boundaries between the seen classes and unseen classes. Note that some of these works appear to address OSR but merely treat the unseen classes as an open set, thereby solving a traditional ZSL problem. In contrast, this work considers seen, unseen, and unknown as separate categories while addressing our proposed OZSL setting.

Other Fields: ZSL has also been explored in other fields such as zero-shot text classification [34, 39] in natural language processing (NLP). Moreover, using pre-trained language models for synthetically generating the entire dataset [59, 60] has been gaining interest in the research community. Similarly, embedding based non-generative approaches for zero-shot node classification [53, 54] has been explored in graph machine learning.

3. Problem Definition and Naïve Approaches

In this section, we first introduce our novel problem setting, and further develop naïve one-stage approaches based on state-of-the-art ZSL and/or OSR methods.

3.1. Problem definition

In this work, we address a realistic novel problem called Open-Set Zero-Shot Learning (OZSL) as shown in Fig. 1(c). We are given a set of training samples from seen classes $D^{\text{tr}} = \{\mathbf{x}, y, \mathbf{a} | \mathbf{x} \in \mathbf{X}_s, y \in Y_s, \mathbf{a} \in \mathbf{A}_s\}$, where $\mathbf{x}$ is the visual feature of the sample extracted by a pre-trained backbone, y is the class of that sample, and $\mathbf{a}$ is the corresponding class attribute vector. $Y_s = \{y_1^s, y_2^s, ..., y_m^s\}$ denotes the set of all m seen classes. Similarly, $\mathbf{A}_s = \{\mathbf{a}_1^s, \mathbf{a}_2^s, ..., \mathbf{a}_m^s\}$ denotes the corresponding seen class attributes. In addition to the seen class samples, the training data also consist of *unseen* class attributes $\mathbf{A}_u = \{\mathbf{a}_1^u, \mathbf{a}_2^u, ..., \mathbf{a}_n^u\}$ corresponding to the n unseen classes $Y_u = \{y_1^u, y_2^u, ..., y_n^u\}$. During testing, a test sample $\mathbf{x}^{\text{ts}}$ may come from the seen classes Y_s, the unseen classes Y_u, or neither (i.e., *unknown*). The objective of the trained

model is to correctly classify the test sample $\mathbf{x}^{\text{ts}}$ into its corresponding class $y \in Y_s \cup Y_u$ if it is in-distribution or reject the sample if it is unknown.

3.2. Naïve One-Stage Approaches

One natural way to address the OZSL problem is to categorize the samples into the corresponding categories, namely, seen, unseen, and unknown in one go. A major challenge of OZSL is the absence of training data for unseen classes. However, most OSR detection methods require in-distribution data. In order to address this problem, we can use an off-the-shelf generative method such as the state-of-the-art GSMFlow [13] to generate synthetic samples for the unseen classes as shown in Fig. 2(a). The generated synthetic data, along with the seen data, are taken as the in-distribution data, reducing the OZSL problem to a standard OSR problem, which can be addressed by popular OSR methods such as MSP [23], ViM [51] and KNN [48]. In our evaluation, we have also utilized these methods as naïve baselines to benchmark our proposed approach on the novel OZSL problem.

4. Proposed Method

As one-stage approaches attempt to handle all three types of samples in one go, the performance can be suboptimal. A potential reason is due to the semantic overlap between seen and unseen classes, as well as the lack of visual features for both unseen and unknown classes. Hence, we propose a two-stage approach to handle the two sources of misclassification, as shown in Fig. 2(b). In Stage I, we aim to separate seen classes and the rest (including unseen and unknown); in Stage II, we aim to handle the unseen and unknown. Meanwhile, to leverage the semantic relatedness between seen and unseen classes, we propose a cross-stage knowledge transfer from Stage I to Stage II.

4.1. Overall Framework

Training. The training process of the proposed approach is outlined in Fig. 3, which is split into two stages.

In Stage I, the model is trained to identify whether an image is from the seen class or not using the training data D^{tr}. Additionally, we also train a classifier in the latent space to identify which seen class the sample belongs to.

In Stage II, the model is trained to identify whether an image is from the unseen class or not. However, it is worth noting that there is no training sample for unseen classes. Hence, to overcome this issue we use synthetic data produced by a generative method conditioned on unseen class attributes. Furthermore, we propose cross-stage knowledge transfer to leverage the knowledge learned from the first stage in the second stage of model training (see Sect. 4.3). Similar to the first stage, we also train a classifier to identify which unseen class a sample belongs to.

Inference. We first identify if the test sample belongs to a seen class or not using the first stage of our model. If the sample is from the seen class, the seen class classifier is used to determine the specific seen class for the sample. If the sample is identified as not belonging to the seen class, then we further identify if it belongs to an unseen class or not using the second stage of our model. If the sample is from an unseen class, the unseen classifier is used to determine the specific unseen class for the sample. If the sample is categorized as not belonging to any seen or unseen class, then it is rejected as unknown.

4.2. Latent Representation Alignment

Our work builds upon existing research that aligns latent representations between visual features and the respective class attribute vectors, in order to effectively utilize the additional class attributes. As shown in Fig. 3, both stages would require latent representation alignment. In the following, we introduce the general methodology for alignment, largely inspired by [11, 15, 57].

Specifically, we employ two variational auto-encoder (VAE) modules, one for the visual features, and the other for class attribute vectors. They are trained using the standard VAE losses, L^F_{VAE} for the visual VAE and L^A_{VAE} for the attribute VAE, respectively which are explained in detail in Appendix F.

To further enhance the alignment between visual features and attributes, we employ the following cross-reconstruction loss on the latent embeddings:

$$L_{\text{CR}} = \|\mathbf{x} - D^F(E^A(\mathbf{a}))\|_1 + \|\mathbf{a} - D^A(E^F(\mathbf{x}))\|_1. \quad (1)$$

Here, E^F and D^F denote the encoder and decoder of the visual VAE module. Similarly, E^A and D^A denote the encoder and decoder of the attribute VAE module.

Moreover, to classify the in-distribution samples to their respective classes, and to learn a more discriminative latent space, a classification loss is included, as follows.

$$L_{\text{cls}} = -\mathbb{E}_{\mathbf{x}}[p_{\mathbf{x}} \log q_{\mathbf{z}_\mathbf{x}}] - \mathbb{E}_{\mathbf{a}}[p_{\mathbf{a}} \log q_{\mathbf{z}_\mathbf{a}}], \quad (2)$$

where $p_{\mathbf{x}}$ and $p_{\mathbf{a}}$ are the one-hot ground-truth class vectors of $\mathbf{x}$ and $\mathbf{a}$, respectively; $q_{\mathbf{z}_\mathbf{x}}$ and $q_{\mathbf{z}_\mathbf{a}}$ are the class distributions predicted by the classifier for $\mathbf{z}_\mathbf{x}$ and $\mathbf{z}_\mathbf{a}$, respectively.

Hence, the overall alignment loss is

$$L_{\text{Align}} = L^F_{\text{VAE}} + L^A_{\text{VAE}} + \lambda_{\text{cr}} L_{\text{CR}} + \lambda_{\text{cls}} L_{\text{cls}}, \quad (3)$$

where λ_{cr} and λ_{cls} are hyperparameters.

4.3. Cross-stage Knowledge Transfer

In our framework, the two stages largely share a similar goal: Separating samples with visual features (real or

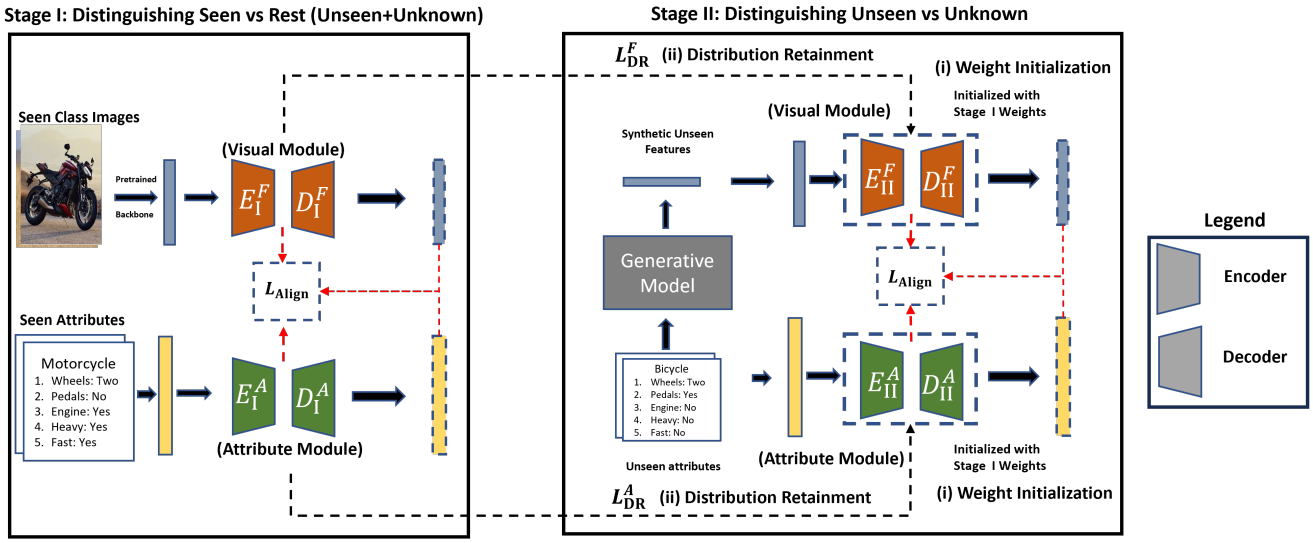


Figure 3. Illustration of the training process of our proposed method. Stage I is trained using the training data of seen classes to distinguish seen class samples from the rest. Stage II is trained using synthetic unseen class samples to distinguish unseen class samples from the unknowns. To leverage the semantic relatedness between seen and unseen classes, we propose two strategies for cross-stage knowledge transfer: (i) Weight Initialization and (ii) Distribution Retainment.

synthetic visual input in Stages I and II, respectively) and those without. Furthermore, seen classes in Stage I share semantic similarity with the unseen classes in Stage II. Hence, knowledge learned from Stage I can be potentially used to enhance Stage II which is trained using only the synthetic samples for unseen classes. In order to transfer the knowledge from Stage I to Stage II, we propose two strategies: (i) Weight Initialization; (ii) Distribution Retainment.

Weight Initialization. As shown in Fig.3, in order to facilitate knowledge transfer, the visual and attribute VAE modules of Stage II are initialized with the trained weights of their counterparts of Stage I. Specifically, let $\Theta^*(E_F^I, D_F^I, E_A^I, D_A^I)$ denote the trained weights of the Stage I VAE modules, consisting of the weights from both encoders and decoders for the visual features and attributes. Similarly, let $\Theta(E_F^{II}, D_F^{II}, E_A^{II}, D_A^{II})$ denote the weights of the Stage II VAE modules. Hence, at the beginning of Stage II, we initialize its weights as follows.

$$\Theta(E_F^{II}, D_F^{II}, E_A^{II}, D_A^{II}) \leftarrow \Theta^*(E_F^I, D_F^I, E_A^I, D_A^I). \quad (4)$$

Distribution Retainment. Furthermore, as shown in Fig. 3, we propose a distribution retainment loss L_{DR} in Stage II, to leverage the seen class distribution learned in Stage I to better learn unseen class distribution in Stage II because of their shared semantic similarity. First, for the visual VAE, we apply the following retainment loss.

$$L_{\text{DR}}^F = \|\phi^F - \phi^{F^*}\|_2^2, \quad (5)$$

where ϕ^F is the parameter of the posterior distribution given by the Stage II visual VAE module on a subset of randomly selected seen samples $\mathbf{x}$ from Stage I training data; $\phi_{\mathbf{x}}^{F^*}$ denotes the parameters of the posterior distribution given by the trained Stage I visual VAE module on the same set of seen samples. Specifically, this allows Stage II to learn new weights for unseen classes while also maintaining the similarity with that of the seen classes. In a similar spirit, we apply a retainment loss to the attribute VAE module on the same subset of seen samples $\mathbf{x}$ used in the visual counterpart, as follows.

$$L_{\text{DR}}^A = \|\phi^A - \phi^{A^*}\|_2^2. \quad (6)$$

Likewise, ϕ^A correspond to the parameter of the posterior distribution given by the Stage II attribute VAE module, and ϕ^{A^*} is given by the trained Stage I module. Thus, the overall distribution retainment loss can be written as

$$L_{\text{DR}} = L_{\text{DR}}^F + L_{\text{DR}}^A. \quad (7)$$

4.4. Training and Inference Processes

Training. First, the visual and attribute VAE modules of Stage I are trained using the seen training data, namely, visual seen features and the corresponding seen class attribute vectors, based on the alignment loss L_{Align} in Eq. (3).

Once Stage I is trained, the visual and attribute VAE modules of Stage II are initialized as in Eq. (6). Next, we employ a generator (see Appendix D for its specification), trained using the seen class visual samples and attributes, to further generate synthetic samples for the unseen classes based on the unseen class attributes. Subsequently, the initialized weights of the VAE modules are further updated by

Dataset	AWA1	CUB	FLO	SUN
# Images	30,475	11,788	8,189	14,340
Attribute length	85	1,024	1,024	102
Seen classes	40	150	82	645
Unseen classes	5	25	10	36

Table 1. Statistics of the datasets.

training on the synthetic samples and their corresponding class attributes. Note that the training of Stage II is regularized by our proposed distribution retainment, L_{DR}, in Eq. (7), based on the following overall loss function.

$$L^{\text{II}} = L_{\text{Align}} + \lambda_{\text{DR}} L_{\text{DR}}, \tag{8}$$

where λ_{DR} is the hyperparameter.

Inference. In Stage I, a test sample $\mathbf{x}^{\text{ts}}$ is first predicted with a binary outcome ω^{I}: $\mathbf{x}^{\text{ts}}$ is either from the `seen` classes or the `rest` (unseen or unknown), based on a threshold on the cosine similarity between the latent visual and attribute representations, as below.

$$\omega^{\text{I}} = \begin{cases} \texttt{seen}, & \max\limits_{\mathbf{a}\in\mathbf{A}^s} (\cos(\mathbf{z}_{\mathbf{x}^{\text{ts}}}, \mathbf{z}_{\mathbf{a}})) \geq \gamma^{\text{I}} \\ \texttt{rest}, & \text{otherwise} \end{cases}, \tag{9}$$

where γ^{I} is the threshold for Stage I. Next, if the test sample $\mathbf{x}^{\text{ts}}$ is categorized as `seen`, it will be further passed to the seen classifier in Stage I to predict the specific seen class, without continuing with Stage II. On the other hand, if $\mathbf{x}^{\text{ts}}$ is classified as `rest`, it proceeds to Stage II.

In Stage II, a test sample $\mathbf{x}^{\text{ts}}$ is again predicted with a binary outcome ω^{II}: It is either from the `unseen` classes or `unknown` (i.e., out-of-distribution), based on a threshold on the cosine similarity between the sample latent vector and unseen class attribute vector, as shown below.

$$\omega^{\text{II}} = \begin{cases} \texttt{unseen}, & \max\limits_{\mathbf{a}\in\mathbf{A}^u} (\cos(\mathbf{z}_{\mathbf{x}^{\text{ts}}}, \mathbf{z}_{a})) \geq \gamma^{\text{II}} \\ \texttt{unknown}, & \text{otherwise} \end{cases}, \tag{10}$$

where γ^{II} is the threshold of Stage II. If the test sample is categorized as `unseen`, it will be passed to the unseen classifier in Stage II to predict the specific unseen class. Otherwise, the sample is rejected as `unknown`.

5. Experiments

In this section, we conduct experiments to evaluate our proposed method in comparison to baseline approaches, along with further model analysis.

5.1. Experiment Setup

Datasets. For our experiments, we consider four commonly used datasets in Table 1, namely, *Animals with Attributes 1* (AWA1) [26], *Caltech-UCSD Birds-200-2011* (CUB) [2], *Oxford Flowers* (FLO) [37] and *SUN Attribute* (SUN) [40]. In particular, AWA1 is coarse-grained with a small number of high-level classes, while CUB, FLO and SUN have more fine-grained classes. Note that these datasets originally contain the seen and unseen splits for ZSL and we tweak their splits for our proposed OZSL setting. Specifically, for each dataset, we randomly choose half of the original unseen classes as unseen, and treat the other half as unknown (i.e., out-of-distribution) samples. The seen classes are maintained as provided in the original split. More detailed descriptions of the datasets are provided in the supplementary material (Appendix A).

Evaluation Metrics. OZSL results are reported by computing top-1 accuracy separately for the seen classes (SA), unseen classes (UA), and the unknown (UnkA). More importantly, to assess the trade-off among the three types and measure the overall performance, we compute a weighted harmonic mean (HM), as shown below. Note that we assign a higher weight to the seen category, as robust seen class performance is generally expected given that actual seen class data are provided for training.

$$\text{HM} = \frac{1}{0.5 \times \frac{1}{\text{SA}} + 0.25 \times \frac{1}{\text{UA}} + 0.25 \times \frac{1}{\text{UnkA}}}. \tag{11}$$

Settings and Implementation Details. Following prior work [56], we use ResNet-101, which is pre-trained on the ImageNet dataset, as the backbone to extract the visual features. The encoder and decoder components of the visual and attribute VAE modules are implemented using a multilayer perceptron. We use a linear LogSoftmax classifier in the latent space of the VAE modules. A detailed description of the hyperparameter settings and implementation can be found in the supplementary material (Appendix B).

Baselines. We compare to the following baselines, which are broadly sourced from three groups. (1) The first group is designed for ZSL, namely, GSMFlow [13]. (2) The second group is designed for OSR, which includes MSP [23], ViM [51] and KNN [48]. (3) The third group is designed to address OZSL and includes the proposed naïve generative methods, namely GSMFlow-MSP, GSMFlow-ViM, GSMFlow-KNN as mentioned in Sect. 3.2. Note that for fair comparisons, we used the same synthetic samples generated via the GSMFlow model for unseen classes across these baselines and our approach. While we can flexibly use any generator, and generally, better generator will lead to better classification performance, GSMFlow has shown strong empirical results.

In addition to the above baselines, we also consider Contrastive Language-Image Pre-training (CLIP) [41]. It is worth noting that CLIP was not originally designed for our setting, as it is pre-trained on a vast amount of instance-level textual information and is likely to have encountered

	AWA1				CUB				FLO				SUN			
Method	**Seen**	**Unseen**	**Unk**	**HM**	**Seen**	**Unseen**	**Unk**	**HM**	**Seen**	**Unseen**	**Unk**	**HM**	**Seen**	**Unseen**	**Unk**	**HM**
GSM (ZSL)	79.9±2.5	82.3±5.6	-	-	63.7±1.1	71.4±3.7	-	-	88.7±0.6	88.2±5.1	-	-	31.5±0.5	37.7±1.1	-	-
CLIP (ZSL)	/	/	/	/	60.5±0.4	61.5±5.7	-	-	71.3±0.2	81.1±9.3	-	-	53.7±0.2	59.2±2.8	-	-
MSP (OSR)	74.1±0.2	-	89.5±5.7	-	59.9±0.3	-	49.4±3.3	-	78.6±0.6	-	61.7±10.3	-	34.8±0.1	-	61.4±1.5	-
ViM (OSR)	26.6±2.3	-	40.8±11.0	-	43.3±1.2	-	44.4±4.8	-	51.9±1.5	-	23.1±6.6	-	34.7±0.2	-	15.1±1.4	-
KNN (OSR)	44.3±2.8	-	72.3±11.2	-	39.7±0.9	-	31.1±3.6	-	58.4±1.4	-	35.1±6.8	-	38.1±0.3	-	24.9±1.7	-
GSM-MSP (OZSL)	60.4±1.5	43.1±10.7	83.3±2.4	57.8±5.5	41.4±1.8	39.1±3.9	70.7±3.8	45.3±1.4	30.2±4.2	33.4±9.1	93.6±3.6	36.8±4.9	29.1±0.2	36.9±1.2	15.9±1.1	25.2±0.7
GSM-ViM (OZSL)	19.1±1.4	42.5±4.1	24.2±7.1	23.1±1.6	41.1±1.2	41.3±3.2	43.9±5.1	41.6±1.6	51.1±0.7	67.5±5.3	20.9±1.8	36.4±1.8	26.3±0.5	18.1±0.6	15.1±0.7	20.2±0.6
GSM-KNN (OZSL)	47.3±1.1	47.2±10.3	65.2±9.4	50.3±4.8	36.4±2.1	45.1±3.1	36.2±4.9	38.1±1.7	55.3±1.3	54.4±3.1	32.8±3.6	44.7±2.5	26.7±0.4	28.1±0.9	23.4±0.1	26.1±0.4
CLIP-MSP (OZSL)	/	/	/	/	57.1±0.6	57.7±6.6	30.4±3.6	46.7±2.8	61.8±0.5	73.5±10.1	54.8±7.3	61.9±2.2	53.4±0.2	58.7±2.8	10.1±1.2	26.1±1.8
CLIPN (OZSL)	/	/	/	/	61.7±0.8	57.7±7.5	13.7±2.5	32.2±3.1	69.2±0.2	75.1±8.3	20.6±9.2	42.4±8.6	54.7±0.1	60.9±3.4	10.6±2.8	26.8±4.3
Ours (OZSL)	69.5±1.0	47.9±3.7	51.6±5.4	**57.9**±3.1	46.1±0.3	41.9±5.4	53.8±2.2	**47.5**±2.4	80.1±0.6	51.1±9.1	61.1±7.1	**65.5**±5.2	29.3±2.1	31.7±1.1	40.8±2.2	**32.1**±1.3

Table 2. Top-1 accuracy of our proposed OZSL method and the baselines. '-' denotes that the method cannot handle a certain category of samples. '/' denotes that CLIP-based methods cannot be applied to the AWA1 dataset, as image input is required by CLIP but AWA1 has not made the images public. GSMFlow is abbreviated as 'GSM'.

the unseen class images during the pre-trainng stage. On the contrary, other baselines and our proposed method are trained on a single description for each in-distribution class and have not seen any unseen class images before inference. Nevertheless, the pre-trained CLIP can still be used for ZSL. Likewise, CLIP+MSP is an extension that employs a threshold on the cosine similarity measure, and can be used for OZSL. Lastly, a variant of CLIP, called CLIPN [50], is designed to handle unknown samples in the OZSL setting. A detailed introduction of the baselines and their settings can be found in the supplementary material (Appendix C).

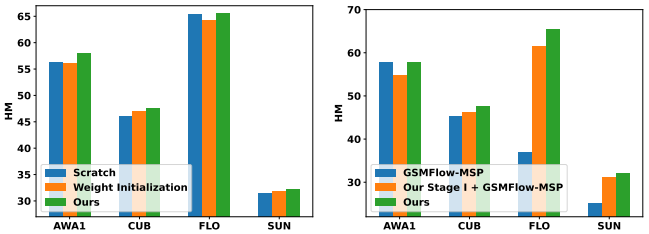
5.2. Performance Comparison with Baselines

We report the results of various baselines and our approach in Table 2, and make several observations.

First, the group of ZSL approaches including GSMFlow and CLIP can detect samples from seen and unseen classes, but cannot detect unknown samples. This limitation restricts their application in open-world environments. Furthermore, the inability to handle unknown samples can significantly increase their false positive rates on the seen and unseen classes, as some unknown samples would be misclassified into these classes.

Similarly, the OSR approaches, namely, MSP, ViM and KNN can detect samples from the seen classes and can flag unknown samples. However, they cannot identify samples from the unseen classes which increases the false positive rates on seen and unknown categories.

In the third group, we compare to several OZSL baselines. First, as discussed in Sect. 1, it is difficult to handle samples from all three categories (seen, unseen, and unknown) simultaneously. Hence, one-stage approaches are inferior, especially in the performance of seen classes. Nevertheless, among the one-stage approaches, GSMFlow-ViM/KNN tend to face more difficulty than the threshold-based methods. A potential reason could be that they estimate the probability of a sample being unknown based on the generated synthetic samples, which can contain significant noises compared to the actual samples. In contrast, threshold-based methods use a confidence threshold



(a) Effect of knowledge transfer (b) Effect of two-stage learning

Figure 4. Results of ablation studies.

to determine unknown samples, performing more consistently across all three categories and emerging as strong baselines. Meanwhile, CLIPN can generalize well for seen and unseen classes, but its performance on unknown is quite low. Finally, our proposed two-stage approach consistently achieves robust performance across all three categories and three datasets, resulting in highest overall performance as measured by the harmonic mean.

5.3. Ablation Studies

We first conduct an ablation study to show the contribution of the proposed knowledge transfer mechanism. In that regard, we consider three variants, namely, (1) *Scratch*: Training Stage II from random initial weights and without distribution retainment; (2) *Weight Initialization*: Training Stage II with the trained weights of Stage I as the initial weights and without distribution retainment; and (3) Our proposed full model, i.e., with both weight initialization and distribution retainment. Fig. 4(a) shows the harmonic mean of the three variants. It can be seen that *Weight Initialization* can sometimes lead to lower performance while our proposed full model always gives better performance. This is because, just weight initialization cannot force the model to maintain seen class similarity while learning new weights for unseen classes.

Second, we consider the effect of performing OZSL in two stages. In that regard, we again compare three variants. (1) *GSMFlow-MSP*: the one-stage baseline method that demonstrates competitive performance across all datasets in

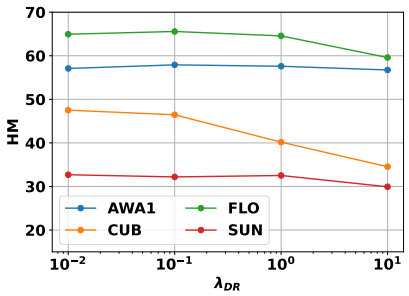


Figure 5. Effect of λ_{DR} on the overall model performance.

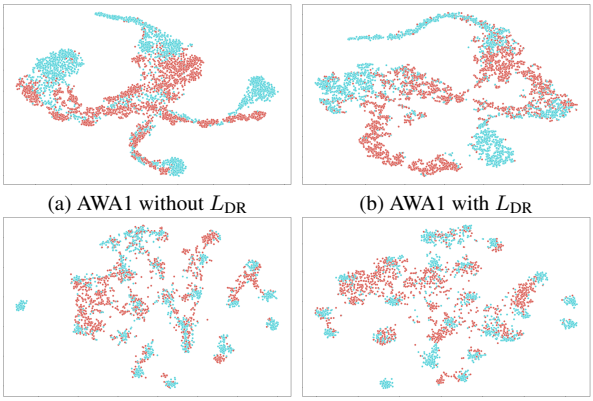
Table 2; (2) *Our Stage I + GSMFlow-MSP*: a modified two-stage baseline where the first stage is our proposed Stage I to determine whether a sample is seen or other classes (unseen or unknown) and the second stage is the GSMFlow-MSP method which determines whether a sample is unseen or unknown; (3) Our proposed two-stage approach. Fig. 4(b) compares the performance of the three variants. Results show that, simply extending our Stage I with a simple threshold-based second stage would already improve performance over the one-stage baseline. Meanwhile, our proposed two-stage approach achieves the best performance, demonstrating the benefit of our design with cross-stage knowledge transfer.

5.4. Parameter Sensitivity Analysis

The proposed distribution retainment loss is weighted by a hyperparameter, λ_{DR}, in order to achieve a balanced trade-off with other losses. Here we investigate the effect of the weight on the overall performance. Specifically, we vary the value of λ_{DR} over $\{0.01, 0.1, 1, 10\}$, and present the harmonic mean on the three datasets in Fig. 5. The results show a similar pattern across the datasets: Generally, lower values of λ_{DR} lead to robust performance. On the other hand, higher values often result in suboptimal performance, implying that the proposed distribution retainment serves as an auxiliary constraint to bridge the two stages, but the two stages are still distinct. Overall, $\lambda_{\text{DR}} = 0.1$ appears to be a robust setting across different datasets.

5.5. Visualization

We aim to get an intuitive understanding of how the proposed distribution retainment impacts the learned representations. Specifically, we plot the latent space of Stage II without or with the distribution retainment loss in Fig. 6, on the AWA1 and CUB datasets. It is evident that employing the distribution retainment loss reduces the overlap between the unseen class samples and the unknown samples as compared to not using the loss. This further verifies that our proposed distribution retainment helps in drawing more crisp decision boundaries between the unknown samples and the unseen class samples.



(a) AWA1 without L_{DR} (b) AWA1 with L_{DR}

(c) CUB without L_{DR} (d) CUB with L_{DR}

Figure 6. t-SNE visualization of the latent space for Stage II of our proposed model, without or with the distribution retainment loss L_{DR}. The blue points represent the unseen samples and the red points represent the unknown samples.

6. Conclusion

Detection of unintentionally encountered unknown samples during inference in the real world is an important problem that needs to be addressed to ensure the trustworthiness of the model. In this paper, we tackle a novel and practical problem, i.e., OZSL by proposing a two-stage approach wherein we first identify the seen class samples from the rest (unseen and unknown) in Stage I and identify unseen class samples from unknown samples in Stage II. Furthermore, we propose a cross-stage knowledge transfer mechanism in order to leverage the semantic relatedness between seen and unseen classes, to improve the overall performance of the model. We show the efficacy of the proposed method on three benchmark ZSL datasets, which are modified to address the OZSL problem.

Acknowledgment

This research/project is supported by the Ministry of Education, Singapore, under its Academic Research Fund Tier 1 grant (22-SIS-SMU-054). It is also supported by the National Research Foundation, Singapore, under its AI Singapore Programme (AISG Award No: AISG2-RP-2021-027). Additionally, this research is part of the DesCartes Programme and is supported by the National Research Foundation, Prime Minister's Office, Singapore, under its Campus for Research Excellence and Technological Enterprise (CREATE) programme. The opinions, findings, conclusions, or recommendations expressed in this material are solely those of the author(s) and do not necessarily reflect the views of the Ministry of Education, Singapore, the National Research Foundation, or any other supporting organizations.

References

[1] Zeynep Akata, Mateusz Malinowski, Mario Fritz, and Bernt Schiele. Multi-cue zero-shot learning with strong supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 59–68, 2016. 3

[2] Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(7):1425–1438, 2015. 6

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

[4] Faisal Alamri and Anjan Dutta. Multi-head self-attention via vision transformer for zero-shot learning. *arXiv preprint arXiv:2108.00045*, 2021. 3

[5] Yuval Atzmon and Gal Chechik. Adaptive confidence smoothing for generalized zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11671–11680, 2019. 3

[6] Abhijit Bendale and Terrance E Boult. Towards open set deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1563–1572, 2016. 3

[7] Yannick Le Cacheux, Herve Le Borgne, and Michel Crucianu. Modeling inter and intra-class relations in the triplet loss for zero-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10333–10342, 2019. 3

[8] Yannick Le Cacheux, Herve Le Borgne, and Michel Crucianu. Modeling inter and intra-class relations in the triplet loss for zero-shot learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10333–10342, 2019. 3

[9] Alexander Cao, Yuan Luo, and Diego Klabjan. Open-set recognition with gaussian mixture variational autoencoders. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 6877–6884, 2021. 2

[10] Jiaoyan Chen, Yuxia Geng, Zhuo Chen, Ian Horrocks, Jeff Z. Pan, and Huajun Chen. Knowledge-aware zero-shot learning: Survey and perspective. In Zhi-Hua Zhou, editor, *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, pages 4366–4373. International Joint Conferences on Artificial Intelligence Organization, 8 2021. Survey Track. 1

[11] Xingyu Chen, Xuguang Lan, Fuchun Sun, and Nanning Zheng. A boundary based out-of-distribution classifier for generalized zero-shot learning. *arXiv e-prints*, pages arXiv–2008, 2020. 2, 3, 4

[12] Zhi Chen, Zi Huang, Jingjing Li, and Zheng Zhang. Entropy-based uncertainty calibration for generalized zero-shot learning. In *Databases Theory and Applications: 32nd Australasian Database Conference, Proceedings 32*, pages 139–151. Springer, 2021. 3

[13] Zhi Chen, Yadan Luo, Sen Wang, Jingjing Li, and Zi Huang. Gsmflow: Generation shifts mitigating flow for generalized zero-shot learning. *IEEE Transactions on Multimedia*, 2022. 3, 4, 6

[14] Zhi Chen, Yadan Luo, Sen Wang, Ruihong Qiu, Jingjing Li, and Zi Huang. Mitigating generation shifts for generalized zero-shot learning. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 844–852, 2021. 1

[15] Tim R Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M Tomczak. Hyperspherical variational auto-encoders. In *34th Conference on Uncertainty in Artificial Intelligence*, pages 856–865, 2018. 4

[16] Georgiana Dinu, Angeliki Lazaridou, and Marco Baroni. Improving zero-shot learning by mitigating the hubness problem. *arXiv preprint arXiv:1412.6568*, 2014. 3

[17] Mohamed Elhoseiny, Kai Yi, and Mohamed Elfeki. Cizsl++: Creativity inspired generative zero-shot learning. *arXiv preprint arXiv:2101.00173*, 2021. 3

[18] Angelos Filos, Panagiotis Tigkas, Rowan McAllister, Nicholas Rhinehart, Sergey Levine, and Yarin Gal. Can autonomous vehicles identify, recover from, and adapt to distribution shifts? In *International Conference on Machine Learning*, pages 3145–3153, 2020. 2

[19] 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. 3

[20] Yanwei Fu, Timothy M Hospedales, Tao Xiang, Zhenyong Fu, and Shaogang Gong. Transductive multi-view embedding for zero-shot recognition and annotation. In *European Conference on Computer Vision*, pages 584–599, 2014. 3

[21] Yanwei Fu and Leonid Sigal. Semi-supervised vocabulary-informed learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5337–5346, 2016. 3

[22] Chandan Gautam, Sethupathy Parameswaran, Vinay Verma, Suresh Sundaram, and Savitha Ramasamy. Refinement matters: Textual description needs to be refined for zero-shot learning. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6127–6140, 2022. 1, 3

[23] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *International Conference on Learning Representations*, 2016. 3, 4, 6

[24] Dat Huynh and Ehsan Elhamifar. Fine-grained generalized zero-shot learning via dense attribute-based attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4483–4493, 2020. 3

[25] Taotao Jing, Hongfu Liu, and Zhengming Ding. Towards novel target discovery through open-set domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9322–9331, 2021. 3

[26] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 951–958, 2009. 6

[27] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *IEEE transactions on pattern analysis and machine intelligence*, 36(3):453–465, 2013. 3

[28] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Advances in Neural Information Processing Systems*, 31, 2018. 3

[29] KIMIN LEE, Kibok Lee, Honglak Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *International Conference on Learning Representations*, 2018. 3

[30] Jimmy Lei Ba, Kevin Swersky, Sanja Fidler, et al. Predicting deep zero-shot convolutional neural networks using textual descriptions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4247–4255, 2015. 3

[31] Jingjing Li, Mengmeng Jing, Ke Lu, Zhengming Ding, Lei Zhu, and Zi Huang. Leveraging the invariant side of generative zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7402–7411, 2019. 3

[32] Jingjing Li, Mengmeng Jing, Ke Lu, Lei Zhu, Yang Yang, and Zi Huang. Alleviating feature confusion for generative zero-shot learning. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 1587–1595, 2019. 2

[33] Kai Li, Martin Renqiang Min, and Yun Fu. Rethinking zero-shot learning: A conditional visual classification perspective. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3583–3592, 2019. 3

[34] Tingting Ma, Jin-Ge Yao, Chin-Yew Lin, and Tiejun Zhao. Issues with entailment-based zero-shot text classification. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 786–796, 2021. 3

[35] Devraj Mandal, Sanath Narayan, Sai Kumar Dwivedi, Vikram Gupta, Shuaib Ahmed, Fahad Shahbaz Khan, and Ling Shao. Out-of-distribution detection for generalized zero-shot action recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9985–9993, 2019. 3

[36] Ashish Mishra, Shiva Krishna Reddy, Anurag Mittal, and Hema A Murthy. A generative model for zero shot learning using conditional variational autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 2188–2196, 2018. 3

[37] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729, 2008. 6

[38] Poojan Oza and Vishal M Patel. C2ae: Class conditioned auto-encoder for open-set recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2307–2316, 2019. 2

[39] Marc Pàmies, Joan Llop, Francesco Multari, Nicolau Duran-Silva, César Parra-Rojas, Aitor González-Agirre, Francesco Alessandro Massucci, and Marta Villegas. A weakly supervised textual entailment approach to zero-shot text classification. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 286–296, 2023. 3

[40] Genevieve Patterson, Chen Xu, Hang Su, and James Hays. The sun attribute database: Beyond categories for deeper scene understanding. *International Journal of Computer Vision*, 108:59–81, 2014. 6

[41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763, 2021. 6

[42] Bernardino Romera-Paredes and Philip Torr. An embarrassingly simple approach to zero-shot learning. In *International Conference on Machine Learning*, pages 2152–2161, 2015. 3

[43] Edgar Schonfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. Generalized zero-and few-shot learning via aligned variational autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8247–8255, 2019. 3

[44] Yuming Shen, Jie Qin, Lei Huang, Li Liu, Fan Zhu, and Ling Shao. Invertible zero-shot recognition flows. In *Computer Vision–ECCV 2020: 16th European Conference Proceedings, Part XVI 16*, pages 614–631, 2020. 3

[45] Ivan Skorokhodov and Mohamed Elhoseiny. Class normalization for (continual)? generalized zero-shot learning. In *International Conference on Learning Representations*, 2021. 1, 3

[46] Ivan Skorokhodov and Mohamed Elhoseiny. Class normalization for (continual)? generalized zero-shot learning. In *International Conference on Learning Representations*, 2021. 3

[47] 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. 3

[48] Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. In *International Conference on Machine Learning*, pages 20827–20840. PMLR, 2022. 4, 6

[49] Vinay Kumar Verma, Gundeep Arora, Ashish Mishra, and Piyush Rai. Generalized zero-shot learning via synthesized examples. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4281–4289, 2018. 3

[50] Hualiang Wang, Yi Li, Huifeng Yao, and Xiaomeng Li. Clipn for zero-shot ood detection: Teaching clip to say no. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1802–1812, 2023. 7

[51] Haoqi Wang, Zhizhong Li, Litong Feng, and Wayne Zhang. Vim: Out-of-distribution with virtual-logit matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4921–4930, 2022. 3, 4, 6

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

[53] Zheng Wang, Jialong Wang, Yuchen Guo, and Zhiguo Gong. Zero-shot node classification with decomposed graph prototype network. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1769–1779, 2021. 3

[54] Zhihao Wen and Yuan Fang. Augmenting low-resource text classification with graph-grounded pre-training and prompting. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 506–516, 2023. 3

[55] Kelvin Wong, Shenlong Wang, Mengye Ren, Ming Liang, and Raquel Urtasun. Identifying unknown instances for autonomous driving. In *Conference on Robot Learning*, pages 384–393, 2020. 2

[56] Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5542–5551, 2018. 3, 6

[57] Jiacheng Xu and Greg Durrett. Spherical latent spaces for stable variational autoencoders. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 4503–4513, 2018. 4

[58] Wenjia Xu, Yongqin Xian, Jiuniu Wang, Bernt Schiele, and Zeynep Akata. Attribute prototype network for zero-shot learning. *Advances in Neural Information Processing Systems*, 33:21969–21980, 2020. 3

[59] Jiacheng Ye, Jiahui Gao, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. ProGen: Progressive zero-shot dataset generation via in-context feedback. *arXiv preprint arXiv:2210.12329*, 2022. 3

[60] Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. ZeroGen: Efficient zero-shot learning via dataset generation. *arXiv preprint arXiv:2202.07922*, 2022. 3

[61] Li Zhang, Tao Xiang, and Shaogang Gong. Learning a deep embedding model for zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2021–2030, 2017. 3