GENERALIZED ZERO-SHOT OBJECT RECOGNITION WITHOUT CLASS-ATTRIBUTE RELATIONS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

MÜSLÜM ERSEL ER

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

FEBRUARY 2021

Approval of the thesis:

GENERALIZED ZERO-SHOT OBJECT RECOGNITION WITHOUT CLASS-ATTRIBUTE RELATIONS

submitted by **MÜSLÜM ERSEL ER** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Halit Oğuztüzün
Head of Department, **Computer Engineering** _____

Assist. Prof. Dr. Ramazan Gökberk Cinbiş
Supervisor, **Computer Engineering, METU** _____

Examining Committee Members:

Assoc. Prof. Dr. Selim Aksoy
Computer Engineering, Bilkent University _____

Assist. Prof. Dr. Ramazan Gökberk Cinbiş
Computer Engineering, METU _____

Assist. Prof. Dr. Emre Akbaş
Computer Engineering, METU _____

Date: 11.02.2021

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Müslüm Ersel Er

Signature :

ABSTRACT

GENERALIZED ZERO-SHOT OBJECT RECOGNITION WITHOUT CLASS-ATTRIBUTE RELATIONS

Er, Müslüm Ersel
M.S., Department of Computer Engineering
Supervisor: Assist. Prof. Dr. Ramazan Gökberk Cinbiş

February 2021, 40 pages

Over the last decade, great improvements have been achieved in image classification performances following the advances in supervised deep learning approaches. These supervised approaches, however, typically require substantial amounts of labeled training examples. Collecting and annotating such examples is a cumbersome and error-prone task, especially when a large number of classes needs to be spanned. One of the promising approaches towards overcoming this limitation of supervised recognition techniques is zero-shot learning. Inspired by the abilities of human vision, zero-shot learning aims to enable recognition of novel object categories purely based on category-wide information, which we refer to as *auxiliary class information*. A more modern variant, called generalized zero-shot learning, aims to build models that can accurately classify novel samples of not only zero-shot classes but also those with supervised training examples. Most of the recent generalized zero-shot learning approaches rely on attribute based auxiliary class information, where the attributes characterising each class of interest needs to be defined by an oracle. In practice, this dependency greatly reduces the practicality of zero-shot learning as it is often difficult to define such class-attribute relationships. To bypass this requirement, in

this thesis, we propose a model that requires only class names of novel classes and implicitly learns *pseudo-attributes* in an end-to-end manner purely based on a set of candidate pseudo-attribute word embeddings. Such word embeddings are much easier to collect than class-attribute annotations, as one can easily select and utilize a set of relevant words from a pre-trained language model that provides vector-space word embeddings. Additionally, we propose a simple contrastive loss term for improving generalized zero-shot learning based on simple class-to-class name similarity scores. Our experimental results show that the proposed approach yields state-of-the-art class name based generalized zero-shot learning.

Keywords: Generalized Zero-Shot Learning, contrastive loss, pseudo-attributes, word representations

ÖZ

SINIF-NİTELİK İLİŞKİLERİ GEREKTİRMEYEN GENELLEŞTİRİLMİŞ SIFIR-ÖRNEKLİ NESNE TANIMA

Er, Müslüm Ersel
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Dr. Öğr. Üyesi. Ramazan Gökberk Cinbiş

Şubat 2021 , 40 sayfa

Son on yılda, gözetimli derin öğrenme yaklaşımlarındaki ilerlemelerin ardından görüntü sınıflandırma performanslarında büyük gelişmeler kaydedildi. Ancak, bu gözetimli yaklaşımlar, genellikle büyük miktarlarda etiketli eğitim verilerine ihtiyaç duymaktadır. Bu örneklerin toplanması ve etiketlenmesi özellikle çok sayıda sınıfı kapsamak gerektiğinde oldukça külfetli ve hataya açık bir iştir. Gözetimli tanıma tekniklerine ait bu sınırın üstesinden gelmeye yönelik umut verici yaklaşımlardan biri, sıfır-örnekli öğrenmedir. İnsanın görme yeteneklerinden esinlenen sıfır-örnekli öğrenme, *yardımcı sınıf bilgisi* olarak adlandırdığımız, tamamen kategori genelindeki bilgilere dayanarak yeni nesne kategorilerinin tanınmasını amaçlamaktadır. Genelleştirilmiş sıfır-örnekli öğrenme adı verilen daha modern bir alternatifi ise, sadece sıfır-örnekli sınıfların yeni örneklerini değil, aynı zamanda gözetimli eğitim örnekleri bulunan sınıfların yeni örneklerini de doğru şekilde sınıflandırabilen modeller oluşturmayı amaçlamaktadır. Son zamanlardaki genelleştirilmiş sıfır-örnekli öğrenme yaklaşımları, her bir ilgili sınıfı karakterize eden niteliklerin bir uzman tarafından tanımlanmasını gerektiren, nitelik tabanlı yardımcı sınıf bilgisine dayanır. Gerçekte, bu

tür sınıf-nitelik ilişkilerini tanımlamak genellikle zor olduğundan, bu gereklilik sıfır-örnekli öğrenmenin uygulanabilirliğini büyük ölçüde azaltır. Bu gerekliliği aşmak için, bu tezde, yalnızca yeni sınıfların adlarına ihtiyaç duyan ve sözde-niteliklerini tamamen bir dizi aday sözde-niteliklerin kelime temsillerine dayalı olarak uçtan uca dolaylı öğrenen bir model öneriyoruz. Bu tür kelime temsillerinin elde edilmesi, sınıf-nitelik ilişkilerine göre çok daha kolaydır, çünkü vektör uzayında kelime temsillerini sağlayan önceden eğitilmiş bir dil modelinden ilgili bir dizi kelime kolayca seçilip kullanılabilir. Ek olarak, genelleştirilmiş sıfır-örnekli öğrenmeyi geliştirmek için sınıftan sınıfa isim benzerlik skorlarına dayanan karşılaştırmalı basit bir kayıp terimi öneriyoruz. Deneysel sonuçlarımız, önerilen yaklaşımın sınıf adına dayalı genelleştirilmiş sıfır-örnekli öğrenme için en başarılı olduğunu göstermektedir.

Anahtar Kelimeler: Genelleştirilmiş Sıfır-Örnekli Öğrenme, karşılaştırmalı loss, sözde-nitelik, kelime temsilleri

To my dear family...

ACKNOWLEDGMENTS

I am very grateful to my supervisor Assist. Prof. Dr. Ramazan Gökberk Cinbiş who taught and led me to do research, helped me with his guidance to find the correct answers and always encouraged me when I was pessimistic. He is a great mentor to light the way.

I give my sincere thanks to committee members, Assoc. Prof. Dr. Selim Aksoy and Assist. Prof. Dr. Emre Akbaş for reviewing my thesis and giving their valuable advice.

I am also grateful to my girlfriend Damla Özdemir who always believed and supported me during my master years.

I also wish to thank my friends. They supported and encouraged me whenever I needed it.

I also thank my colleagues since they always forced me to write my thesis whenever I feel tired.

I also would like to mention that this work was supported in part by the TUBITAK Grant 116E445.

Lastly, I give my sincere thanks to my family. They supported me at any time and in all circumstances.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

ZSL	Zero-Shot Learning
GZSL	Generalized Zero-Shot Learning
CNN	Convolutional Neural Network
Eq.	Equation
Att.	Attribute
Emb.	Embedding
ConvNet	Convolutional Neural Network

CHAPTER 1

INTRODUCTION

1.1 Motivation and Problem Definition

Image classification over a small number of classes is generally considered as a well-addressed task in computer vision with the recent advances in deep learning approaches. On the other hand, a large number of labelled training data is necessary for these approaches. It is well known that data collection and annotation is a very laborious task, especially in the presence of fine-grained classes. Therefore, a variety of learning frameworks have been proposed over the past couple years towards reducing data set collection efforts, such as semi-supervised learning [1, 2, 3], transductive learning [4, 5, 6], self-supervised learning [7, 8], unsupervised learning [9, 10], zero-shot learning [11, 12, 13] and few-shot learning [14, 15, 16].

Zero-shot learning, specifically zero-shot object recognition, is the task of identification of the objects that are not observed during training by transferring knowledge from seen classes to unseen classes. ZSL requires shared auxiliary information among all classes to enable knowledge transfer. Typically, these auxiliary sources can either be *attributes* which indicate the presence or absence of certain visual properties in the images of a particular class, *word vectors* which are high dimensional representation of the class names given by a language model learned from a corpus or *textual descriptions* of classes.

Figure 1.1 shows an illustration of the ZSL problem. On the upper side, there are training samples that belong to seen classes such as *horse*, *leopard* and *tiger*. As can be seen in the figure, all classes are presented as points in a vector space called *semantic space* as either attribute vectors (left) or word vectors (right). In the attribute

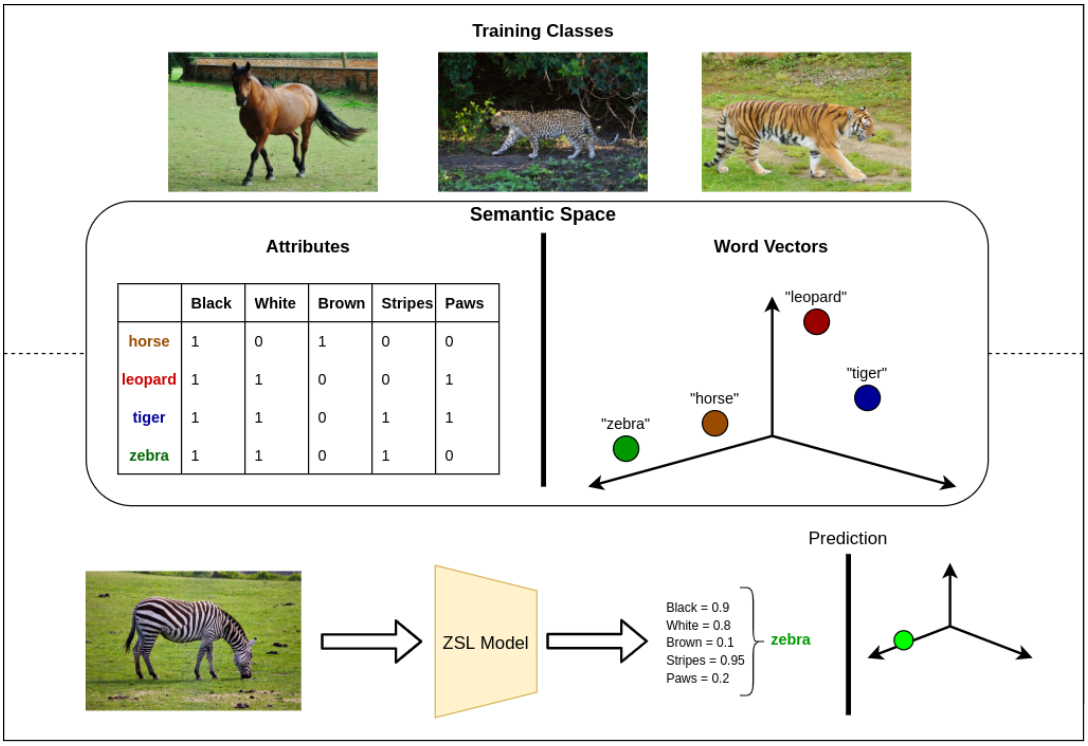

Figure 1.1: Illustration of zero-shot learning: A zero-shot learning model aims to recognize an unseen object defined on semantic space by learning a representation from seen classes.

based approach, zebra as an unseen test class has common visual attributes with other training classes such as *stripes* of the tiger. Since the shared attributes among the seen and unseen classes can be visually meaningful, a ZSL model is able to recognize unseen classes through such common attributes. In the class name based approach, all classes are represented in terms of class name embedding vectors. Since word embeddings are high dimensional representation of the words learned from a corpus, they are much easier to obtain yet they typically have only weak visual clues. In general, a zero-shot learning model aims to recognize an unseen object defined on semantic space (as attributes or word vectors) by learning a transfer model in the semantic space, based on the training examples of seen classes.

In conventional ZSL setting, test samples belong to only unseen classes which is not practical in real world scenarios. Thus, we focus on generalized ZSL setting (GZSL) [17], where the assumption is that test samples may belong to unseen or seen classes. Xian et al. [17] has shown that many traditional ZSL approaches perform worse in

GZSL setting. Traditional ZSL approaches tend to produce models that fit well only to the seen classes due to the domain shift problem, therefore these approaches have poor performance in GZSL setting. Aiming to address this problem, inspired from TCN [18], we propose a contrastive loss function to make our model more robust to the unseen classes. The intuition behind our approach is the recognition ability of human for the objects that they have never seen but heard names of them from similar classes. For example, we as human beings are able to recognize whether an animal in a given image is a *horse* or an *elephant* or a *humpback whale* if we have already seen an image of *blue whale*. In order to achieve this, we propose a contrastive loss function which measures distance between the seen images and the closest unseen class word vector to the true class name vector of the seen images. Thus, we enable the model to learn unseen classes from seen classes that have similar names and force the model to be more robust to unseen classes. In other words, our model learns unseen classes from samples that belongs to similar seen classes.

In addition, some traditional ZSL approaches [19] which use word vectors, learn a mapping from visual space to semantic space. However, these approaches are not able to perform well in GZSL setting due to lack of visual information in semantic space. For example, an image of *horse* has attributes such as *tail*, *head*, *leg* etc. which are some kind of visual information. However, the word representation of the *horse* itself does not contain any visual clue. Aiming to overcome this problem, similar to the intuition in A2CN [20], we propose to incorporate attribute names or attribute semantic vectors into our ZSL model. Thus, we aim to utilize attribute names to decrease the gap between visual space and semantic space while improving the recognition. Different from A2CN [20], we define a flexible model that allows us to use not only attribute name embeddings but also arbitrary word embeddings.

Finally, we conduct extensive experiments on five zero-shot learning benchmarks which are CUB [21], AWA1 [11], AWA2 [17], SUN [22] and APY [23]. We specifically compare against baselines on class name based GZSL, which is known to be a more practical yet more difficult problem, compared to GZSL based on (manually) predefined per-class attributes. Experimental results show that our approach improves the baseline and outperforms state of the art models.

1.2 Contributions and Novelties

Main contributions of this work are as follows: First, we explore the use of implicitly recognized visual concepts as mid-level concepts, which we call *pseudo-attributes*, for attribute-free GZSL. Our model, namely *pseudo-attribute name embedding (PANE)*, learns pseudo-attributes internally through end-to-end training. Second, to address the problem of over-confident seen class predictions, we propose a contrastive loss term by using class similarities of the seen and unseen classes, thus the model becomes more robust to unseen classes. Experimental results shows that the contrastive loss term greatly improves the performance of attribute-free GZSL. Third, we explore the use of pseudo-attribute names in order to tackle the lack of visual clue in class name word representations. In addition, we show that the proposed contrastive loss term can be applied to other models. In particular, we show that our novel loss term improves attribute-free GZSL performance of not only our model but also that of the basic ALE [24] model. Finally, unlike commonly studied attribute based GZSL methods, our approach does not require prior knowledge on class-attribute relations, which are cumbersome to collect, during training or testing. Instead, our approach requires only class name embeddings and attribute name embeddings.

1.3 The Outline of the Thesis

The rest of the thesis is as follows: In Chapter 2, we review the most relevant works and discuss the similarities and differences to the proposed approach. In Chapter 3, we explain our proposed approach to the problem of object recognition in GZSL. In Chapter 4, first of all, we give details about the experimental setup and then provide the experimental results. In Chapter 5, we summarize and conclude the thesis.

CHAPTER 2

RELATED WORK

In this section, we review the most related works to our approach and recent works of ZSL. In Section 2.1, we review the ZSL literature and group the works. In Section 2.2, we discuss the differences and similarities of our approach to the closest works in detail.

2.1 Zero-Shot Learning

2.1.1 Discriminative and generative approaches

In the last decade, a significant amount of work has been reported in the literature of ZSL. Compatibility based approaches, such as DEVISE [25], ALE [24], SJE [26], ESZSL [27], learn a linear compatibility function between the visual space and semantic space. In contrast to linear approaches, LatEm [28] proposes non-linear compatibility function and CMT [29] learns non-linear mapping from visual space to semantic space. [25, 24, 27, 28, 29] learn mapping from one modality to another. Alternatively, JLSE [30] maps the visual space and the semantic space into two latent spaces and learns a compatibility function over the latent spaces whereas SSE [31] maps the visual and semantic space into a common intermediate space. Although these approaches [25, 24, 27, 28, 29, 30, 31, 32] have key differences from each other, they can be grouped as discriminative methods where the models learn to estimate associations between the visual space and the semantic space.

Although several works build on discriminative learning formulations, in recent years, generative methods have gained attention in the ZSL literature [33, 34, 35, 36, 37,

38]. GFZSL [33] models the class-conditional distribution as a Gaussian and learns mapping from semantic space to latent space. GMN [34], CADA-VAE [36], f-CLSWGAN [37] and f-VAEGAN-D2 [38] learn to synthesize image features from class semantics e.g. attribute vectors and convert the ZSL problem into the traditional supervised object classification problem and learn a discriminative model in a supervised setting. Generative models have shown remarkable performance in zero-shot object classification, however, they are typically greatly more complicated and often difficult to tune. In this work, we focus on discriminative approaches.

2.1.2 Transductive learning

Another direction of zero-shot learning is transductive learning [4] where the assumption is that it is allowed to access unlabelled samples of the unseen classes. Typically, transductive methods aim to solve the domain shift problem [4]. In general, transductive ZSL approaches substantially outperforms the inductive approaches due to use of unseen data during training or in the pre-training phase [39, 5, 6, 40]. However, arguably the transductive setting is against the nature of the ZSL in most practical scenarios. In this work, therefore, we do not study transductive zero-shot learning. In addition, it is not fair to compare any inductive approach against the transductive ones. Thus, we don't compare our method against transductive ZSL approaches in our experiments.

2.1.3 Semantic space

In zero-shot learning literature, all class labels (seen and unseen) are defined on a common vector space, called semantic space, which enables to transfer knowledge from seen classes to unseen classes. Existing ZSL works use different semantic spaces; a large group of work, such as ALE [24], TCN [18], APN [41], DAZLE [42], RNet [14], DVBE [43] uses human-annotated attribute vectors which indicate the presence or absence of the attributes on the classes. Some of these works, such as ALE [24], and others, such as ConSE [19], LatEm [28], CAPD [44] (alternatively) use word embeddings [45, 46], which are high dimensional representation of the words

learned from a corpus and also some works [47, 48, 49, 50] use textual descriptions of classes for semantic embeddings. Typically, methods that leverage attributes as auxiliary sources outperform the others due to richer visual information. In this work, although the attributes represent the classes better in semantic space, we focus on the use of word vectors since obtaining attribute vectors typically require manual human annotations which are very difficult to collect in most cases.

2.2 Discussion on Closest Approaches

2.2.1 Domain shift problem and score calibration

Zero-shot learning methods are biased to seen (source) classes since (i) models are trained over seen class samples and (ii) unseen (target) and seen data have different distributions due to having completely disjoint sets. This is one of the main challenges in ZSL, and closely related to the domain shift problem [4] in general. To address this issue, a recent work TCN [18] argues that unseen classes can be learned from the similar seen classes and proposes a contrastive model by using the class similarities. The goal of the TCN is to make the model more robust to the unseen classes. TCN [18] uses sparse coding to compute the class similarities, instead we directly use cosine similarities of the class name embeddings which is a way simpler but still efficient. And also they use attributes whereas we use GloVe [45] word representations of the class names as semantic embeddings. It is likely to obtain more reliable class similarities when attributes are used as class embeddings due to richer visual information. However, our work aims to eliminate the need of attributes for zero-shot learning, thus we learn class similarities over the class name embeddings.

Deep Calibration Networks (DCN) [51] is another close work to ours which projects visual representations of the images and semantic representations of the classes into a common space to maximize the compatibility of the seen samples to both the seen and unseen classes. DCN [51] applies temperature calibration [52] over the seen classes and uncertainty calibration over the unseen classes in order to handle the problem of overconfidence to the seen classes due to training the deep networks by only seen samples. Different from DCN [51], we compute the cosine similarities of seen and

unseen classes to apply contrastive loss function. In addition, our model learns a compatibility function that maximizes the relevance score of the samples with the true classes where DCN [51] directly uses cosine similarity to compute the compatibility and applies calibration. Lastly, another completely different point of our approach compared to DCN is that our model learns mid-level representations via pseudo-attributes.

2.2.2 Class name embeddings

Our model implicitly learns mid-level representations and combines them with attribute name embeddings. In a variant of our model, we replace the attribute name embeddings with the class name embeddings which makes the model similar to ConSE [19], however our model differs in many fundamental ways. ConSE [19] is a two-stage approach: first it predicts seen class probabilities and computes the convex combination of the class semantics which can be either attributes or word representations of the seen classes with pre-computed class probabilities in order to project the given images into semantic space. Unlike ConSE [19], our model is trainable in an end-to-end manner, therefore our model allows using arbitrary word embeddings such as name embeddings of the seen classes, unseen classes or all of them in order to compute convex combinations. In addition, ConSE [19] incorporates only positive convex combinations due to the probabilistic approach whereas our model is able to combine the additive and subtractive combinations of the class name embeddings. Finally, we explicitly handle the score calibration problem for generalized zero-shot learning.

2.2.3 Attribute name embeddings

In our work, we focus on the use of attribute names or attribute semantic vectors towards improving recognition. Another work that utilizes attribute names is SGV [53] which aims to regularize graph convolutional network to improve their results. A recent work DAZLE [42] extracts attribute features in an image and aligns them with attribute word vectors to predict the scores of the attributes in the image which

is used for computing final score.

Demirel *et al.*'s approach [20] which is one of the closest work to our approach also uses attributes names but for a very different purpose, where the goal is to avoid the need of attributes of unseen classes. Our model formulation is inspired from Demirel *et al.*'s model [20], but differs in many details: (i) our model does not require attribute-class relations for the training classes, (ii) we use pseudo-attributes, instead of attributes, as mid-level constructs, which allows us to use other visually relevant concepts such as class names, (iii) our model incorporates not only positive convex combinations but arbitrary (additive and subtractive) combinations of mid level concepts. In addition, our model incorporates the additional GZSL loss term and is trainable in an end-to-end manner.

2.2.4 Baseline method: Attribute label embedding

One of the similar approaches to our work is the Attribute Label Embedding (ALE) [24]. Our model learns a compatibility function similar to the ALE [24] method, however there are some major differences in detail. First of all, ALE [24] learns the compatibility function using image representations and the class embeddings whereas our model uses image representations to obtain pseudo-attribute based image embedding which is the weighted sum of the attribute name embeddings with the corresponding pseudo-attribute predictions. Afterwards, pseudo-attribute based image embeddings and the class name embeddings are projected to a common space to learn the compatibility function that maximizes relevance score for the true class label. Our approach, in essence, learns mid-level concepts to estimate the compatibility of the image and the class name embeddings. Another major difference is that our model utilizes attribute name embeddings to reduce the gap between visual space and the semantic space. In addition, the original ALE method uses a ranking based loss function while our model is trained optimizing an contrastive loss function which makes the model more generalized to unseen classes.

Experimental results of ALE [24] that are provided by [17] are obtained using attribute vectors as class embeddings. However, for fair comparison, we need to have word embedding based results. Thus, we re-implemented ALE with class name em-

beddings. In Section 4.2.2 we present experimental results that are obtained by applying our proposed contrastive loss function and discuss them in detail .

CHAPTER 3

METHOD

First we define the problem of ZSL and GZSL in Section 3.1. In Section 3.2, we explain our approach to the ZSL and GZSL problem. In Section 3.3, we present our contrastive loss function that significantly improves the GZSL performance.

3.1 Problem Definition

For the task of zero-shot learning we assume that we have two disjoint sets called seen classes $\mathcal{Y}_{\mathrm{s}} = \{1, \ldots, C_s\}$ and unseen classes $\mathcal{Y}_{\mathrm{u}} = \{C_s + 1, \ldots, C_s + C_u\}$ such that they do not have a common class label $\mathcal{Y}_{\mathrm{s}} \cap \mathcal{Y}_{\mathrm{u}} = \emptyset$. Therefore, the set of all class labels corresponds to $\mathcal{Y}_{\mathrm{all}} = \mathcal{Y}_{\mathrm{s}} \cup \mathcal{Y}_{\mathrm{u}}$. In addition, we have N labelled images from only seen classes for training set $\mathcal{D}_{\mathrm{tr}} = \{(x_i, y_i) \mid x_i \in \mathcal{X}, y_i \in \mathcal{Y}_{\mathrm{s}}\}_{i=1}^{N}$, where $\mathcal{X}$ denotes the visual space.

In order to relate seen classes with unseen ones we have a unique semantic representation for each class. One option for these semantic representations is attribute vectors describing the relevance of each attribute for each class. While attributes provide powerful class representations and commonly used in most ZSL and (nearly all) GZSL approaches, they are often difficult to collect and annotate. Therefore, in this thesis, we presume that we do not have access to attribute-based class definitions and rely on only d_z-dimensional vector space embeddings of class names, which can be obtained using unsupervised word embedding models trained on relevant text corpora.

In our work, instead of building a model that relates image features directly to class

features, we aim to benefit from textual concepts, represented in the vector space word embeddings, as a mid-layer representation. Since we aim to take such a candidate set of mid-level word representations and leverage them in inferring an image representation tailored for zero-shot recognition purposes, we refer to these candidate words as *pseudo-attributes*. Our approach does not require any additional supervision and agnostic to the choice of such mid-level concepts, therefore, provides the flexibility of using any set of word embeddings for this purpose. In our experiments, we explore the use of a variety of such pseudo-attribute candidates, including actual attribute names, without requiring class-to-attribute annotations, and seen and/or unseen class names. In our formulation below, we refer to each candidate pseudo-attribute a's embedding as $z_a \in \mathbb{R}^{d_z}$.

In our experimental setup, we presume that each image is represented by a d_x dimensional feature vector, extracted using a pre-trained ConvNet. In the training set, each image x_i is paired with a class label $y_i \in \mathcal{Y}_s$. Additionally, we have access to a set of pseudo-attribute words and their word embeddings, with no per-class annotations regarding the pseudo-attributes. The goal, in conventional ZSL, is to learn a classifier on $\mathcal{D}_{\text{tr}}$ in order to predict the class label of an test image belonging to unseen classes $f_{zsl}\colon \mathcal{X} \rightarrow \mathcal{Y}_u$. In GZSL, the test image may belong to unseen classes as well as seen classes, therefore, the goal is learning a more comprehensive classifiers $f_{gzsl}\colon \mathcal{X} \rightarrow \mathcal{Y}_{all}$.

3.2 Proposed Approach

In this section we explain our novel approach, which we call *pseudo-attribute name embedding* (PANE).

One of the main goals of this work is to utilize (pseudo)-attribute names to fill in the visio-semantic gap between the visually-rich image features and the semantically-rich class word representations. For this purpose, we first aim to determine the presence (*i.e.* relevance) of the pseudo-attributes in a given image. Let $g\colon \mathbb{R}^{d_x} \rightarrow \mathbb{R}^{n_a}$ be the pseudo-attribute prediction function that predicts the presence or absence of the pseudo-attributes in an image x, where $x \in \mathbb{R}^{d_x}$ is the d_x dimensional feature repre-

sentation of the image x and n_a is the number of pseudo-attributes. Pseudo-attribute predictions are combined with pseudo-attribute names by computing an unnormalized weighted sum of attribute name embeddings, where each name is scaled by the corresponding prediction:

$$\psi(x) = \sum_a [g(x)]_a \, z_a \tag{3.1}$$

where z_a is the word representation of the pseudo-attribute name a, $z_a \in \mathbb{R}^{d_z}$. The output is a d_z dimensional word representation of the image x. We refer to $\psi(x)$ as *pseudo-attribute based image embedding*. In our experiments, we express $g(x)$ in terms of a fully connected layer with $\tanh$ activation function. Here, we intentionally choose an activation function with both positive and negative output possibilities to let model learn not only additive but also subtractive accumulations.

To obtain a zero-shot predictor, we define a compatibility function $f\colon \mathbb{R}^{d_z} \times \mathbb{R}^{d_z} \to \mathbb{R}$:

$$f(\psi(x), z_c; \theta) \tag{3.2}$$

where θ is the trainable parameters of the compatibility function. The role of the compatibility function is to measure the relevance of the input pseudo-attribute based image embedding $\psi(x)$ and the given class embeddings $z_c \in \mathbb{R}^{d_z}$ for some class c.

One simple option to construct a compatibility function is to use a bi-linear model between the pseudo-attribute based image embeddings and the class embeddings, as a commonly used in attribute-based zero-shot learning models, *e.g.* ALE [24]. We, however, have observed in our preliminary experiments that the model yields significantly higher results when it is expressed as the dot product of transformed pseudo-attribute based image embeddings and class name embeddings, probably due to weaker class knowledge provided by the class name embeddings, compared to attribute based definitions. Therefore, we define f as follows:

$$f(\psi(x), z_c) = \sigma(\tanh(\theta_\psi^T \psi(x))^T \tanh(\theta_c^T z_c)) \tag{3.3}$$

where θ_ψ and θ_c correspond to the trainable parameters of two fully-connected layers with $\tanh$ activation functions. This definition, therefore, non-linearly transforms input image and class embeddings into a shared space with activation values in range

$(-1, +1)$ and computes the correlation in this space. The final score is computed through sigmoid $\sigma()$.

Relation to existing methods. We note that our compatibility function takes the same role as in other label-embedding methods, but instead of directly utilizing the original image descriptors, it uses the pseudo-attribute based image embeddings. Our model is also closely related to A2CN [20], which also aims to compute an attribute-driven embedding of images as a mid-level representation for comparing against class name embeddings. Our model PANE has, however, in the following notable differences: (i) while A2CN requires class-attribute definitions of training classes, PANE requires no such annotations; (ii) while A2CN specifically requires a set of attributes and their vector space embeddings, where visual attribute name embeddings can be uninformative (such as names of color attributes) or hard to extract, PANE can utilize any set of (meaningfully chosen) names and their embeddings (such as class names) as pseudo-attributes; (iii) while A2CN requires pre-trained attribute recognition models, PANE is trained in an end-to-end manner, without requiring explicit attribute recognition models.

3.3 Contrastive Loss

In order to enable model to infer from seen samples simply we can apply cross-entropy loss function over the seen classes for training:

$$\mathcal{L}_{CE} = -\sum_{i=1}^{N}\sum_{j=1}^{C_s} t_{i,j}\log(s_{ij}) \tag{3.4}$$

where $s_{ij} = f(\psi(x_i), z_{c_j}; \theta)$ is the compatibility score of the image x_i and the j-th class, and $t_{i,j}$ is the truth value which takes 0 or 1 according to following condition:

$$t_{i,j} = \begin{cases} 1, & y_i = c_j \\ 0, & otherwise \end{cases} \tag{3.5}$$

However, Eq. 3.4 alone is not sufficient for modeling unseen classes. Therefore, we propose to add a new loss term called *contrastive loss term* which is computed over the unseen classes. It is important to note that, we are not able to access the samples

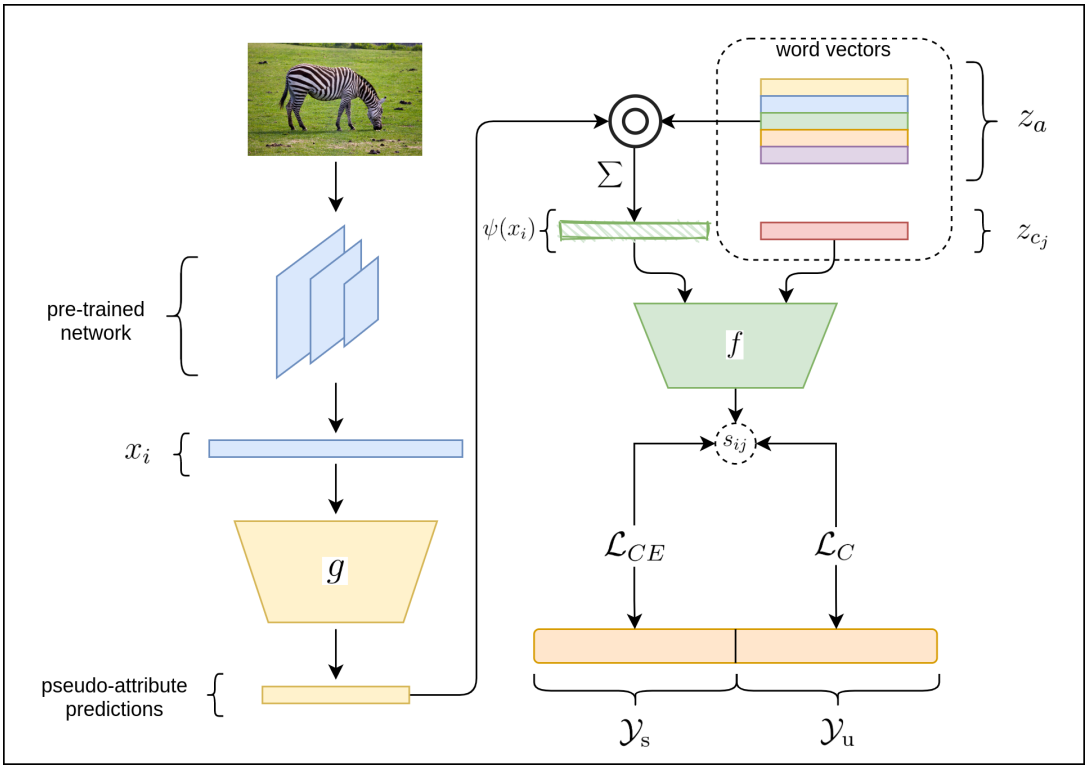

Figure 3.1: Illustration of the proposed approach.

which belongs to unseen classes. Thus, we utilize seen samples as well as similarities of the seen and unseen classes to enable contrastive learning:

$$\mathcal{L}_C = -\sum_{i=1}^{N} \sum_{j=C_s+1}^{C_s+C_u} t_{i,j} \log(s_{ij}) \tag{3.6}$$

where $t_{i,j}$ is again the truth value. However the true class label is y_i which is among the seen classes $\mathcal{Y}_\mathrm{s}$ and it is not possible to compute the $t_{i,j}$ with Eq. 3.5. In order to learn the unseen classes, we compute Eq. 3.6 as if the image x_i belongs to an unseen class. To do that we use class similarities of the seen and unseen classes and find the most similar unseen class to each seen class y_i:

$$t_{i,j} = \begin{cases} 1, & j = \mathrm{argmax}_{c \in \mathcal{Y}_\mathrm{u}}(\mathrm{cossim}(y_i, c)) \\ 0, & otherwise \end{cases} \tag{3.7}$$

where similarity function $\mathrm{cossim}()$ is the cosine similarity of the word vectors and

computed as follows:

$$\text{cossim}(y_i, c) = \frac{z_{y_i}^T z_c}{\|z_{y_i}\| \|z_c\|} \tag{3.8}$$

Finally, we formulate the total loss by integrating cross-entropy loss computed with Eq. 3.4 and the contrastive loss computed with Eq. 3.6:

$$\mathcal{L} = \mathcal{L}_{CE} + \alpha \mathcal{L}_C \tag{3.9}$$

where α is the hyper-parameter to be learned which aims to adjust the contrastive loss term.

The overall approach is summarized in Figure 3.1. For a given image, we obtain image features from a pre-trained network. The image features are forwarded to the function g in order to obtain pseudo-attribute predictions. After that, pseudo-attribute based image embedding $\psi(x_i)$ is obtained by computing the weighted sum of the attribute name embeddings with corresponding pseudo-attribute predictions. The final score s_{ij} is obtained through the function f by comparing the name embedding of the class j and the pseudo-attribute based embedding of the image i. $\mathcal{L}_C$ is computed over the unseen classes and $\mathcal{L}_{CE}$ is computed over the seen classes.

It is important to notice that the contrastive loss term can be used with any other method. In Chapter 4, we present extensive experiments to show the efficiency of our approach and the novel loss term.

CHAPTER 4

EXPERIMENTS

We conduct extensive experiments to measure the performance of our approach. In this section, first we explain our experimental setup in detail and then we present our experimental results.

4.1 Experimental Setup

4.1.1 Dataset

We experimented our proposed approach on five widely used GZSL benchmarks that are proposed by [17]: AWA1, AWA2, APY, CUB and SUN. As it is stated in [17], some test classes of these datasets are among the 1K classes of ImageNet and image features are extracted through pre-trained Deep Neural Networks (DNNs) such as ResNet [54] which are trained with ImageNet 1K dataset. It is against the nature of ZSL to have test samples whose class labels are known by the pre-trained networks. [17] emphasizes that accuracy of the overlapping test classes are higher than the other test classes. Therefore, [17] proposes new dataset splits called proposed splits (PS) ensuring that none of test classes are included in ImageNet. In addition, [17] a new dataset called Animals with Attributes2 (AwA2) to enable vision research on the AwA dataset since images of the original AwA (or AwA1) dataset are not publicly available. In our experiments, we use only proposed splits (PS).

Table 4.1 shows the details of the used datasets on our experiments. Attribute Pascal and Yahoo (aPY) [23] is a small scale and coarse grained dataset that contains 18,627 images, 20 classes from aPascal and 12 classes from aYahoo. Animals with

Dataset	Granularity	Image	Att	Train + Validation	Test	All
APY	coarse	15339	64	15 + 5	12	32
AWA1	coarse	30475	85	27 + 13	10	50
AWA2	coarse	37322	85	27 + 13	10	50
CUB	fine	11788	312	100 + 50	50	200
SUN	fine	14340	102	580 + 65	72	717

Table 4.1: Statistics of the used datasets in experiments.

Attributes (AwA) [11] is a medium scale and coarse grained dataset containing 30,475 images from 50 animal classes. AwA2 which has the characteristics of AwA1 contains 37,322 images. Caltech-UCSD-Birds-200-2011 (CUB) [21] is a medium scale and fine grained dataset that contains 11,788 images from 200 different types of birds. SUN Attribute (SUN) [22] is a medium scale and fine grained dataset that contains 14,340 images from 717 different types of scenes. Following [17] we use 15, 27, 27, 100, 580 classes for training, 5, 13, 13, 50, 65 classes for validation and 12, 10, 10, 50, 72 classes for test respectively for aPY, AwA1, AwA2, CUB and SUN datasets.

Table 4.1 also shows the number of attributes for each dataset. Each object class in aPY dataset is annotated with 64 attributes such as "Ear", "Nose", "Eye", "Plastic", "Wood", "Cloth" etc. Each class in AwA1 and AwA2 datasets is annotated with 85 attributes such as "stripes", "paws", "tail" etc. CUB and SUN datasets have more per-class attribute annotations according to aPY and AwA, they have 102 and 312 respectively. Note that we only use the attribute name embeddings instead of attributes itself.

4.1.2 Attribute name embedding

In our experiments, first we need to obtain the attribute name embeddings in order to investigate use of attribute names improves the recognition. We opt to use uncased 300-dimensional Global Vectors for Word Representation (GloVe) [45] which is trained on common crawl data. The reason we used uncased representations is that having a capital letter in the name of the attributes has no significant importance.

In general, we use the vector itself if the attribute name consists of only one word, we

use the average of the vectors of the words contained in the attribute name otherwise. However for some cases we need to take additional actions. For example; all attribute names of the CUB [21] dataset start with the word "has". We discarded "has" when computing the average of the word vectors. Another exception is that we could not directly obtain the vectors of some attribute names of AwA [11] and we had to split the words manually. For example; "chewteeth" is the original name of the attribute in AwA, however we had to split it as "chew+teeth" then find each word vector and take the average of it. Also some attribute names contain additional information in parentheses such as "has_size::large_(16_-_32_in)" which, in our opinion, may cause noise. Thus we removed the additional information from the original name.

4.1.3 Class embeddings

In general, either per-class continuous attribute annotations or class name word representations such as Word2Vec [46] and GloVe [45] are used as class embeddings for ZLS. However, attributes are preferred more than the word representations since attributes contain richer visual information which leads to higher experimental results. Especially, in recent years, most of the works use attributes provided by [17]. However, it requires very significant human effort to annotate all the images which is also not suitable for real world problems. Thus, we believe that using class name embeddings is more valuable for ZSL tasks.

In order to obtain class name embeddings, we followed the steps as we obtain the attribute name embeddings. If the class name consists of only one word, we directly get the GloVe [45] of the word, otherwise we compute the average of the vectors of each word in the class name. Again, we opt to use uncased and 300-dimensional GloVe [45] to obtain class name embeddings.

4.1.4 Evaluation protocol

In our experiments, we measure the top-1 mean class accuracy in order to avoid misleading results by most populated classes. Therefore, we take the sum of the per-class

top-1 accuracy and we compute the average by dividing the number of classes.

$$acc_c = \frac{1}{|C|} \sum_{c=1}^{|C|} \frac{\text{\# of correctly predicted samples in } c}{\text{\# of samples in } c} \tag{4.1}$$

In ZSL, at the evaluation time, the model is restricted to unseen classes to assign the samples, however in GZSL setting the model includes both the training classes and the test classes to assign the samples. Following [17], we evaluate our experimental results by computing the harmonic mean which forces the both the unseen test accuracy acc_u and seen test accuracy acc_s to be high.

$$H = \frac{2 \times acc_s \times acc_u}{acc_s + acc_u} \tag{4.2}$$

4.1.5 Implementation details

We implmented our approach using PyTorch [55]. For function g, we use a fully-connected layer to transform image features to the same dimension of mid-level representations such as pseudo-attributes. For function f, we use two distinct fully-connected layers to linearly transform pseudo-attribute based image embeddings and the class name embeddings to a common space without changing the dimensions. We use hyperbolic tangent functions as output of the all fully-connected layers. We train our model by optimizing the Eq. 3.9. We use ADAM [56] for stochastic optimization. We fine-tune the hyper-parameters on validation set provided by [17]. For fair comparison, we selected the hyper-parameters separately for each individual experiment. We selected the learning rate in $[10^{-5}, 10^{-2}]$, weight decay in $[0, 0.9]$, batch size in $[128, 2048]$ and α in $[10^{-3}, 1]$. We run all of the experiments for 5 times with exactly same settings and reported the average values in the experimental results.

4.1.6 Baseline methods

We compare our approach with state of the art methods reported by [17]: DAP [11], IAP [11], ConSE [19], CMT [29], SSE [31], LATEM [28], ALE [24], DEVISE [25],

SJE [26], ESZSL [27], SYNC [57], SAE [58] and GFZSL [33]. However, results of these methods in [17] have been reported by using attributes as class embeddings. For fair comparison, we need to have word embedding based results. Since label embedding is shown to be a versatile and stable approach [17], we use ALE [24] as our baseline method. We reproduce ALE results with class name word vectors.

4.2 Experimental Results

4.2.1 Sanity check

We conducted extensive experiments on five benchmarks to measure the performance of our approach. For fair comparison we use the image features provided by [17] however the baseline methods are evaluated by using attributes whereas our method is evaluated using unsupervised class name word representations. Therefore we opt to re-evaluate ALE [24] method with class name embeddings. However, for sanity check, we re-evaluate ALE [24] method with attributes and compare our results with the one that is reported by [17]. Table 4.2 shows that our reproduced results are close to the original one. The ALE* is the re-evaluated results of the ALE with attributes. For all of the datasets, ZSL results of ALE and ALE* are consistent and the difference between these two methods is less than 4.3%. On the other hand, we compare H scores of the GZSL results and the performance difference between ALE and ALE* is less than 1.7% for all the datasets except SUN. The H score of our implementation for the SUN dataset is greater than the original one by 4.2%. The re-evaluated results for APY and CUB are slightly better, but slightly worse for AWA1 and AWA2. It is obvious that the H scores of the two methods are also consistent. In our opinion, small differences in performances is due to the small implementation differences such as optimizer and also the hyperparameter tuning.

4.2.2 Generalized zero-shot learning results

In this section we present our experimental results in GZSL setting where a test sample may belong to seen or unseen classes. Here, we compare our method with the

Dataset	Method	ZSL	GZSL		
			u	s	H
APY	ALE	**39.7**	4.6	73.7	8.7
	ALE*	35.4 (-4.3)	5.1	62.7	**9.4** (+0.7)
AWA1	ALE	59.9	16.8	76.1	**27.5**
	ALE*	**61.4** (+1.5)	15.5	76.3	25.8 (-1.7)
AWA2	ALE	**62.5**	14.0	81.8	**23.9**
	ALE*	61.7 (-0.8)	13.6	65.8	22.5 (-1.4)
CUB	ALE	54.9	23.7	62.8	34.4
	ALE*	**56.3** (+1.4)	24.6	64.4	**35.6** (+1.2)
SUN	ALE	58.1	21.8	33.1	26.3
	ALE*	**62.4** (+4.3)	25.2	38.8	**30.5** (+4.2)

Table 4.2: Sanity check for the implementation of ALE

baseline method ALE [24] and methods that use word vectors as class embeddings. Table 4.3 shows the detailed results of our experiments where **u** is the top-1 accuracy of the test samples belongs to unseen classes, **s** is the top-1 accuracy of the test samples belongs to seen classes and **H** is the harmonic mean of the **u** and **s** which evaluates the performance of the method. **s** and **u** is computed using Eq. 4.1 and **H** is computed using Eq. 4.2. **SS** stands for semantic space that is used by models and based on attributes (A) or word vectors (W). According to results in table 4.3, using class name word vectors instead of attributes as class embeddings on ALE* method dramatically decreases the performance of the method in H score for all dataset except APY. It is obvious that using class name word vectors is more challenging since class name representations don't have visual clues as attributes have.

It is important to note that, proposed contrastive loss term is applicable to any other model. In our experiments we also apply contrastive loss to the ALE method. Table 4.3 also shows that using the proposed loss term significantly improves the performance of the model for the unseen classes. For APY dataset, ALE*-C obtains 27.2% in H score that is the second highest score among all of other methods and improves the word vectors based ALE* by 12.6%. For AWA1, AWA2 and CUB datasets, ALE*-C greatly improves the word vector based ALE* by 13.3%, 18.2% and 4.2% respectively. However, for the SUN dataset, proposed contrastive loss term does not improve the performance of the word vector based ALE* since the 0.4% improvement on test samples which belongs to unseen classes is not significant enough

Method	SS	APY			AWA1			AWA2			CUB			SUN		
		u	s	H	u	s	H	u	s	H	u	s	H	u	s	H
CAPD [44]	W	-	-	-	**43.2**	46.4	**44.7**	-	-	-	**30.3**	7.2	11.7	-	-	-
A2CN* [20]	A+W	9.2	25.8	13.5	9.4	34.6	14.7	8.4	20.5	11.2	-	-	-	-	-	-
ALE* [24]	A	5.1	62.7	9.4	15.5	76.3	25.8	13.6	65.8	22.5	24.6	64.4	35.6	25.2	38.8	30.5
ALE* [24]	W	8.6	66.0	14.6	4.5	**84.5**	8.5	3.0	**89.0**	5.8	3.7	**59.8**	7.1	6.1	**35.3**	10.4
ALE*-C	W	17.6	60.3	27.2	17.8	28.9	21.8	18.3	37.0	24.0	6.7	35.9	11.3	6.5	19.6	9.7
PANE	W	7.6	68.5	13.6	3.8	66.4	7.3	2.4	80.2	4.5	3.2	38.6	6.0	6.8	22.8	10.5
PANE-C_{SC}	W	20.5	**70.8**	31.8	21.3	62.7	31.7	18.3	56.7	27.6	10.8	19.5	13.9	8.3	31.0	13.1
PANE-C	W	**21.3**	66.6	**32.3**	24.6	60.4	34.8	**23.4**	59.4	**33.2**	9.7	27.3	**14.2**	**19.4**	18.2	**18.8**

Table 4.3: Comparison with word vector based methods and baseline method ALE [24]. Best results are represented as bold. "*" means that the method is implemented by us. "-" indicates that no result reported in the corresponding paper. "-C" indicates that proposed contrastive loss is applied. A2CN required attribute-based seen class definitions at training time, therefore, categorized as $A+W$.

to make a substantial improvement in the harmonic mean score.

On the other hand, compared to word vector based ALE* our method PANE without contrastive loss term performs slightly worse. The difference between PANE and word vector based ALE* is less than 1.3% for all the datasets. However, when the contrastive loss is applied to our model, PANE-C outperforms all other word vector based methods except CAPD [44] on AWA1 dataset. For APY, AWA2, CUB and SUN datasets, PANE-C obtains 32.3%, 33.2%, 14.2% and 18.8% respectively in H that are the highest scores among all of other word vector based methods. These results are greater 17.7%, 27.4%, 7.1% and 8.4% respectively than the word vector based ALE*. For AWA1 dataset, CAPD [44] performs better than our approach by 9.9%. However, For CUB dataset our model PANE-C performs better than CAPD [44] by 2.5%. It is obvious that our approach significantly improves the baseline methods especially for the coarse-grained datasets.

In addition, we compare our results with Demirel *et al.*'s approach [20] which is one of the closest work to our approach. Our model formulation is inspired from Demirel *et al.*'s model, but differs in many details as discussed in Chapter 2. First of all, for fair comparison, we implemented the Demirel *et al.*'s approach A2CN [20] since the reported results are not evaluated in GZSL setting. For the implementation, we use

image features provided by [17] and same Glove [45] word vectors as in our own experiments.

A2CN [20] is a two-step approach; first it trains binary attribute classifiers to recognize the attributes for a given image. Then it learns a transformation function using predicted attributes, attribute name embeddings and class name embeddings. Therefore, firstly, we trained attribute classifiers using training samples. After that, we implemented the model as in explained in the paper. Table 4.3 presents the comparison of our method with the A2CN [20]. Our approach without contrastive loss term, PANE, performs close to A2CN [20] method for only APY dataset. For AWA1 and AWA2 datasets the resulsts of PANE is under the performance of the A2CN [20] by 7.4% and 6.7% respectively. However, our method PANE-C which uses additional contrastive loss greatly outperforms the A2CN [20].

Additionally, our novel contrastive loss term is inspired from the TCN [18] approach which proposes a novel way that is similar to *sparse coding (sc)* in order to learn similarities between the seen and unseen classes. Unlike TCN [18], in this work, we directly use cosine similarity on the word vectors of the class names to obtain the class similarities. In order to show the effectiveness of the proposed contrastive loss, we apply class similarity learning method in TCN [18] to our approach and present the experimental results in Table 4.3 as PANE-C_{SC}. We would like to emphasize that, in these experiments we change only the obtaining method of the class similarities, we still use our own contrastive loss function to optimize the model. For APY and CUB dataset, PANE-C_{SC} obtains 31.8% and 13.9% respectively in H score which are quite close to results of PANE-C. For AWA1, AWA2, and SUN datasets PANE-C obtains 3.1%, 5.6%, and 5.7% greater results than the PANE-C_{SC}. Experimental results shows that using cosine similarity is clearly more effective than the learning method in TCN [18] for our proposed approach.

4.2.3 Effect of alpha

In this section we explore how the proposed loss term affects the unseen and seen test accuracies. Thus, we experimented by fixing all the hyper-parameters and selecting the $\alpha \in [0, 1]$ on the validation of the APY. Figure 4.1 shows that the unseen test

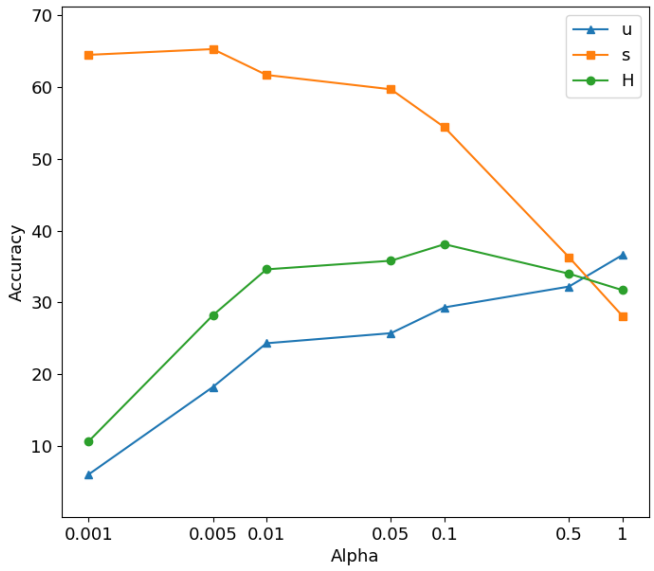

Figure 4.1: Effect of α learning unseen classes of APY on validation set. Graph is plotted as base-10 log scale for the x-axis.

accuracy, seen test accuracy and the H score for different values of α. It can be seen that unseen test accuracy and the H score continually increases until $\alpha = 0.1$. After that, for the value of 0.5, although the unseen test accuracy slightly increases the H score does not improve due to the dramatic decrease in seen test accuracy.

In our approach, the model learns unseen classes by using seen samples thanks to the contrastive loss term. However, in our opinion, after a value for α the contrastive loss term forces the model to assign seen test samples to unseen classes which causes a dramatic decrease in seen test accuracy. Here, we experimented only for the APY dataset but we observed the same situation for the other datasets during hyperparameter tuning.

4.2.4 Effect of attribute names

In order to improve recognition, we incorporate the attribute name word vectors into the proposed model. In this part, we measure how much our approach utilizes original attribute names in contrast to arbitrary ones. To do this, thanks to the flexibility

Dataset	Used att. names	GZSL		
		u	s	H
AWA2	AWA2	**23.4**	59.4	**33.2**
AWA2	CUB	18.7	**65.5**	28.8

Table 4.4: Experimenting using different word vectors instead of original attributes names.

of our model, we change the output dimension of the attribute prediction function g and use arbitrary word vectors instead of the original attribute names. For this experiment, we use AWA2 [17] dataset and take the attribute name vectors from another unrelated dataset which is CUB [21]. AWA2 has mostly common attributes among all animals such as *black, white, big, small* etc., whereas CUB dataset includes more rare attributes that are specific to the birds such as "wing_color::brown", "bill_shape::dagger" etc. Therefore, the test samples from AWA2 dataset will not include attributes of the CUB.

Table 4.4 shows that using word vectors that are unrelated with the dataset degrades the performance of the model. In other words, our approach utilizes attribute names due to the performance increase when original attribute names are used and reaches the best performance when used with the contrastive loss term. Results in Table 4.3 shows that using only attribute names is not enough however the model reaches the best performance when contrastive loss is applied. The model PANE which predicts the pseudo-attributes and tries to utilizes attribute names to improve recognition does not perform well, however the model PANE-C reaches the best performance among others.

4.2.5 Combination of class names

One of the advantages of our model is flexibility which allows us to make model variants as in the Section 4.2.4. In this part, we change output dimension of the function g and instead of predicting pseudo-attributes we predict the class labels. We replace z_a with class name word embeddings in Eq. 3.1. Thus, our model becomes similar

Method	Dataset	Training semantic emb.	GZSL		
			u	s	H
PANE-C	AWA2	Seen	24.1	56.7	33.5
PANE-C	AWA2	Unseen	22.8	50.4	31.1
PANE-C	AWA2	All	24.6	57.8	34.4
PANE-C	AWA2	PCA Components, n=30	**26.5**	49.9	**34.5**
PANE-C	AWA2	PCA Components, n=50	25.4	45.5	32.6
ConSE [19]	AWA2	Seen	0.5	**90.6**	1.0

Table 4.5: Experimenting using class name word vectors and PCA components instead of attributes names.

to ConSE [19]. ConSE predicts the probability of belonging to the each seen class for a given image. Then, it takes top T seen class probabilities and compute the convex combination of the semantic embeddings by weighting with the probabilities in order to project the image into the semantic space. However, usage of probabilistic approach in ConSE [19] makes the model to generate only positive combination of semantic embeddings. Unlike ConSE, [19] our model is able to generate combination of the class names by not only adding but also subtracting. Additionally, our model is trainable in end-to-end manner, therefore, allows using arbitrary word embeddings which is not possible with ConSE [19]. Also, different from ConSE, we use a generalization loss function to learn unseen classes.

Table 4.5 shows the comparison of our approach with ConSE [19]. It is important to emphasize that the result of ConSE [19], reported by [17] and obtained by using attributes as semantic embeddings. On the other hand, for this experiment we use class word vectors as semantic embeddings. Results in Table 4.5 shows that all variants of our model outperforms the ConSE [19] with a high margin. It is a quite expected result that our model reaches the best performance when combination of semantic embeddings is computed by using all class names embeddings.

In addition, we replaced the attribute name embeddings with PCA basis vectors that are obtained by applying principal component analysis on the embeddings of all class names. The motivation here is to use orthonormal basis vectors, which can be used

to linearly reconstruct class name embeddings, as pseudo-attribute names. For this experiment, we set the output dimensionality of the function g to the number of basis vectors. Then, we compute the weighted sum with the corresponding prediction by replacing z_a with PCA components in Eq 3.1. Thus, we obtain a 300-dimensional vector to estimate relevance score of the given image with the classes through compatibility function f in Eq 3.2. We present the results of these experiments in Table 4.5. We observe that using PCA components produces quite stable results in H score. PANE-C with 30 PCA components obtains 1.9% greater result than the PANE-C with 50 PCA components, probably due to focusing on the most valuable PCA basis vectors, and the reduction in the overall model complexity.

4.2.6 Comparison with state of the art methods

In this part, we compare our GZSL results against attribute based state-of-the-art methods. Since generative approaches are beyond the scope of this work, we take only the non-generative approaches into account and compare with them. Table 4.6 presents the results of the recent state-of-the-art methods. Upper side of the table shows the results that are reported by [17]. The bottom line is the our proposed approach with contrastive loss.

For APY, AWA1 and AWA2 datasets, PANE-C obtains 32.3%, 34.8% and 33.2% respectively in H that are the highest scores among all of other methods reported by [17] (methods that are at upper side of the table). These results are greater 19.0%, 7.3% and 5.4% respectively than the best methods reported by [17]. For CUB and SUN and datasets our model obtains 14.2% and 18.8% which are better results than the most of the methods reported by [17].

However, our model is not able to yield the state-of-the-art results as the other attribute based methods. For APY, AWA1 and AWA2 datasets, CAPD [44], TCN [18] and DAZLE [42] produces the best results as 37.0%, 60.0% and 67.1% respectively. For SUN and CUB datasets the best results obtained by APN [41] as 67.2% and 37.6% respectively. These methods outperforms our approach with a high margin. However, it is important to emphasize that these methods uses attributes as class embeddings which is an easier task since attributes includes richer visual information. In our

Method	SS	APY			AWA1			AWA2			CUB			SUN		
		u	**s**	**H**	**u**	**s**	**H**	**u**	**s**	**H**	**u**	**s**	**H**	**u**	**s**	**H**
DAP [11]	A	4.8	78.3	9.0	0.0	88.7	0.0	0.0	84.7	0.0	1.7	67.9	3.3	4.2	25.1	7.2
IAP [11]	A	5.7	65.6	10.4	2.1	78.2	4.1	0.9	87.6	1.8	0.2	**72.8**	0.4	1.0	37.8	1.8
ConSE [19]	A	0.0	**91.2**	0.0	0.4	88.6	0.8	0.5	90.6	1.0	1.6	72.2	3.1	6.8	39.9	11.6
CMT [29]	A	1.4	85.2	2.8	0.9	87.6	1.8	0.5	90.0	1.0	7.2	49.8	12.6	8.1	21.8	11.8
SSE [31]	A	0.2	78.9	0.4	7.0	80.5	12.9	8.1	82.5	14.8	8.5	46.9	14.4	2.1	36.4	4.0
LATEM [28]	A	0.1	73.0	0.2	7.3	71.7	13.3	11.5	77.3	20.0	15.2	57.3	24.0	14.7	28.8	19.5
DEVISE [25]	A	4.9	76.9	9.2	13.4	68.7	22.4	17.1	74.7	27.8	23.8	53.0	32.8	16.9	27.4	20.9
SJE [26]	A	3.7	55.7	6.9	11.3	74.6	19.6	8.0	73.9	14.4	23.5	59.2	33.6	14.7	30.5	19.8
ESZSL [27]	A	2.4	70.1	4.6	6.6	75.6	12.1	5.9	77.8	11.0	12.6	63.8	21.0	11.0	27.9	15.8
SYNC [57]	A	7.7	66.3	13.3	8.9	87.3	16.2	10.0	90.5	18.0	11.5	70.9	19.8	7.9	**43.3**	13.4
SAE [58]	A	0.4	80.9	0.9	1.8	77.1	3.5	1.1	82.2	2.2	7.8	54.0	13.6	8.8	18.0	11.8
GFZSL [33]	A	0.0	83.3	0.0	1.8	80.3	3.5	2.5	80.1	4.8	0.0	45.7	0.0	0.0	39.6	0.0
ALE [24]	A	4.6	73.7	8.7	16.8	76.1	27.5	14.0	81.8	23.9	23.7	62.8	34.4	21.8	33.1	26.3
CAPD [44]	A	**26.8**	59.5	**37.0**	45.2	68.6	54.5	-	-	-	44.9	41.7	43.3	35.8	27.8	31.3
DCN [51]	A	14.2	75.0	23.9	25.5	84.2	39.1	-	-	-	28.4	60.7	38.7	25.5	37.0	30.2
RNet [14]	A	-	-	-	31.4	**91.3**	46.7	30.0	**93.4**	45.3	38.1	61.4	47.0	-	-	-
TCN [18]	A	24.1	64.0	35.1	**49.4**	76.5	**60.0**	**61.2**	65.8	63.4	52.6	52.0	52.3	31.2	37.3	34.0
APN [41]	A	-	-	-	-	-	-	56.5	78.0	65.5	**65.3**	69.3	**67.2**	41.9	34.0	**37.6**
DAZLE [42]	A	-	-	-	-	-	-	60.3	75.7	**67.1**	56.7	59.6	58.1	**52.3**	24.3	33.2
PANE-C	W	21.3	66.6	32.3	24.6	60.4	34.8	23.4	59.4	33.2	9.7	27.3	14.2	19.4	18.2	18.8

Table 4.6: Comparison with recent state-of-the-art models. Best results are represented as bold. "-" indicates the that no result reported in the related paper.

opinion, using class name word representations is worth more to explore and feasible to real world problems.

In this chapter, we have explained our experimental setups and discussed our experiments. In Chapter 5, we summarize and conclude the thesis and discuss the future works.

CHAPTER 5

CONCLUSION AND FUTURE WORK

In this chapter, we will conclude the thesis and discuss the future works that can be worth to explore as extensions of this work.

5.1 Conclusion

In this thesis, we target the zero-shot learning problem which has gained popularity in recent years, since ZSL is a promising approach to tackle the limits of supervised recognition techniques and more feasible to solve real world problems.

In ZSL, many recent approaches focus on recognizing objects by their attributes which requires annotations from experts. This requirement is a high-cost and error-prone issue and makes the ZSL approaches not applicable to real world problems. In this thesis, therefore, we re-explore using class name embeddings that can be obtained from any language model. Our approach eliminates the need of attributes instead explores mid-level constructs such as pseudo-attributes by utilizing attribute name embeddings. In addition, in this thesis, we explore the use of contrastive loss function for recognizing objects that belong to unseen classes. To this end, we propose a contrastive loss term which is computed using class similarities of the seen and unseen classes.

Experimental results on five benchmarks in GZSL settings shows the capability of our proposed approach. Although the results are obtained using class name word vectors instead of attributes, which makes the task more challenging, our proposed approach achieves state-of-the-art results. In addition, the experimental results present that

using our proposed contrastive loss term in GZSL greatly improves the performance of the model and enables the model to learn the unseen classes.

5.2 Future Work

In this thesis, we propose a simple yet effective model. One of the advantages of our proposed model is the flexibility to change and generate new variants. A new direction that might be worth exploring is alternative formulations for better utilizing name based pseudo-attributes. One of our experiments we replaced the original attribute names with unrelated ones and showed that the performance of the model degrades. In our opinion, therefore, pseudo-attributes are of great importance to improve the recognition.

In addition, another future work that might be worth exploring is using different language models. In this work, we use GloVe representations of the class names to compute the class similarities which might cause suboptimal results, since word representations of the similar class names such as *blue whale* and *humpback whale* might not be close enough to each other in word space. Also, using different similarity functions is another promising work. In fact, learning the class similarities with the model itself in an end-to-end manner is worth exploring.

REFERENCES

[1] K. P. Bennett and A. Demiriz, "Semi-supervised support vector machines," in *Advances in Neural Information Processing Systems 11, [NIPS Conference, Denver, Colorado, USA, November 30 - December 5, 1998]* (M. J. Kearns, S. A. Solla, and D. A. Cohn, eds.), pp. 368–374, The MIT Press, 1998.

[2] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling, "Semi-supervised learning with deep generative models," in *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada* (Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, eds.), pp. 3581–3589, 2014.

[3] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2017.

[4] Y. Fu, T. M. Hospedales, T. Xiang, Z. Fu, and S. Gong, "Transductive multi-view embedding for zero-shot recognition and annotation," in *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part II* (D. J. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, eds.), vol. 8690 of *Lecture Notes in Computer Science*, pp. 584–599, Springer, 2014.

[5] J. Song, C. Shen, Y. Yang, Y. Liu, and M. Song, "Transductive unbiased embedding for zero-shot learning," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 1024–1033, IEEE Computer Society, 2018.

[6] Z. Wan, D. Chen, Y. Li, X. Yan, J. Zhang, Y. Yu, and J. Liao, "Transductive zero-shot learning with visual structure constraint," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC,*

Canada (H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, eds.), pp. 9972–9982, 2019.

[7] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," in *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, vol. 119 of *Proceedings of Machine Learning Research*, pp. 1597–1607, PMLR, 2020.

[8] C. Doersch and A. Zisserman, "Multi-task self-supervised visual learning," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pp. 2070–2079, IEEE Computer Society, 2017.

[9] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pp. 1422–1430, IEEE Computer Society, 2015.

[10] K. He, H. Fan, Y. Wu, S. Xie, and R. B. Girshick, "Momentum contrast for unsupervised visual representation learning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pp. 9726–9735, IEEE, 2020.

[11] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 3, pp. 453–465, 2014.

[12] D. Jayaraman and K. Grauman, "Zero-shot recognition with unreliable attributes," in *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada* (Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, eds.), pp. 3464–3472, 2014.

[13] P. Kankuekul, A. Kawewong, S. Tangruamsub, and O. Hasegawa, "Online incremental attribute-based zero-shot learning," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3657–3664, 2012.

[14] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 1199–1208, IEEE Computer Society, 2018.

[15] S. Ravi and H. Larochelle, "Optimization as a model for few-shot learning," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, OpenReview.net, 2017.

[16] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA* (I. Guyon, U. von Luxburg, S. Bengio, H. M. Wallach, R. Fergus, S. V. N. Vishwanathan, and R. Garnett, eds.), pp. 4077–4087, 2017.

[17] Y. Xian, C. H. Lampert, B. Schiele, and Z. Akata, "Zero-shot learning - A comprehensive evaluation of the good, the bad and the ugly," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 9, pp. 2251–2265, 2019.

[18] H. Jiang, R. Wang, S. Shan, and X. Chen, "Transferable contrastive network for generalized zero-shot learning," in *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*, pp. 9764–9773, IEEE, 2019.

[19] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. Corrado, and J. Dean, "Zero-shot learning by convex combination of semantic embeddings," in *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings* (Y. Bengio and Y. LeCun, eds.), 2014.

[20] B. Demirel, R. G. Cinbis, and N. Ikizler-Cinbis, "Attributes2classname: A discriminative model for attribute-based unsupervised zero-shot learning," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pp. 1241–1250, IEEE Computer Society, 2017.

[21] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD Birds-200-2011 Dataset," Tech. Rep. CNS-TR-2011-001, California Institute of Technology, 2011.

[22] G. Patterson and J. Hays, "Sun attribute database: Discovering, annotating, and recognizing scene attributes," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2751–2758, 2012.

[23] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1778–1785, 2009.

[24] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, "Label-embedding for attribute-based classification," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 819–826, 2013.

[25] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. A. Ranzato, and T. Mikolov, "Devise: A deep visual-semantic embedding model," in *Advances in Neural Information Processing Systems* (C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, eds.), vol. 26, pp. 2121–2129, Curran Associates, Inc., 2013.

[26] Z. Akata, S. Reed, D. Walter, H. Lee, and B. Schiele, "Evaluation of output embeddings for fine-grained image classification," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2927–2936, 2015.

[27] B. Romera-Paredes and P. Torr, "An embarrassingly simple approach to zero-shot learning," in *Proceedings of the 32nd International Conference on Machine Learning* (F. Bach and D. Blei, eds.), vol. 37 of *Proceedings of Machine Learning Research*, (Lille, France), pp. 2152–2161, PMLR, 07–09 Jul 2015.

[28] Y. Xian, Z. Akata, G. Sharma, Q. N. Nguyen, M. Hein, and B. Schiele, "Latent embeddings for zero-shot classification," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 69–77, IEEE Computer Society, 2016.

[29] R. Socher, M. Ganjoo, C. D. Manning, and A. Ng, "Zero-shot learning through cross-modal transfer," in *Advances in Neural Information Processing Systems*

(C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, eds.), vol. 26, pp. 935–943, Curran Associates, Inc., 2013.

[30] Z. Zhang and V. Saligrama, "Zero-shot learning via joint latent similarity embedding," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6034–6042, 2016.

[31] Z. Zhang and V. Saligrama, "Zero-shot learning via semantic similarity embedding," in *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pp. 4166–4174, IEEE Computer Society, 2015.

[32] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell, "Zero-shot learning with semantic output codes," in *Advances in Neural Information Processing Systems* (Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, eds.), vol. 22, pp. 1410–1418, Curran Associates, Inc., 2009.

[33] V. K. Verma and P. Rai, "A simple exponential family framework for zero-shot learning," in *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18-22, 2017, Proceedings, Part II* (M. Ceci, J. Hollmén, L. Todorovski, C. Vens, and S. Dzeroski, eds.), vol. 10535 of *Lecture Notes in Computer Science*, pp. 792–808, Springer, 2017.

[34] M. B. Sariyildiz and R. G. Cinbis, "Gradient matching generative networks for zero-shot learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2168–2178, 2019.

[35] A. Pambala, T. Dutta, and S. Biswas, "Generative model with semantic embedding and integrated classifier for generalized zero-shot learning," in *The IEEE Winter Conference on Applications of Computer Vision*, pp. 1237–1246, 2020.

[36] E. Schönfeld, S. Ebrahimi, S. Sinha, T. Darrell, and Z. Akata, "Generalized zero- and few-shot learning via aligned variational autoencoders," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 8247–8255, Computer Vision Foundation / IEEE, 2019.

[37] Y. Xian, T. Lorenz, B. Schiele, and Z. Akata, "Feature generating networks for zero-shot learning," in *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pp. 5542–5551, IEEE Computer Society, 2018.

[38] Y. Xian, S. Sharma, B. Schiele, and Z. Akata, "F-VAEGAN-D2: A feature generating framework for any-shot learning," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pp. 10275–10284, Computer Vision Foundation / IEEE, 2019.

[39] S. Deutsch, A. L. Bertozzi, and S. Soatto, "Zero shot learning with the isoperimetric loss," in *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pp. 10704–10712, AAAI Press, 2020.

[40] Y. Li, D. Wang, H. Hu, Y. Lin, and Y. Zhuang, "Zero-shot recognition using dual visual-semantic mapping paths," in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pp. 5207–5215, IEEE Computer Society, 2017.

[41] W. Xu, Y. Xian, J. Wang, B. Schiele, and Z. Akata, "Attribute prototype network for zero-shot learning," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual* (H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, eds.), 2020.

[42] D. Huynh and E. 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 (CVPR)*, June 2020.

[43] S. Min, H. Yao, H. Xie, C. Wang, Z. Zha, and Y. Zhang, "Domain-aware visual bias eliminating for generalized zero-shot learning," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pp. 12661–12670, IEEE, 2020.

[44] S. Rahman, S. H. Khan, and F. Porikli, "A unified approach for conventional zero-shot, generalized zero-shot, and few-shot learning," *IEEE Trans. Image Process.*, vol. 27, no. 11, pp. 5652–5667, 2018.

[45] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014.

[46] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States* (C. J. C. Burges, L. Bottou, Z. Ghahramani, and K. Q. Weinberger, eds.), pp. 3111–3119, 2013.

[47] R. Qiao, L. Liu, C. Shen, and A. van den Hengel, "Less is more: Zero-shot learning from online textual documents with noise suppression," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 2249–2257, IEEE Computer Society, 2016.

[48] L. J. Ba, K. Swersky, S. Fidler, and R. Salakhutdinov, "Predicting deep zero-shot convolutional neural networks using textual descriptions," in *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, pp. 4247–4255, IEEE Computer Society, 2015.

[49] M. Elhoseiny, B. Saleh, and A. Elgammal, "Write a classifier: Zero-shot learning using purely textual descriptions," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2584–2591, 2013.

[50] S. E. Reed, Z. Akata, H. Lee, and B. Schiele, "Learning deep representations of fine-grained visual descriptions," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 49–58, IEEE Computer Society, 2016.

[51] S. Liu, M. Long, J. Wang, and M. I. Jordan, "Generalized zero-shot learning with deep calibration network," in *Advances in Neural Information Processing*

Systems (S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, eds.), vol. 31, pp. 2005–2015, Curran Associates, Inc., 2018.

[52] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015.

[53] Y. Hu, G. Wen, A. Chapman, P. Yang, M. Luo, Y. Xu, D. Dai, and W. Hall, "Semantic graph-enhanced visual network for zero-shot learning," *arXiv preprint arXiv:2006.04648*, 2020.

[54] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 770–778, IEEE Computer Society, 2016.

[55] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32* (H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, eds.), pp. 8024–8035, Curran Associates, Inc., 2019.

[56] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (Y. Bengio and Y. LeCun, eds.), 2015.

[57] S. Changpinyo, W. Chao, B. Gong, and F. Sha, "Synthesized classifiers for zero-shot learning," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pp. 5327–5336, IEEE Computer Society, 2016.

[58] E. Kodirov, T. Xiang, and S. Gong, "Semantic autoencoder for zero-shot learning," in *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pp. 4447–4456, IEEE Computer Society, 2017.