Zero Shot Learning for Visual Object Recognition with Generative Models

by

Maunil Vyas

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved April 2020 by the
Graduate Supervisory Committee:

Sethuraman Panchanathan, Chair
Troy McDaniel
Hemanth Venkateswara

ARIZONA STATE UNIVERSITY

May 2020

ABSTRACT

Visual object recognition has achieved great success with advancements in deep learning technologies. Notably, the existing recognition models have gained human-level performance on many of the recognition tasks. However, these models are data hungry, and their performance is constrained by the amount of training data. Inspired by the human ability to recognize object categories based on textual descriptions of objects and previous visual knowledge, the research community has extensively pursued the area of zero-shot learning. In this area of research, machine vision models are trained to recognize object categories that are not observed during the training process. Zero-shot learning models leverage textual information to transfer visual knowledge from seen object categories in order to recognize unseen object categories.

Generative models have recently gained popularity as they synthesize unseen visual features and convert zero-shot learning into a classical supervised learning problem. These generative models are trained using seen classes and are expected to implicitly transfer the knowledge from seen to unseen classes. However, their performance is stymied by overfitting towards seen classes, which leads to substandard performance in generalized zero-shot learning. To address this concern, this dissertation proposes a novel generative model that leverages the semantic relationship between seen and unseen categories and explicitly performs knowledge transfer from seen categories to unseen categories. Experiments were conducted on several benchmark datasets to demonstrate the efficacy of the proposed model for both zero-shot learning and generalized zero-shot learning. The dissertation also provides a unique Student-Teacher based generative model for zero-shot learning and concludes with future research directions in this area.

DEDICATION

To my parents, Dr. Rohit Vyas and Hetal Vyas, for their endless love and support, and for giving me the freedom to shape my life.

ACKNOWLEDGEMENTS

I would like to extend my sincere thanks to Dr. Hemanth Venkateswara for his constant support, guidance, and motivation throughout my masters' journey at ASU. Honestly, I would have not reached this level without his encouragement and trust in my research ideas.

I would like to thank my committee members Dr. Sethuraman Panchanathan and Dr. Troy McDaniel, for their advice and feedback on my research work and helping me to become a part of the CUbiC lab at ASU.

I would also like to thank the members of the ASU CUbic lab, specifically Raghavendran, Badrinath, and Piyush for insightful discussions that helped me to improve my research work and provided the right boost. I cannot forget the support my roommates have provided to me throughout this research journey from reading my paper drafts to encouraging me to push myself further to create something new. Deep and Satyak, I will never forget the time I spent with you guys.

Last but not least, I would like to thank my parents, my sister, and my grandfather, for their constant love and support that helped me to stay focused on my work. No words or actions will ever be enough to do justice to everything they have done for me.

TABLE OF CONTENTS

Page

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

Consider the following discussion between a kindergarten teacher and her student.

Teacher: *Today we will learn about a new animal that roams the grasslands of Africa. It is called the Zebra.*

Student: *What does a Zebra look like ?*

Teacher: *It looks like a short white horse but has black stripes like a tiger.*

That description is nearly enough for the students to recognize a zebra the next time they see it. The students are able to take the verbal (textual) description and relate it to the visual understanding of a horse and a tiger and generate a zebra in their mind. In this work, I focus on a Zero-shot learning model that transfers knowledge from the text to the visual domain to learn and recognize previously unseen image categories.

Collecting and curating large labeled datasets for training deep neural networks is both labor-intensive and nearly impossible for many of the classification tasks, especially for the fine-grained categories in specific domains. Hence, it is desirable to create models that can mitigate these difficulties and learn not to rely on large labeled training sets. Inspired by the human ability to recognize object categories solely based on class descriptions and previous visual knowledge, the research community has extensively pursued the area of "Zero Shot Learning" (ZSL) (Lampert *et al.* (2013); Larochelle *et al.* (2008); Rohrbach *et al.* (2011); Yu and Aloimonos (2010); Xu *et al.* (2017); Ding *et al.* (2017)).

1.1 Zero Shot Learning

Zero-shot learning aims to recognize objects that are not seen during the training process of the model. Mainly, the zero-shot paradigm has two domains, seen and unseen. The data from seen and unseen classes are disjoint. ZSL leverages textual descriptions/attributes to transfer knowledge from seen to unseen classes. Ideally, the goal behind ZSL is to leverage the learning from the seen classes and generalize it for the unseen classes. Notice that the ZSL paradigm evaluates the models' generalizability using its performance on the unseen classes.

1.2 Generalized Zero Shot Learning

Although ZSL tries to mimic human intelligence in the learning models, its evaluation process limits the learning constraint for the Artificial models and often not able to reveal the true potentials/weaknesses of them. Therefore, the research community has proposed a more realistic evaluation setting named "Generalized Zero Shot Learning" (Scheirer *et al.* (2012)), where both seen and unseen classes are considered in the evaluation process of the learning model. Fig. 1.1 depicts the Zero-shot and Generalized Zero-shot setting.

1.3 Goals and Motivations

The goal of this dissertation is to propose generative models to perform zero-shot learning for image classification problems in computer vision. It seeks to highlight the role of zero-shot learning in machine learning/deep learning and summarize the literature. It also intends to outline a set of future directions for research in this domain.

Figure 1.1: Image credit (Xian *et al.* (2018a)), The left side showcases the training phase, and the right side is for the Test phase. Note that, at training time, for both ZSL and GZSL, the images and descriptions/attributes of the seen classes (D^s) are available. During the test time, the trained model is evaluated only on unseen classes (D^u) in the ZSL, whereas in the GZSL, it is evaluated on both seen and unseen classes ($D^s \cup D^u$). To facilitate classification without labels, both tasks use some form of auxiliary information, e.g., descriptions/attributes.

This dissertation has been inspired by some overarching challenges in the existing zero-shot models, specifically, generative zero-shot models. Despite the recent progress on such approaches, the generative models still have some key limitations. First of all, these models are trained only on the seen classes as the visual features for the unseen classes are not available. Knowing the fact that the seen and unseen classes share the same semantic feature space, it is expected from the generator to synthesize meaningful visual features for the unseen classes as well. However, these models show a large quality gap between the synthesized and the actual unseen visual features. The synthesized features for the unseen classes are prone to the seen class references. This behavior indicates the domain shift problem (Fu *et al.* (2015)). As

a result, the performance of generalized zero-shot (GZSL) learning suffers a lot since many of the synthesized unseen features are classified as seen classes.

The second major concern behind the existing generative models is the assumption that the semantic features are available in the desired form for a class category. e.g., clean attributes. However, in reality, it is hard to get. Getting the clean semantic features require a domain expert to annotate the attributes manually. Moreover, collecting a sufficient number of attributes for all the class categories is again labor-intensive and costly.

1.4 Contributions

The contributions of the dissertation are as follows.

1. Detail survey of the existing zero-shot methods for visual object classification.
2. A novel Student-Teacher generative model with an episodic meta learning-based training procedure.
3. A novel LsrGAN generative model to address the overfitting concern, it uses a novel Semantic-Regularized loss, leading to the state of the art performance on seven benchmark datasets in ZSL and GZSL.
4. Both Attribute-based and Wikipedia-based semantic information is considered for knowledge transfer.
5. The proposed SR-Loss is orthogonal to any GZSL model. Hence, it can be integrated with any GZSL model without adding any extra parameters.

1.5 Dissertation Outline

The dissertation is structured in the following manner.

Chapter 2 provides an overview of zero-shot learning. The first section introduces the various semantic information employed in the existing methods to address the ZSL and GZSL. The second section discusses the conventional ZSL methods termed as "Embedding methods". It is followed by an introduction to generative methods for ZSL. Primarily, this section reasons the use of generative models to tackle zero-shot object recognition and outlines the current state of art models in the field. The third section provides a mathematical overview of the zero-shot problem setting. Afterward, the fourth section describes the considered benchmark datasets in this dissertation, followed by a table containing a set of terminologies used throughout this dissertation to ease the reading.

Chapter 3 illustrates the Vanilla GAN model to address zero-shot learning. It highlights the main components of GANs such as Generator, Discriminator, and discusses a way to perform the zero-shot recognition. The subsequent section discusses some of the critical remarks behind the vanilla GAN model.

Chapter 4 describes a novel Student-Teacher model to simulate the zero-shot inference process in training itself. The first section introduces the novel student-teacher concept in generative models. The second section proposes a meta learning-based episodic training to simulate the zero-shot inference in the training process. It also discusses the multi students single teacher network and how an additional discriminator can help the Teacher to preserve the true distribution. The third section examines the performance of the Student-Teacher network using Wikipedia-based

datasets. Finally, the chapter concludes by mentioning the problems in the existing Student-Teacher model.

Chapter 5 progresses to demonstrate the gist of this dissertation, a novel LsrGAN model. It introduces the novel Semantic Regularized loss that helps the LsrGAN to learn from the unseen classes together with the seen classes. Thus, leading to address the overfitting concern towards seen classes. The chapter concludes with showcasing various experiments on seven benchmark datasets to prove the LsrGAN as a new state of the art generative model for ZSL and GZSL.

Chapter 6 concludes the dissertation by summarizing the contributions of the dissertation and highlights some of the future research directions.

Chapter 2

BACKGROUND AND RELATED WORK

This chapter will introduce the problem background, the available datasets, and existing approaches to address the zero-shot object recognition. Mainly the chapter is organized as follows. Section 2.1 introduces the various semantic information employed in the existing methods to address the ZSL and GZSL. Section 2.2 discusses the conventional ZSL methods termed as "Embedding methods". It is followed by an introduction to generative methods for ZSL in section 2.3. Primarily, this section reasons the use of generative models to tackle zero-shot object recognition and outlines the current state of art models in the field. Section 2.4 provides a mathematical overview of the zero-shot problem setting. Afterward, section 2.5 describes the considered benchmark datasets in this dissertation, followed by a table containing a set of terminologies used throughout this dissertation to ease the reading.

2.1 Semantic Information for ZSL and GZSL

In the zero-shot learning to perform the knowledge transfer, a piece of auxiliary information is required for an object. Most of the recent works use object attributes - nameable and shared visual properties of an object, as auxiliary information (mentioned in Fig. 1.1). However, getting this information for each class is a labor-intensive and costly practice. Therefore, there are approaches (Rohrbach *et al.* (2010); Akata *et al.* (2015b); Xian *et al.* (2016); Frome *et al.* (2013); Qiao *et al.* (2016); Lei Ba *et al.* (2015)) to explore other sources to retrieve a piece of auxiliary information. Mainly, these rely on the word embedding such as Word2Vec (Mikolov

et al. (2013)), glove (Pennington *et al.* (2014)), wordnet hierarchy (Miller (1995)) and recently BERT (Devlin *et al.* (2018)). However, getting these embeddings also require to have some standard semantic information available for an object class. Again that is a costly practice in the real world for fine-grained classes. To address this, (Elhoseiny *et al.* (2013)) proposes to use the Wikipedia articles for an object class. This information is easy to get but comes with a lot of noise and redundant information. Therefore, there are limited approaches that discuss the utilization of Wikipedia information to perform ZSL and GZSL. This dissertation discusses models that are capable of utilizing both Wikipedia and attribute-based auxiliary information.

2.2 Embedding Models

Early works on the zero-shot methods (Lampert *et al.* (2013); Norouzi *et al.* (2013a); Kankuekul *et al.* (2012); Jayaraman and Grauman (2014)) were using the human crafted attributes as a part of the semantic information. These models were all two-stage based to infer the label of an unseen class. In the first stage, the attributes of an input image are estimated, followed by searching the class, which attains the highest similarity with these estimated attributes. DAP (Lampert *et al.* (2013)) was one of the baseline models that proposes a two-state solution. It computes the posterior for each attribute (both seen and unseen) for a given input image feature using the probabilistic attribute classifiers in the first phase. Later, it estimates the class posteriors and predicts the class labels using MAP estimate. (Al-Halah *et al.* (2016)) similarly uses the probabilistic classifier for each attribute in the first stage and later uses the random forest to give labels. CONSE (Norouzi *et al.* (2013a)) uses Word2vec (Mikolov *et al.* (2013)); first, it predicts the seen class posteriors followed by projecting an image feature into the word2vec space as a part of its second stage. These

two-stage methods were facing domain shift issue (Fu *et al.* (2015)) and had limited performance on the zero-shot classes.

Afterward, the research community had started building models that map knowledge from one space to the other space. These models were referred to as "mapping models". For example, mapping from visual feature space to the semantic space. SOC (Palatucci *et al.* (2009)) maps the visual feature into the semantic space and then uses the nearest neighbor method on semantic features to get the nearest class for the label assignment. SJE (Akata *et al.* (2015b)) optimizes the structural SVM loss for mapping learning. ESZSL (Romera-Paredes and Torr (2015)) utilizes square loss to learn the mapping; it also regularizes the objective w.r.t Fornenius norm. On the other hand, ALE (Akata *et al.* (2015a)) learns the bilinear compatibility function between the image and the attribute space using ranking loss. Autoencoders (Baldi (2012)) were also used to learn this visual to the semantic mapping, specifically, SAE (Kodirov *et al.* (2017)) proposed a semantic autoencoder that tries to reconstruct the visual feature projected in the semantic space. These mappings later learned by nonlinear models such as Neural Networks. CMT (Socher *et al.* (2013)) uses a neural network with two hidden layers to learn a nonlinear projection from visual space to the semantic space - Word2vec (Mikolov *et al.* (2013)). Compared to previous models that directly use the trained deep models to extract the visual features (Lei Ba *et al.* (2015)), trains a deep convolution network to learn the visual features together with the mapping.

The above-mentioned models were following mapping from visual feature space to the semantic feature space. Later, (Zhang *et al.* (2017b)) claims that the visual feature space is more discriminate compared to the semantic feature space and could become beneficial for the zero-shot learning tasks. Therefore, it proposes the reverse mapping deep model that maps the semantic feature space to the visual feature space.

Following this logic, (Changpinyo *et al.* (2017)) also proposes a similar mapping model that projects the class semantic to the visual feature space. Later finds the nearest visual neighbor to assign the label. The mapping is learned using SVM with seen class examples.

Furthermore, there are other approaches (Zhang and Saligrama (2015, 2016)) that projects both semantic and visual features to the latent space and learns to embed them jointly there.

2.3 Generative Models

The embedding methods were preliminary models, but it played a crucial role to motivate the research community for pursuing the zero-shot learning, and since then, there is continuous research work progressing in this field. Especially, after the advances in the Generative models (Goodfellow *et al.* (2014); Arjovsky *et al.* (2017); Gulrajani *et al.* (2017)), the community has started utilizing these models to gain better zero-shot performance. The focus of this dissertation is to also work in the paradigm of generative models to address the zero-shot learning problem.

If we carefully look at the zero-shot problem setting. The ideal scenario is to have access to the visual features of unseen classes. Let us assume that there is a black box that gives us these features. In such a case, we could use the seen class visual features and unseen class visual features to apply the supervised learning to train a classifier for all the classes. Afterward, this trained classifier can be used to perform the zero-shot and generalized zero-shot classification. The generative model-based zero-shot approaches try to achieve the exact scenario. Mainly, using generative models, researchers try to replicate the human level reasoning to classify the unseen class objects. They claim that humans generally practice hallucination to visualize the unseen class object based on their prior knowledge (a form of auxiliary

knowledge for an object, e.g., semantic attributes). Later they used this imagination to perform the classification for unseen objects. Similarly, the trained generative models hallucinate (generates) the visual characteristics (visual features) of an object from its semantic features, which later used to perform zero-shot and generalized zero-shot classification.

Among the GAN based models, f-CLSWGAN (Xian *et al.* (2018b)) was one the earliest to highlight the use of GAN for zero-shot learning. Specifically, it used a conditional W-GAN (Arjovsky *et al.* (2017); Gulrajani *et al.* (2017)) with a simple classification loss. The classification loss enables the GAN to enforce the generated features to remain in its respective class categories. The model became the state of the art when it was published as it showed a significant improvement in zero-shot learning across the benchmark datasets compared to the previous embedding based methods. It used the attribute as semantic information. Inspired by the fact that training the GAN is hard, (V. Verma and Rai (2018)) proposed a variational autoencoder (VAE) based generative zero-shot model. Again this is a conditional generative model where VAE takes semantic information as an input and generates the visual feature for a class. Together with VAE, (V. Verma and Rai (2018)) uses a novel regressor network that aims to map the generated visual feature back to its original semantic form. Attributes are considered as semantic information here as well.

Later the advancement of the cycle GANs (Zhu *et al.* (2017)) inspired the community to utilize cyclic models to address the zero-shot learning problem. In cyclic generative models (Felix *et al.* (2018)), two conditional generators are employed, one that generates the visual features from the semantic feature information, and the other learns to obtain semantic information from the visual feature. This model outperformed the single GAN and VAE models across all the benchmark datasets, and the cyclic loss helps it to have a better generalization. Attributes are employed as

semantic information here. Regularizing generative models to improve the knowledge transfer capability of these models is also an active research area. (Li *et al.* (2019)) uses the same W-GAN model as f-CLSWGAN (Xian *et al.* (2018b)) but employs the novel regularizer using the soul samples to improve the ZSL performance. The authors suggest that soul samples capture the overall visual feature information of a class. Threfore forcing the generator to generate samples close to the soul sample results in more realistic visual feature generation. Here, the soul samples are nothing but the mean of the visual feature for a class. Seeing the benefits of various regularized losses, the community has started employing regularizers in the cyclic generative models as well. As a result, (Huang *et al.* (2019)) uses three mappings with the GANs, visual to semantic, semantic to visual, and metric learning. After realizing the hubness problems (Radovanović *et al.* (2010)) in the generative zero-shot models where the transformed data become hubs for the nearby class embedding leading to performance degradation in both zero-shot and generalized zero-shot learning. (Paul *et al.* (2019)) proposes a solution to address it by employing an intermediate network layer to fine-tune the actual visual features using the semantic features. It also discusses the bias problem in GANs for the seen classes and details a method to address using the validation set.

The single GAN and VAE based models that generate visual features from the semantic feature information were giving an adequate performance on the unseen classes, but they were not up to the mark, and mainly suffers in GZSL setting. Recently, researchers have started employing more complex models that have more than one generative models. CADA-VAE (Schonfeld *et al.* (2019)) uses two aligned VAEs to encode and decode the visual and semantic information in an aligned manner. The authors use a novel regularizer loss that tries to make the latent representation of the visual and semantic feature to remain close. They employ an MSE loss between

the mean and variance of two representations to force the latent representation to remain close enough. Later, DAZL (Atzmon and Chechik (2018)) proposes a way to incorporate two classifiers to get better performance on the generalized zero-shot learning setting, where one of the classifier address the seen classes and the other operates on the unseen classes. They also provided a novel gating classifier to select which classifier to choose during the inference time under GZSL.

All the above mentioned generative models use attributes as semantic information in their learning. There is not much work done using the Wikipedia articles in the zero-shot field as these sets of information are very noisy and contain redundant information. (Elhoseiny *et al.* (2013)) first time proposed the use case of noisy descriptions (Wikipedia articles) in zero-shot learning. It built a model that used Term Frequency - Inverse Document Frequency (TF- IDF) to deal with the semantic information for zero-shot learning. (Qiao *et al.* (2016)) showcases a way to denoise these noisy text descriptions by encouraging group sparsity on the connections to the textual terms. Later, (Elhoseiny *et al.* (2013)) proposed a framework that identifies the relevant text part from the text descriptions by understanding the visual feature characteristics of an object, and keeps only this information to train the zero-shot model. In the recent past, (Zhu *et al.* (2018)) presents the first GAN based model that utilizes the Wikipedia text information using the TF - IDF and gives the state of the art results. It employed an extra feed-forward layer together with the GAN as a denoiser to process the noisy text.

For the Wikipedia-based semantic information, there is a couple of recent work that employes state of the art cycle GAN and Meta- learning techniques to address ZSL and GZSL. Among these, (Chen *et al.* (2020)) proposes a Cycle GAN that learns the two-way generation from semantic to visual and from visual to semantic, almost similar to (Felix *et al.* (2018)). (Hu *et al.* (2018)) proposes a correction network,

a two-level meta-learning method, where the first module is considered as a learner that maps the text deceptions of an object to its corresponding visual feature mean. Later, the meta learner tries to add ϵ to do the correction.

2.4 Problem Setting

The zero-shot learning problem consists of seen (observed) and unseen (unobserved) categories of images and their corresponding text information. Images belonging to the seen categories are passed through a feature extractor (ResNet-101) to yield features $\{\boldsymbol{x}_i^s\}_{i=1}^{n_s}$, where $\boldsymbol{x} \in \mathcal{X}^s$. The corresponding labels for these features are $\{y_i^s\}_{i=1}^{n_s}$, where $y^s \in \mathcal{Y}^s = \{1, \ldots, C_s\}$ with C_s seen categories. The image features for the unseen categories are denoted as $\{\boldsymbol{x}_i^u\}_{i=1}^{n_u}$, where $\boldsymbol{x} \in \mathcal{X}^u$, the space of all image features is $\mathcal{X} := \mathcal{X}^s \cup \mathcal{X}^u$. As the name indicates, unseen categories are not observed, and the zero-shot learning model attempts to hallucinate these features with the rest of the information provided. Although we do not have the image features for the unseen categories, we are privy to the C_u unseen categories, where the corresponding labels for the unseen image features would be $\{y_i^u\}_{i=1}^{n_u}$, with $y^u \in \mathcal{Y}^u = \{C_s + 1, \ldots, C\}$, where $C = C_s + C_u$. From the text domain, we have the semantic features for all categories which are either binary attribute vectors or Term-Frequency-Inverse-Document-Frequency (TF-IDF) vectors. The category-wise semantic features are denoted as $\{\boldsymbol{t}_c^s\}_{c=1}^{C_s}$, for seen categories and $\{\boldsymbol{t}_c^u\}_{c=C_s+1}^{C}$ with $\boldsymbol{t} \in \mathcal{T} := \mathcal{T}^s \cup \mathcal{T}^u$. The goal of zero-shot learning is to build a classifier $\mathcal{F}_{zsl} : \mathcal{X}^u \to \mathcal{Y}^u$, mapping image features to unseen categories, and the goal of the more difficult problem of generalized zero-shot learning is to build a classifier $\mathcal{F}_{gzsl} : \mathcal{X} \to \mathcal{Y} := \mathcal{Y}^s \cup \mathcal{Y}^u$, mapping image features to seen and unseen categories.

Table 2.1: Dataset Information. For the attribute-based datasets, the (number) in seen classes denotes the number of classes used for test in GZSL.

	Attribute-based			Wikipedia descriptions			
	AWA	CUB	SUN	CUB (Easy)	CUB (Hard)	NAB (Easy)	NAB (Hard)
No. of Samples	30,475	11,788	14,340	11,788	11,788	48,562	48,562
No. of Features	85	312	102	7551	7551	13217	13217
No. of Seen classes	40(13)	150(50)	645(65)	150	160	323	323
No. of Unseen classes	10	50	72	50	40	81	81

2.5 Datasets

In this dissertation, I have considered a total of seven benchmark datasets. Mainly, these datasets have two subcategories based on the availability of the semantic information. They are categorized as (1) Attribute-based datasets, and (2) Wikipedia descriptions-based datasets.

Attribute-based datasets : For the attribute-based datasets, I have considered three datasets: *Animal with Attributes* (AWA) (Lampert *et al.* (2013)), *Caltech-UCSD-Birds 200-2011* (CUB) (Welinder *et al.* (2010)) and Scene UNderstanding (SUN) (Patterson and Hays (2012)). AWA is a medium scale coarse-grained animal dataset having 50 animal classes with 85 attributes annotated. CUB is a fine-grained, medium-scale dataset having 200 bird classes annotated with 312 attributes. SUN is a medium scale dataset having 717 types of scenes with 102 annotated attributes. Mainly I followed the split mentioned in (Xian *et al.* (2018a)) to have a fair comparison with existing approaches.

Wikipedia descriptions-based datasets: In order to address a more challenging ZSL problem with Wikipedia descriptions as auxiliary information, I have used two common fine-grained datasets with textual descriptions: CUB and *North America Birds* (NAB) (Van Horn *et al.* (2015)). The NAB dataset is larger compared to CUB having 1011 classes in total. I have used two splits, suggested by (Elho-

seiny *et al.* (2017)) in all the experiments to have a fair comparison with existing approaches. The splits are termed as *Super-Category-Shared* (SCS, Easy split) and *Super-Category-Exclusive* (SCE, Hard split). These splits represent the similarity between seen and unseen classes. The SCS-split has at least one seen class for every unseen class belonging to the same parent. For example, "Harris's Hawk" in the unseen set and "Cooper's Hawk" in the seen set belong to the same parent category, "Hawks." On the other hand, in the SCE-split, the parent categories are disjoint for the seen and unseen classes. Therefore, SCS and SCE splits are considered as Easy and Hard splits. The details for each dataset and class splits are given in Table 2.1.

Table 2.2: Terminology Used in This Dissertation.

Notation	Description
D^s	Seen class set
D^u	Unseen class set
$\widetilde{D^s}$	Train seen class set for the student network
$\widetilde{D^h}$	Train seen class set for the teacher network (held out set)
$\mathcal{X}$	Visual Feature space
$\mathcal{X}_c^s$	Visual Feature space for the seen class c
$\mathcal{X}_c^u$	Visual Feature space for the unseen class c
$\widetilde{x_c}$	Generated visual feature for the class c
$\mathcal{T}$	Semantic space
$\mathcal{T}_c^s$	Semantic vector (TF-IDF) for the seen class c
$\mathcal{T}_c^u$	Semantic vector (TF-IDF) for the unseen class c
$\mathcal{Y}$	Label space
y_c^s	Label for the seen class c
y_c^u	Label for the unseen class c
$\mathcal{Z}$	Normal vector from $\mathcal{N}(0,1)$, size :100
G_{θ_g}	Generator network with θ_g parameter
D_{θ_d}	Discriminator network with θ_d parameter
μ_c	Visual feature mean of x_c
$\widetilde{\mu_c}$	Generated visual feature mean of $\widetilde{x_c}$
ϵ_c	Difference between μ_c and $\widetilde{\mu_c}$
$\widetilde{\epsilon_c}$	Generated difference between μ_c and $\widetilde{\mu_c}$ by Teacher Network
$Teacher_\theta$	Teacher Network with parameter θ
$\mathcal{T}_{sim}(c_i, c_j)$	Semantic similarity between class c_i and c_j
$\mathcal{X}_{sim}(\boldsymbol{\mu}_{c_i}, \boldsymbol{\mu}_{c_j})$	Visual similarity between the mean visual features of class c_i and c_j
ϵ_d	The margin difference between $\mathcal{T}_{sim}$ and $\mathcal{X}_{sim}$

Chapter 3

VANILLA GAN BASED APPROACH FOR ZERO SHOT LEARNING

The use of GANs has become popular among the researchers because they allow the learning models to have the human-level capability of performing hallucinations while doing recognition tasks. Generally, humans practice the hallucination to visualize the class objects when something is not known to them visually. Similarly, the researchers try to model the same hallucination practice in the learning models by employing conditional generative models.

The conditional generator hallucinates (generates) the visual characteristics (visual features) of an object from its semantic features. Once the generator trained, it can be utilized to generate as many visual features as one wants for the unseen classes by arbitrarily sampling the noise vector. Later, using these generated visual features, the classification task simplifies to the conventional supervised learning problem. Although the GAN based models seem promising, the generator network is not powerful enough to give us the accurate visual features for all the unseen classes, and that lead to the various GAN based architectures development to tackle zero-shot learning in the research community. This chapter illustrates the Vanilla GAN model to address zero-shot learning. Section 3.1 highlights the main components of GANs such as Generator and Discriminator. It also discusses a way to perform the zero-shot recognition using these models. The subsequent section 3.2 analyze some of the critical remarks behind the vanilla GAN model.

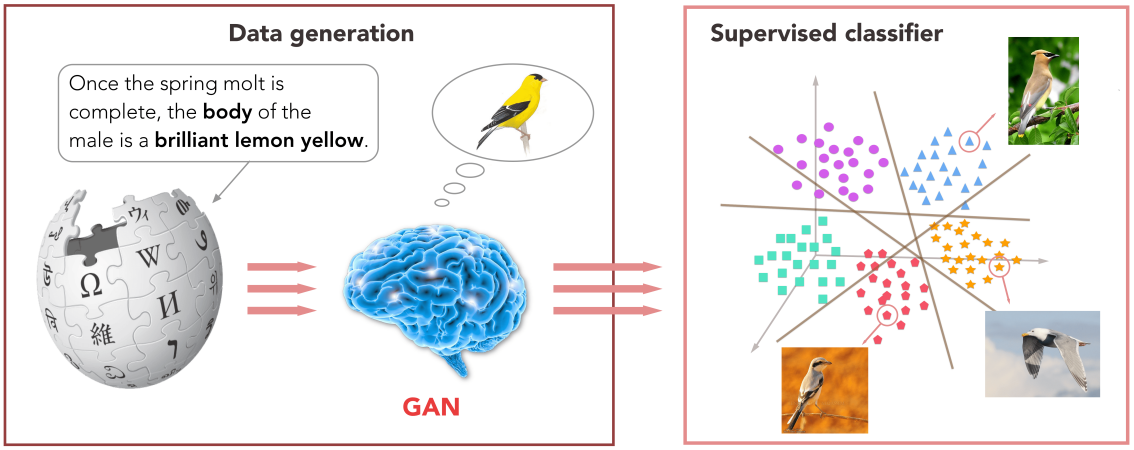


Figure 3.1: Image credit (Zhu *et al.* (2018)), Illustration of the generative zero-shot learning, the GAN uses semantic information to hallucinate (generate) the visual feature information for the unseen classes. Later together with the seen visual features and generated visual features, a supervised classifier can be trained to predict the label for zero-shot and generalized zero-shot settings.

3.1 Introduction of Vanilla GAN for Zero Shot Learning

Let suppose the conditional generator as $G_{\theta_g} : \mathcal{Z} \times \mathcal{T} \rightarrow \mathcal{X}$, and the discriminator as $D_{\theta_d} : \mathbb{R}^X \rightarrow \{0, 1\} \times \mathbb{L}_{cls}$, where $\mathbb{L}_{cls}$ is the set of class labels. θ_g and θ_d are the parameters of the generator and the discriminator, respectively. Fig. 3.1 represents the idea behind utilising GANs to address the zero-shot learning.

3.1.1 Generator (G)

As mentioned earlier, the conditional generator ($G_{\theta g}$) is used as a feature generator. It takes the semantic feature ($\mathcal{T}_i$) of a class and a random vector $\mathcal{Z}$ sampled from the Gaussian distribution $\mathcal{N}(0, 1)$ to generate the visual feature ($\mathcal{X}_i$). The generator network is a fully connected network since we are dealing with the visual features not the images. After the advancement of WGANs (Arjovsky *et al.* (2017); Gulrajani *et al.* (2017)) the training becomes more stable in GANs. Thus, the vanilla GAN that I am introducing here is based on WGAN. The loss of the generator is,

$$\mathcal{L}_G = -\mathbb{E}_{\sim p_z}[D_{\theta_d}(G_{\theta_g}(\mathcal{T}, \mathcal{Z}))] + L_{cls}(G_{\theta_g}(\mathcal{T}, \mathcal{Z})), \tag{3.1}$$

Where the first term is a Wasserstein loss (Arjovsky *et al.* (2017); Gulrajani *et al.* (2017)) and the second term is a classification loss (cross entropy). Once the Generator network is trained well. We can use it to generate the visual feature via $\tilde{x}_c \leftarrow G_{\theta_g}(\mathcal{T}_c, \mathcal{Z})$.

3.1.2 Discriminator (D)

The discriminator has two branches, where one is used to distinguish between the real and fake visual feature, and the second branch performs the classification - to categorize the generated features belong to their respective classes. It is trained with the following loss,

$$\begin{aligned}\mathcal{L}_D =& \mathbb{E}_{z\sim p_z}[D_{\theta_d}(G_{\theta_g}(T, z))] - \mathbb{E}_{x\sim p_{data}}[D_{\theta_d}(x)] + \lambda L_{GP} \\ &+ \frac{1}{2}(L_{cls}(G_{\theta_g}(\mathcal{T}, \mathcal{Z})) + L_{cls}(x)),\end{aligned} \tag{3.2}$$

The first two terms are for the Wasserstein distance (Arjovsky *et al.* (2017); Gulrajani *et al.* (2017)) between the real and fake distributions. The third term enforces the Lipschitz constraint - Gradient penalty term, and the last two terms are used for classification (cross entropy).

3.1.3 Zero-Shot Recognition

After finishing the training, we can use the trained generator to generate features for the unseen classes conditioning on the semantic features as $\tilde{x_u} = G_{\theta_g}(t_u, \mathcal{Z})$. We can generate an arbitrary number of visual features by sampling different $\mathcal{Z}$ for the same semantic feature t_u. Together with the real visual features of the seen classes and generated features of the unseen classes, zero-shot recognition becomes a conventional

supervised classification problem as mentioned in Fig. 3.1. Any supervised learning algorithm can work then after. Generally, researchers use SVM classifier or nearest neighbourhood to perform the classification. In the further chapters will introduce more details about this.

3.2 Remarks on Vanilla GAN Model for Zero Shot Learning

The vanilla GAN model was one of the simplest models to address the ZSL. The purpose of introducing this model here is to make the reader aware of the basic idea behind employing GANs to address ZSL. Notice that the vanilla GAN model does not have any regularizer that can help it to generate visual features that mirror characteristics of the actual visual features except the feedback from the discriminator, this hinders the generation performance significantly. Also, since we do not have the availability of the unseen visual features, the vanilla GAN model is trained using the seen classes only. This leads to an issue of overfitting towards seen classes if the training is extensively performed on seen classes, will give more details about it in the next chapters. Primarily, the seen class overfitting is the driving motivation behind my research work and the two novel models that I have proposed to tackle it.

Chapter 4

PROPOSED STUDENT-TEACHER MODEL FOR ZERO SHOT LEARNING

The generative models have shown their value by uplifting the ZSL and GZSL performance with a significant margin compared to the conventional embedding methods. However, as mentioned in chapter 3, these models have some key limitations. The current GAN based model follows the conventional learning process meaning, the GAN is trained on the seen classes only, and it is not aware of the fact that there is a possibility that it may encounter an entirely new class during inference time. For instance, on the seen classes, this GAN based model gives a classification accuracy of around 89% for the CUB, SCS split; however, when testing it on the unseen examples - zero-shot testing, the accuracy drops to 40%. One way to think about this flaw is because the GAN is not trained in a way to counter something new during the inference part (zero-shot testing). In short, the learning process requires some imitation of the zero-shot pattern. With this in a view, I am introducing a Student-Teacher based model. Here, the students are considered as conditional WGANs, similar to the one mentioned in chapter 3, who generate the visual features from the semantic information. On the other hand, the Teacher network tries to improve the student's generation for the unseen class.

The chapter is organized as follows. Section (4.1) provides an introduction to the proposed generative model. Section (4.2) outlines the model training approach by providing the Meta Learning based Episodic Training for Student-Teacher Model. Finally, section (4.3) provides the results on various experiments with the proposed

model on a Wikipedia-based datasets and showcase some of the limitations of the Student-Teacher model.

4.1 Student - Teacher GAN : A Novel Generative Model Using the Meta Learning Concepts

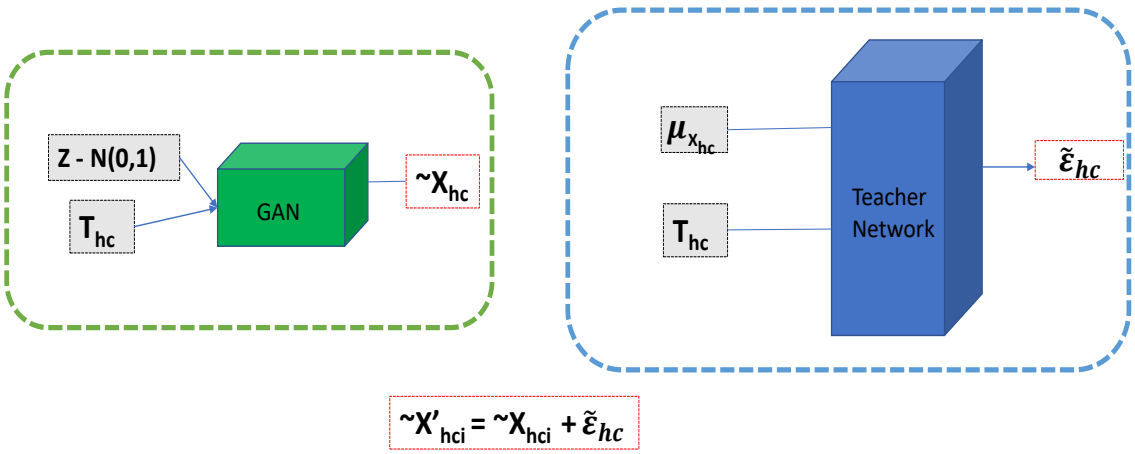


Figure 4.1: Model overview of the Student-Teacher network. Note that the GAN mentioned in the Green box is the Student network for zero-shot learning; essentially, it follows the same architecture mentioned in chapter 3. The blue box depicts the Teacher network; it is a fully connected network that takes the mean of the generated features, Text description of a class, and outputs the $\widetilde{\epsilon}$ - the improvement for the student. Later, $\widetilde{\epsilon}$ is added in the generated visual feature from the student network as mentioned in the figure to have a rich visual feature synthesis.

4.1.1 Student Network

As mentioned in Fig. 4.1, the student network is the vanilla WGAN model. It has to learn the mapping from the semantic features ($\mathcal{T}$) to visual features ($\mathcal{X}$) using the seen class data examples.

4.1.2 Teacher Network

The Teacher Network is a simple feed-forward network that takes the mean of the visual features generated by the student GAN, the text description of a class, and outputs the improvement ($\widetilde{\epsilon}$). The loss of the Teacher network is defined as,

$$\mathcal{L}_t = \frac{1}{C}\sum_{c=1}^{C} ||\epsilon_c - \widetilde{\epsilon}_c||^2 \tag{4.1}$$

Here ϵ_c is the difference between the generated visual feature mean and the real visual feature mean. Basically, it is $\mu_c - \widetilde{\mu}_c$ for a class c.

4.2 Meta Learning Based Episodic Training for Student-Teacher Model

This section will discuss the training strategy that can help to imitate the zero-shot setting in the training process itself for the generative model. Notice that we have access to the visual ($\mathcal{X}_s$) and semantic ($\mathcal{T}$) features of the seen dataset (D^s). The unseen dataset (D^u) only allows us to have the text information ($\mathcal{T}_u$). Therefore, to imitate the zero-shot setting in the training part itself, we need to put a set of classes aside from the seen class dataset (D^s) and treat them as a zero-shot class. The subset of classes from the seen class set that is not used in the training phase of the GAN is considered as a held-out set ($\widetilde{D^h}$), and the remaining classes used for the training is termed as a new train set ($\widetilde{D^s}$). This is the basic plan behind the episodic based meta-learning strategy for the Student-Teacher model.

The training process for Student and Teacher network is independent of each other, meaning; it is a two-stage process. First, the GAN (Student network) is trained using the train set ($\widetilde{D^s}$). Afterward, I will fix the GAN parameters and get the inference using the held-out set ($\widetilde{D^h}$) from the GAN. The output of the GANs for the held-out set ($\widetilde{D^h}$) will become the training input to the Teacher network. Since the true epsilon

(ϵ) for the held-out set ($\widetilde{D^h}$) is already known, the Teacher network will be trained using the loss mentioned in the equation 4.1 in the second stage of the training.

In the two-stage process, the held-out set ($\widetilde{D^h}$) seems zero-shot classes; the model tries to adopt the zero-shot behavior in the training process itself with these held out set. Notice that, the held-out set ($\widetilde{D^h}$) becomes a zero-shot set for the student network, where the Teacher network tries to correct the student network's guess. Ideally, the Teacher should have more knowledge and experience on a particular subject compared to his/her students. Following the same intuition, it is not a good idea to train the Teacher network only using the output of the students on a held-out set ($\widetilde{D^h}$), which has only a few sets of seen classes. Additionally, we have one more challenge here; the Teacher network relies on the student network's learning. Therefore, to mitigate the aforementioned challenges, the next section will discuss the multi Students single Teacher model described in Fig. 4.2.

4.2.1 Multi Students Single Teacher Network

In Fig. 4.2, we can see that there are five GAN models (student networks) and one Teacher network. Here, I am specifically focusing on one of the benchmark datasets - CUB 200 (Welinder *et al.* (2010)) with SCS split to explain the Multi Students Single Teacher Network. CUB - 200, has a total of 200 classes from which 150 are considered as seen classes, and the rest 50 are unseen classes. The split between seen and unseen classes for the CUB 200 is standardized, and the zero-shot research community follows the same. Ideally, we should utilize all the 150 seen classes to train the Teacher network, making sure that the Teacher gets all the available knowledge. To achieve the same, I trained five GANs (students). In order to have the zero-shot pattern in the training set, I need to make sure that each student learns partial classes from the overall seen class set. Therefore, I split the seen set (D^s) into 120 classes for

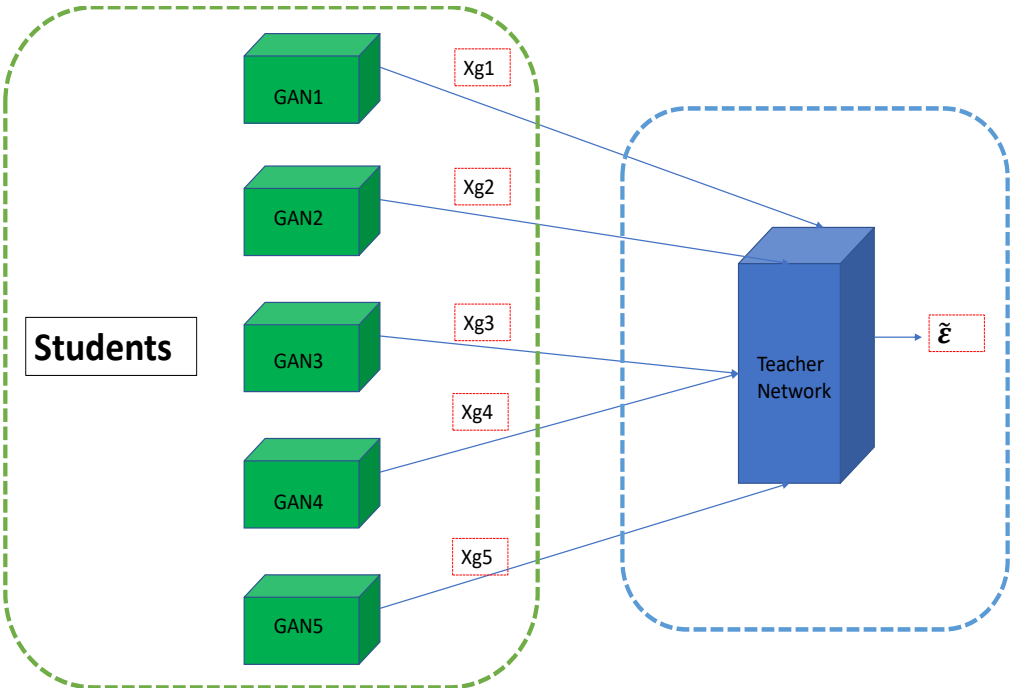


Figure 4.2: The green box contains students (GANs); each of them is trained with a seperate train set ($\widetilde{D^s}$). The blue box represents the Teacher network. It takes the output of the students and generates the improvement - $\widetilde{\epsilon}$. Note that the input and output of the Teacher network are the same as mentioned in Fig 4.1.

train set ($\widetilde{D^s}$) and rest 30 as held-out set ($\widetilde{D^h}$). Moreover, each GAN has a mutually exclusive held-out set ($\widetilde{D^h}$) from each other. So when I combine these held-out sets for all the five GANs, I should get the entire seen dataset (D^s), $30 * 5 = 150$. Each student (GAN) learns from its respective train set ($\widetilde{D^s}$) as a part of the first phase training. The training process for the Teacher network now becomes the ideal one with the above mentioned episodic based data splitting. The teacher will get a chance to learn from all the seen classes. During the second phase of the training, the Teacher network will invoke each student to guess from its held-out set ($\widetilde{D^h}$) regarding the visual characteristic of a class. Later, the Teacher will try to improve the student's guess using equation 4.1, as mentioned above. This process has been carried out for every student. Learning from each student helps the Teacher network to generalize well for all the seen classes.

4.2.2 Zero Shot Recognition Using Student - Teacher Network

Now for the unseen (D^u) feature generation, an equal number of samples are extracted from each student GAN, the teacher network will later correct them, and will have the unseen visual feature set. Notice that we already have access to the seen visual features (x_i^s) so we can use the generated unseen visual features ($\widetilde{x_i^u}$) and seen visual features (x_i^s) to train a classifier in a supervised manner to perform the ZSL and GZSL.

4.2.3 Teacher Network with Discriminator

The previous section discussed the multi Student single Teacher network and the training procedure. Note that the Teacher network has only one loss $\mathcal{L}_t$. With this, the Teacher network can help to improve the visual feature generation part but, adding $\widetilde{\epsilon}$ to the generated visual feature from GAN can arbitrarily change the visual feature distribution for a particular class. It could lead to the worst results for the zero-shot and generalized zero-shot learning as the data distribution is not preserved in the generated feature. An additional discriminator is employed with the teacher network to preserve the feature distribution.

4.2.4 Discriminator with Teacher Network

The Teacher Discriminator (T_D) is similar to the Discriminator mentioned in chapter 3. It has mainly two branches, one to distinguish between the real and fake visual features, and the second branch performs the classification to categorize the generated features to their respective classes. It is trained with the following loss,

$$\begin{aligned}\mathcal{L}_{T_D} =& \mathbb{E}_{z\sim p_z}[T_D(Teacher_\theta(\mathcal{T}, \mu(\tilde{x}_c))] - \mathbb{E}_{x\sim p_{data}}[T_D(x)] + \lambda L_{GP} \\ &+ \frac{1}{2}(L_{cls}(Teacher_\theta(T, \mu(\tilde{x}_c))) + L_{cls}(x)),\end{aligned} \tag{4.2}$$

Similarly, the loss for the Teacher network is also changed now,

$$\mathcal{L}_t = -\mathbb{E}_{z\sim p_z}[T_D(Teacher_\theta(\mathcal{T}, \mu(\tilde{x}_c)))] + L_{cls}(Teacher_\theta(\mathcal{T}, \mu(\tilde{x}_c))) + \frac{1}{C}\sum_{c=1}^{C} ||\epsilon_c - \widetilde{\epsilon}_c||^2, \tag{4.3}$$

In equation 4.2, the first two terms are for the Wasserstein distance between the real and fake distributions. The third term enforces the Lipschitz constraint - Gradient penalty term, and the last two terms are used for classification (cross-entropy). Note that the equation 4.3 is an updated version of 4.1 for the Teacher network.

4.3 Experiment Results

This section will discuss the various experiments that are conducted to showcase the performance of the Student-Teacher generative model.

4.3.1 Datasets

In order to evaluate the Student-Teacher model, mainly Wikipedia-based datasets are explored. The details of the datasets are already mentioned in chapter 2.

4.3.2 Implementation Details and Performance

The part-based features (e.g., belly, leg, wing, etc.) from VPDE-net (Zhang *et al.* (2016)) are used as visual features, suggested by (Zhu *et al.* (2018); Xian *et al.* (2018b)). We have utilized the TF-IDF to extract the features from the Wikipedia

descriptions. For a fair comparison, all of the experiment settings are kept the same as reported in (Zhu *et al.* (2018)).

The base block of the Student-Teacher model is WGAN, which is implemented using a multi-layer perceptron. Specifically, the student generator G_{θ_g} has one hidden unit having 4096 neurons and LeakyReLU as an activation function. It has a Tanh as an output activation since the VPDE-net feature varies from -1 to 1. $\mathcal{Z}$ is sampled from the normal Gaussian distribution. To perform the denoising and dimensionality reduction for Wikipedia descriptions, we have employed a fully connected layer with a feature generator. In the proposed model, the student discriminator D_{θ_d} has two branches. One is used to play the real/fake game, and the other performs the classification on the generated/real visual feature. The discriminator also has 4096 units in the hidden layer with ReLU as an activation. The teacher network is a simple multi-layer perceptron having 4096 hidden neurons with LeakyReLU as an activation. It also uses Tanh, an output layer, to compensate for the -1 to 1 visual feature range. The associated discriminator with the Teacher network follows the same student discriminator architecture. To perform the zero-shott recognition nearest neighbor prediction is employed. Top-1 accuracy is used to assess the ZSL setting.

Table 4.1 presents the summaries of the results for the Student-Teacher Network (Multi Students Single Teacher). CUB (Welinder *et al.* (2010)) and NAB-404 (Van Horn *et al.* (2015)) are used with the standard splits, SCS and SCE for the performance evaluation. Various state of the art methods is considered to showcase the comparison performances. It is evident that Student-Teacher Network is not giving us the state of the art results, and GAZSL (Zhu *et al.* (2018)) still holds the highest performance. Despite that, the performance of the Student-Teacher network is way better than the other models. Also, notice that the Student-Teacher Network for the

Table 4.1: ZSL results on CUB and NAB datasets with Wikipedia descriptions as semantic information on the two-split setting. We have used Top-1 % accuracy for ZSL.

Methods	Zero Shot Learning			
	CUB		NAB	
	Easy	Hard	Easy	Hard
WAC-Linear (Elhoseiny *et al.* (2013))	27.0	5.0	-	-
WAC-Kernal (Elhoseiny *et al.* (2016))	33.5	7.7	11.4	6.0
ESZSL (Romera-Paredes and Torr (2015))	28.5	7.4	24.3	6.3
ZSLNS (Qiao *et al.* (2016))	29.1	7.3	24.5	6.8
Sync-fast (Changpinyo *et al.* (2016))	28.0	8.6	18.4	3.8
ZSLPP (Elhoseiny *et al.* (2017))	37.2	9.7	30.3	8.1
GAZSL (Zhu *et al.* (2018))	**43.7**	**10.3**	**35.6**	8.6
Student - Teacher	42.5	10.1	34.6	**8.7**

SCE split of the NAB-404 gives superior results than GAZSL, although the improvement is too low. It is still an indication that there is a scope for an improvement. In the following section, will discuss some of the potential issues with this Student-Teacher model and some future direction to deal with them.

4.4 Limitations of Student-Teacher Model

Clearly, from Table 4.1, it is evident that the Student-Teacher model is not giving us the state of the art results compared to other models such as (Zhu *et al.* (2018)). Here, in this section, I will mention the issues and reasons for the performance. As discussed earlier, the training process for the Student-Teacher model is a two-stage process. The seen set is divided into two parts - train set ($\widetilde{D^s}$) and held-out set ($\widetilde{D^h}$). When I evaluate the performance on the held-out set ($\widetilde{D^h}$) using the trained students (GANs) only, the accuracy seems quite low. It was around 0.9% to 1%. This clearly

shows that the GANs are not generalizing well, and students only know the classes they have been trained on. The second stage of the training process uses GANs with the held-out dataset to train the Teacher network. Clearly, at every epoch, the performance on the held-out set increases, meaning the Teacher is helping the GANs (Students) to improve their performance. However, the performance of the zero-shot classes starts decreasing after certain epochs, meaning the Student-Teacher model overfits the held-out set and have less generalizability towards unseen classes. Initially, without the Teacher network, the GANs (students) were giving 40% to 41 % accuracy on the unseen classes that reduces to 36% to 37% with the Teacher network. This showcases the overfitting concern on the held-out set.

As mentioned above, the Teacher is not helping the GANs (Students) in the desired manner; actually, it is worsening the zero-shot performance. Further understanding this from the real-world perspective, I realized that I was training the Teacher network using the held-out set ($\widetilde{D^h}$) only. The GANs (Students) have not seen those classes before, so they would have made mistakes, and the teacher will rectify them by supplying large $\widetilde{\epsilon}$. Unfortunately, following this strategy, the Teacher network always tries to give large $\widetilde{\epsilon}$ as an output; since it assumes that all GANs (Students) are foolish. However, in the real world, it may not be the case always. So to mitigate this issue, I incorporated few classes from the train set ($\widetilde{D^s}$) of GANs (for which the students were performing well) and forced the Teacher network to improve these classes as well. It helped the Teacher network to control the output value of $\widetilde{\epsilon}$, and finally, the teacher stopped worsening the student performance. Eventually, I see improvement in performance. However, the improvement was not to a larger extent, the combined performance of the Teacher-Student network is almost similar to GAZSL (Zhu *et al.* (2018)). It seems the added complexity is not worth. Hence stopped pursuing this idea.

Chapter 5

PROPOSED LSRGAN MODEL FOR ZERO SHOT LEARNING

Driven by the recent advances in generative modeling (Goodfellow *et al.* (2014); Arjovsky *et al.* (2017); Gulrajani *et al.* (2017)), there is a growing interest in the research community to develop generative models (Xian *et al.* (2018b); V. Verma and Rai (2018); Felix *et al.* (2018); Li *et al.* (2019); Zhu *et al.* (2018)) to tackle the ZSL problem. Broadly, these models are conditional generative models that use the semantic information (descriptions/attributes) to synthesize artificial examples. Later, a classifier is trained using these synthesized examples to perform the zero-shot classification. Since the conditional image generation is an arduous task as the images are too subtle, these generative methods rely on the visual features extracted from the deep models as mentioned in earlier chapters. For the reminder purpose, here again describing the key limitations of the generative zero shot models. First of all, these models are trained only on the seen classes as the visual features for the unseen classes are not available. Knowing the fact that the seen and unseen classes share the same semantic feature space, it is expected from these generative models to synthesize meaningful visual features for the unseen classes as well. However, these models show a large quality gap between the synthesized and the actual unseen features. The synthesized features for the unseen classes are prone to the seen class references. This behavior indicates the domain shift problem (Fu *et al.* (2015)). As a result, the performance of generalized zero-shot (GZSL) learning suffers a lot since many of the synthesized unseen features are classified as seen classes. For instance, the baseline model F-GAN (Xian *et al.* (2018b)) achieves accuracy of 57.3% on *Caltech-UCSD-Birds 200-2011* (CUB) dataset (Welinder *et al.* (2010)) for ZSL. However, when it

comes to GZSL, the accuracy drops to 43.7% (13.6% drop) for the unseen classes. Notice that GZSL is a more realistic setting where the test set contains both seen and unseen class instances.

The second major concern behind the existing generative models is the assumption that the semantic features are available in the desired form for a class category. e.g., clean attributes. However, in reality, it is hard to get. Getting the clean semantic features require a domain expert to annotate the attributes manually. Moreover, collecting a sufficient number of attributes for all the class categories is again labor-intensive and costly. To address these concerns, this chapter introduces a novel LsrGAN - a generative model that Leverages the Semantic Relationship between seen and unseen categories and explicitly performs knowledge transfer by incorporating a novel Semantic Regularized Loss (SR-Loss). The proposed model learns to transfer semantic knowledge from both noisy text descriptions (like Wikipedia articles) as well as semantic attributes for zero-shot learning and generalized zero-shot learning.

The chapter is organized as follows. Section (5.1) provides an introduction to the proposed generative model with an intuitive example. Section (5.2) outlines the model approach by providing the various components of the model in detail. This section also introduces a novel SR-loss proposed in this work to uplift the LsrGAN performance. The training algorithm of the proposed generative model is mentioned in section (5.3). Section (5.4) provides the results of various experiments with the proposed model on a total of seven benchmark datasets, including the Wikipedia text-based CUB and NABirds splits, and Attribute-based AWA, CUB, and SUN to showcase the superiority of the proposed LsrGAN model.

5.1 LsrGAN : A Novel Generative Model for ZSL and GZSL

LsrGAN leverages semantic relationships between seen and unseen categories and transfers the same to the generated image features. The knowledge transfer has been implemented through a unique semantic regularization framework called the Semantic Regularized Loss (SR-Loss). In Fig. 5.1, "Dolphin", an unseen class, has a high semantic similarity with classes such as "Killer whale" and "Humpback whale" from the seen class set. These two seen classes could become the potential neighbors of Dolphin in the visual space. Therefore, if we do not have the luxury to get the real visual feature for Dolphin class, we could utilize these neighbors to form indirect visual references to make the learning possible for the Dolphin class. SR-Loss primarily helps to achieve the same. In this way, supporting the generative model to learn from the unseen classes helps it better understand the difference between seen and unseen classes. The LsrGAN also trains a classifier that guides in the feature generation, and since the classifier is integrated it is not required to train a separate classifier to perform ZSL and GZSL recognition. Extensive experiments on seven widely used standard benchmark datasets demonstrate that LsrGAN outperforms the previous state-of-the-art approaches.

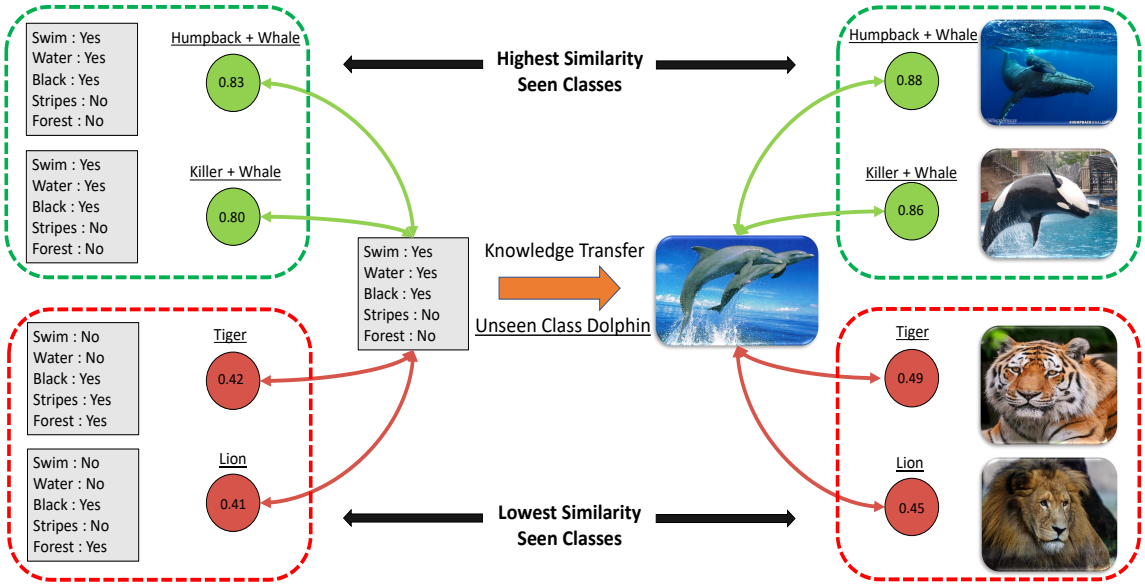


Figure 5.1: Driving motivation behind leveraging the semantic relationship between seen and unseen classes to infer the visual characteristics of unseen classes. Notice that though the feature representations are different, the class similarity values are almost the same. e.g. "Dolphin" has almost identical similarity values in visual and semantic space with other seen classes. The similarity values are mentioned in the circles, and computed using the cosine distance.

5.2 Proposed Approach

In this section, will introduce the proposed approach for the LsrGAN model, and will discuss the novel SR-Loss and how it helps the LsrGAN to learn from the unseen classes leading to address the overfitting concern towards seen classes.

5.2.1 Adversarial Image Feature Generation

Using the image features of the seen categories and the semantic features of the seen and unseen categories, I propose a generative adversarial network to hallucinate the unseen image features for each of the unseen categories. A conditional Wasserstein Generative Adversarial Network (WGAN) is employed to generate image features for the unseen categories using semantic features as input (Arjovsky *et al.* (2017)). The

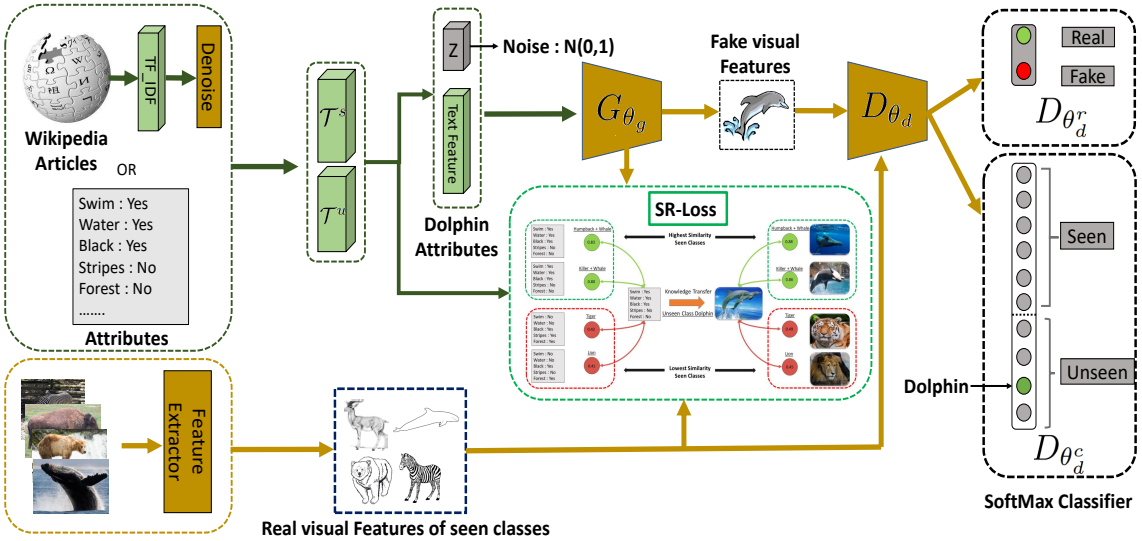


Figure 5.2: Conceptual illustration of the proposed LsrGAN model. The basis of LsrGAN is a conditional WGAN. The novel SR-Loss is introduced to help the G_{θ_g} to understand the semantic relationship between classes and guide it for applying the same during visual feature generation. The G_{θ_g} will use $\mathcal{T}^s$ and $\mathcal{T}^u$ to generate visual features. The D_{θ_d} has two branches used to perform real/fake game and classification. Notice that when I train the G_{θ_g} using $\mathcal{T}^u$, only the classification branch remains active in D_{θ_d} as the unseen visual features are not available.

WGAN aligns the real and generated image feature distributions. In addition, the LsrGAN has a feature classifier that is trained to classify image features into C categories of seen and unseen classes. The components of the WGAN are described in the following.

Feature Generator: The conditional generator in the WGAN has parameters θ_g and is represented as $G_{\theta_g} : \mathcal{Z} \times \mathcal{T} \rightarrow \mathcal{X}$, where $\mathcal{Z}$ is the space of random normal vectors $(0, I)$ of $|\mathcal{Z}|$ dimensions. Since the TF-IDF features from the Wikipedia articles may contain repetitive and non-discriminating feature information, I have applied a denoising transformation upon the TF-IDF vector using a fully-connected neural network layer as proposed by (Zhu *et al.* (2018)). The WGAN takes as input a random noise vector $\boldsymbol{z} \in \mathcal{Z}$ concatenated with the semantic feature vector $\boldsymbol{t}_c$ for a

category c, and generates an image feature $\tilde{\boldsymbol{x}}_c \leftarrow G_{\theta_g}(\boldsymbol{z}, \boldsymbol{t}_c)$. The generator is trained to generate image features for both seen categories ($\tilde{\boldsymbol{x}}_c^s \leftarrow G_{\theta_g}(\boldsymbol{z}, \boldsymbol{t}_c^s)$) and unseen categories ($\tilde{\boldsymbol{x}}_c^u \leftarrow G_{\theta_g}(\boldsymbol{z}, \boldsymbol{t}_c^u)$). In order to generate image features that are structurally similar to the real image features, I have added visual pivot regularization, $\mathcal{L}_{vp}$ that aligns the cluster centers of the real image features with the cluster centers of the generated image features for each of the C_s categories (Zhu *et al.* (2018)). This is implemented only for the seen categories where we have real image features.

$$\mathcal{L}_{vp} = \min_{\theta_g} \frac{1}{C_s} \sum_{c=1}^{C_s} \big|\big| \mathbb{E}_{(\boldsymbol{x}, y=c)\sim(\mathcal{X}^s, \mathcal{Y}^s)}[\boldsymbol{x}] - \mathbb{E}_{(\boldsymbol{z}, \boldsymbol{t}_c^s)\sim(\mathcal{Z}, \mathcal{T}^s)}[G_{\theta_g}(\boldsymbol{z}, \boldsymbol{t}_c^s)] \big|\big|. \tag{5.1}$$

Feature Discriminator: I train the WGAN with an adversarial discriminator having two branches to perform the real/fake game and classification. The discriminator has parameters θ_d^r and θ_d^c for two branches respectively and is denoted as D_{θ_d}. The real/fake branch of the discriminator learns a mapping $D_{\theta_d^r} : \mathcal{X} \rightarrow \mathbb{R}$ using the generated and real image features to output $D_{\theta_d^r}(\tilde{\boldsymbol{x}}^s)$ and $D_{\theta_d^r}(\boldsymbol{x}^s)$ that are used to estimate the objective term $\mathcal{L}_d$. The objective $\mathcal{L}_d$ is maximized w.r.t. the discriminator parameters θ_d^r and minimized w.r.t. the generator parameters θ_g.

$$\mathcal{L}_d = \min_{\theta_g} \max_{\theta_d} \mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}^s}\big[D_{\theta_d}(\boldsymbol{x})\big] - \mathbb{E}_{(\boldsymbol{z}, \boldsymbol{t})\sim(\mathcal{Z}, \mathcal{T}^s)}\big[D_{\theta_d}(G_{\theta_g}(\boldsymbol{z}, \boldsymbol{t}))\big] + \lambda_{gp}\mathcal{L}_{gp} \tag{5.2}$$

where, the first two terms control the alignment of real image feature and the generated image feature distributions. The third term is the gradient penalty to enforce the Lipschitz constraint with $\mathcal{L}_{gp} = (||\nabla_{\boldsymbol{x}} D_{\theta_d}(\boldsymbol{x})||_2 - 1)^2$ where, input $\boldsymbol{x}$ are real image features, generated image features and random samples on a straight line connecting real image features and generated image features (Gulrajani *et al.* (2017)). The parameter λ_{gp} controls the importance of the Lipschitz constraint. The discriminator parameters are trained using only seen category image features since $\boldsymbol{x}^u$ are unavailable.

Feature Classifier: The category classifier has parameters θ_d^c and is denoted as $D_{\theta_d^c}$. It is a softmax cross-entropy classifier for the generated and real image features $\boldsymbol{x}^s, \tilde{\boldsymbol{x}}^s, \tilde{\boldsymbol{x}}^u$ and is trained to minimize the loss $\mathcal{L}_c$. For ease of notation the real/generated image features are represented as $\boldsymbol{x}$ and corresponding labels y

$$\mathcal{L}_c = \min_{\theta_g, \theta_d^c} -\mathbb{E}_{(\boldsymbol{x},y)\sim(\mathcal{X},\mathcal{Y})}\Big[\sum_{c=1}^{C} 1(y=c)\log(D_{\theta_d^c}(\boldsymbol{x}))_c\Big], \tag{5.3}$$

where $(D_{\theta_d^c}(\boldsymbol{x}))_c$ is the c-th component of the C-dimension softmax output and $1(y=c)$ is the indicator function. While the discriminator performs a marginal alignment of real and generated features, the classifier performs category based conditional alignment.

So far, in the proposed WGAN model the *Generator* generates image features from seen and unseen categories. The *Discriminator* aligns the image feature distributions and the *Classifier* performs image classification. The LsrGAN model is illustrated in Fig. 5.2. In the following will introduce a novel regularization technique that transfers knowledge across the semantic and image feature spaces for the unseen and seen categories respectively.

5.2.2 Semantic Relationship Regularization

Conventional zero-shot learning approaches only use the seen classes in the training process when generating image features (Xian *et al.* (2018b); Zhu *et al.* (2018); Li *et al.* (2019)). This hinders the learning capability of the generator since there is no knowledge about the unseen classes during the training phase. Moreover, this also leads to overfitting the generator towards the seen classes leading to poor performance in generalized zero-shot learning. The proposed model aim to mitigate these issues by having a novel regularization procedure that will explicitly transfer knowledge of

unseen classes from the semantic domain and guide the generator in generating seen and unseen image features. I term this the "Semantic Regularized loss (SR-Loss)".

Since the visual and semantic feature spaces share a common underlying latent space that generates the visual and semantic features. I propose to exploit this relationship by transferring knowledge from the semantic space to the visual space to generate image features. Knowing the inter-class relationships in the semantic space can help us impose the same relationship constraints among the generated visual features. This is the idea behind the SR-Loss in the WGAN where we transfer inter-class relationships from the semantic domain to the visual domain. Fig. 5.1 illustrates this concept. The visual similarity between class c_i and c_j is represented as $\mathcal{X}_{sim}(\boldsymbol{\mu}_{c_i}, \boldsymbol{\mu}_{c_j})$, where $\boldsymbol{\mu}_c$ is the mean of the image features of class c. Note that for visual similarity we are considering the relationship between the class centers and not between individual image features. Likewise, the semantic similarity between class c_i and c_j is represented as $\mathcal{T}_{sim}(\boldsymbol{t}_{c_i}, \boldsymbol{t}_{c_j})$. For semantic similarity, since there is only one semantic vector $\boldsymbol{t}_c$ available for the every category. I have not considered the mean value of it, although the proposed approach can be extended to include multiple semantic feature vectors. I impose the following semantic relationship constraint for the image features,

$$\mathcal{T}_{sim}(\boldsymbol{t}_{c_i}, \boldsymbol{t}_{c_j}) - \epsilon_{ij} \;\leq\; \mathcal{X}_{sim}(\boldsymbol{\mu}_{c_i}, \boldsymbol{\mu}_{c_j}) \;\leq\; \mathcal{T}_{sim}(\boldsymbol{t}_{c_i}, \boldsymbol{t}_{c_j}) + \epsilon_{ij}, \tag{5.4}$$

where, hyper-parameter $\epsilon_{ij} \geq 0$ is a soft margin enforcing the similarity between semantic and image features for classes i and j. Large values of ϵ_{ij} allow for more deviation between semantic similarities and visual similarities. The constraints are incorporated into the objective by applying the penalty method (Lillo *et al.* (1993)),

$$\begin{aligned} p_{c_{ij}} \big[&|| \max(0, \mathcal{X}_{sim}(\boldsymbol{\mu}_{c_i}, \boldsymbol{\mu}_{c_j}) - (\mathcal{T}_{sim}(\boldsymbol{t}_{c_i}, \boldsymbol{t}_{c_j}) + \epsilon_{ij}))||^2 \\ &+ || \max(0, (\mathcal{T}_{sim}(\boldsymbol{t}_{c_i}, \boldsymbol{t}_{c_j}) - \epsilon_{ij}) - \mathcal{X}_{sim}(\boldsymbol{\mu}_{c_i}, \boldsymbol{\mu}_{c_j}))||^2 \big], \end{aligned} \tag{5.5}$$

where, $p_{c_{ij}}$ is the penalty for violating the constraint. The penalty becomes zero when the constraints are satisfied and is non-zero otherwise.

Mainly the goal is to transfer semantic inter-class relationships to enhance the image feature representations that are output from the generator. Consider a seen class c_i. I will estimate its semantic similarity $\mathcal{T}_{sim}(\boldsymbol{t}_{c_i}, \boldsymbol{t}_{c_j})$ with all the other seen semantic features $\boldsymbol{t}_{c_j}$ where $j \in \{1, \ldots, C_s\} \wedge j \neq i$. Not all the similarities are important and for the ease of implementation the highest n_c similarities are considered. Let I_{c_i} represent the set of n_c seen categories with the highest semantic similarity with c_i. I will apply Eq 5.5 to train the generator to output image features that satisfy the semantic similarity constraints against the top n_c similarities from the seen categories. For the seen image categories, the objective function is,

$$\mathcal{L}^s_{sr} = \min_{\theta_g} \frac{1}{C_s} \sum_{i=1}^{C_s} \sum_{j \in I_{c_i}} \big[|| \max(0, \mathcal{X}_{sim}(\boldsymbol{\mu}^s_{c_j}, \tilde{\boldsymbol{\mu}}^s_{c_i}) - (\mathcal{T}_{sim}(\boldsymbol{t}^s_{c_j}, \boldsymbol{t}^s_{c_i}) + \epsilon)) ||^2 + || \max(0, (\mathcal{T}_{sim}(\boldsymbol{t}^s_{c_j}, \boldsymbol{t}^s_{c_i}) - \epsilon) - \mathcal{X}_{sim}(\boldsymbol{\mu}^s_{c_j}, \tilde{\boldsymbol{\mu}}^s_{c_i})) ||^2 \big], \tag{5.6}$$

where, penalty $p_{ij} = 1$, and $\boldsymbol{\mu}^s_{c_j} := \mathbb{E}_{(\boldsymbol{x}, y=c_j) \sim (\mathcal{X}^s, \mathcal{Y}^s)}[\boldsymbol{x}]$ is the mean of the image features for seen class c_j, and $\tilde{\boldsymbol{\mu}}^s_{c_i} := \mathbb{E}_{\boldsymbol{z} \sim \mathcal{Z}}[G_{\theta_g}(\boldsymbol{z}, \boldsymbol{t}^s_{c_i})]$ is the mean of the generated image features of seen class c_i. A constant value of ϵ is considered as the soft margin to simplify the solution. Similarly, the objective function for the unseen categories is,

$$\mathcal{L}^u_{sr} = \min_{\theta_g} \frac{1}{C_u} \sum_{i=C_s+1}^{C} \sum_{j \in I_{c_i}} \big[|| \max(0, \mathcal{X}_{sim}(\boldsymbol{\mu}^s_{c_j}, \tilde{\boldsymbol{\mu}}^u_{c_i}) - (\mathcal{T}_{sim}(\boldsymbol{t}^s_{c_j}, \boldsymbol{t}^u_{c_i}) + \epsilon_{ij})) ||^2 + || \max(0, (\mathcal{T}_{sim}(\boldsymbol{t}^s_{c_j}, \boldsymbol{t}^u_{c_i}) - \epsilon_{ij}) - \mathcal{X}_{sim}(\boldsymbol{\mu}^s_{c_j}, \tilde{\boldsymbol{\mu}}^u_{c_i})) ||^2 \big], \tag{5.7}$$

where, $\tilde{\boldsymbol{\mu}}^u_{c_i} := \mathbb{E}_{\boldsymbol{z} \sim \mathcal{Z}}[G_{\theta_g}(\boldsymbol{z}, \boldsymbol{t}^u_{c_i})]$ is the mean of the generated image features of unseen class c_i.

5.2.3 LsrGAN Objective Function

The LsrGAN leverages the semantic relationship between seen and unseen categories to generate robust image features for unseen categories using the objective function defined in Eq. 5.6 and 5.7. The model generates robust seen image features conditioned by the regularizer in Eq. 5.1. The LsrGAN trains a classifier over all the C categories as outlined in Eq. 5.3. The LsrGAN is based on a WGAN model that aligns the image feature distributions using the objective function defined in Eq. 5.2. The overall objective function of the LsrGAN model is given by,

$$\lambda_c \mathcal{L}_c + \mathcal{L}_d + \lambda_{vp} \mathcal{L}_{vp} + \lambda_{sr} (\mathcal{L}^s_{sr} + \mathcal{L}^u_{sr}) \tag{5.8}$$

where, λ_c, λ_{vp} and λ_{sr} are hyper parameters controlling the importance of each of the loss terms. Unlike standard zero-shot learning models that generate image features and then have to train a supervised classifier (Xian *et al.* (2018b); Zhu *et al.* (2018); Li *et al.* (2019)), the LsrGAN model has an inbuilt classifier that can also be used for evaluating zero-shot learning and generalized zero shot learning.

5.3 Training Algorithm

Below, illustrated the training procedure for the LsrGAN model. Mainly, the Generator (G_{θ_g}) and Discriminator (D_{θ_d}) are trained alternately with the Adam optimizer. Notice that the training of LsrGAN contains two phases, one for the seen classes and another for the unseen classes.

Algorithm 1 Training procedure for the LsrGAN

Input: number of epochs N_E, the batch size m, discriminator iterations $N_d = 5$ for seen classes, loss hyper parameters λ_c, λ_{vp} and λ_{sr}, $N_c = 1$ or 2 discriminator iterations for unseen classes, and Adam parameters $\beta_1 = 0.5$ and $\beta_2 = 0.9$.

for iter $= 1, ..., N_E$ **do**

  // Seen Class Training

  for $i = 1, ..., N_d$ **do**

    Minibatch sampling from $\mathcal{T}^s$ with matching images from $\mathcal{X}^s$ and noise $\mathcal{Z}$

    $\tilde{\boldsymbol{x}} \leftarrow G_{\theta_g}(\boldsymbol{t}^s, \mathcal{Z})$

    Discriminator and classifier loss computation $\mathcal{L}_d$ and $\mathcal{L}_c$ using Eq. 2 and 3

    $\theta_d \leftarrow \text{Adam}(\nabla_{\theta_d^r}\mathcal{L}_d, \bigtriangledown_{\theta_d^c}\mathcal{L}_c, \theta_d, \lambda_c, \beta_1, \beta_2)$

  end for

  Minibatch sampling from $\mathcal{T}^s$ and noise $\mathcal{Z}$

  Generator loss computation L_G using Eq. 8

  $\tilde{\boldsymbol{x}} \leftarrow G_{\theta_g}(\boldsymbol{t}^s, \mathcal{Z})$

  $\theta_g \leftarrow \text{Adam}(\nabla_{\theta_g}\mathcal{L}_d, \bigtriangledown_{\theta_g}\mathcal{L}_{vp}, \bigtriangledown_{\theta_g}\mathcal{L}_c, \bigtriangledown_{\theta_g}\mathcal{L}^s_{sr}, \theta_g, \lambda_c, \lambda_{vp}, \lambda_{sr}, \beta_1, \beta_2)$

  // Unseen Class Training

  for $i = 1, ..., N_c$ **do**

    Minibatch sampling from $\mathcal{T}^u$ and noise $\mathcal{Z}$

    $\tilde{\boldsymbol{x}} \leftarrow G_{\theta_g}(\boldsymbol{t}^u, \mathcal{Z})$

    Classifier loss computation $\mathcal{L}_c$ using Eq. 3

    $\theta_d^c \leftarrow \text{Adam}(\nabla_{\theta_d^c}\mathcal{L}_c, \theta_d^c, \lambda_c, \beta_1, \beta_2)$

  end for

  Minibatch sampling from $\mathcal{T}^u$ and noise $\mathcal{Z}$

  Generator loss computation L_G using Eq. 8

  $\tilde{\boldsymbol{x}} \leftarrow G_{\theta_g}(\boldsymbol{t}^u, \mathcal{Z})$

  $\theta_g \leftarrow \text{Adam}(\nabla_{\theta_g}\mathcal{L}_c, \bigtriangledown_{\theta_g}\mathcal{L}^u_{sr}, \theta_g, \lambda_c, \lambda_{sr}, \beta_1, \beta_2)$

end for=0

5.4 Experiments & Results

This section will discuss the various experiments that are conducted to showcase the superiority of the LsrGAN with previous state of the art models.

5.4.1 Datasets

In order to evaluate the LsrGAN, I have considered a seven benchmark datasets. These datasets include either Attribute-based or Wikipedia-based semantic information. The details of the datasets are already mentioned in chapter 2.

5.4.2 Implementation Details and Performance Metrics

The 2048-dimensional ResNet-101 (He *et al.* (2016)) features are considered as a real visual feature for attribute-based datasets, and part-based features (e.g., belly, leg, wing, etc.) from VPDE-net (Zhang *et al.* (2016)) are used for the Wikipedia-based datasets, as suggested by (Zhu *et al.* (2018); Xian *et al.* (2018b)). I have utilized the TF-IDF to extract the features from the Wikipedia descriptions. For a fair comparison, all of the experiment settings are kept the same as reported in (Xian *et al.* (2018a); Zhu *et al.* (2018); Xian *et al.* (2018b)).

The base block of the proposed model is GAN, which is implemented using a multi-layer perceptron. Specifically, the feature generator G_{θ_g} has one hidden unit having 4096 neurons and LeakyReLU as an activation function. For attribute-based datasets, I intend to get the top max-pooling units of ResNet - 101 (visual features). Hence, the output layer has ReLU activation in the feature generator. On the other hand, for the Wikipedia-based datasets, I have used Tanh as an output activation for the feature generator since the VPDE-net feature varies from -1 to 1. $\mathcal{Z}$ is sampled from the normal Gaussian distribution. To perform the denoising and dimensionality

reduction from Wikipedia descriptions, a fully connected layer is employed with a feature generator. Also, notice that the semantic similarity for the SR-Loss is computed using the denoiser's output in Wikipedia-based datasets and it will be discarded when dealing with the attribute-based datasets. The discriminator D_{θ_d} of LsrGAN has two branches. One is used to play the real/fake game, and the other performs the classification on the generated/real visual feature. The discriminator also has 4096 units in the hidden layer with ReLU as an activation. Since the cosine distance is less prone to the curse of dimensionality when the features are sparse (semantic features), I have considered it as a distance measure for the SR-loss.

To perform the Zero shot recognition I have used nearest neighbor prediction on datasets having Wikipedia descriptions, and the classifier attached to the discriminator for the attributes based recognition. The Top-1 accuracy is used to assess the ZSL setting. Furthermore, to capture the more realistic scenario, I have also examined the Generalized Zero-shot recognition performance. As suggested by (Chao *et al.* (2016)), the area under the seen and unseen curve (AUC score) is considered as GZSL performance metric for Wikipedia-based datasets, and the harmonic mean of the seen and unseen Top-1 accuracies is reported for Attribute-based dataset. Notice that the choice of these measures and predictions models is to make a fair comparison with existing methods.

5.4.3 ZSL and GZSL Performance

The results for the ZSL are provided in the left part of Table 5.1 and 5.2. It can be seen that LsrGAN achieves superior performance in both attribute and Wikipedia-based datasets compared to the previous state of the art models, especially with generative models GAZSL, F-GAN, cycle-CLSWGAN, and LisGAN. It is worth noticing that all the mentioned generative models have the same base architecture. e.g.,

Table 5.1: ZSL and GZSL results on AWA, CUB, and SUN with attributes as semantic information. T1 indicates the Top-1 % accuracy in the ZSL setting. On the other hand, "U", "S" and "H" denotes the Top-1% accuracy for the unseen, seen, and Harmonic mean (seen + unseen).

Methods	Zero Shot Learning			Generalized Zero Shot Learning								
	AWA	CUB	SUN	AwA			CUB			SUN		
	T1	T1	T1	U	S	H	U	S	H	U	S	H
DAP (Lampert *et al.* (2013))	44.1	40.0	39.9	0.0	88.7	0.0	1.7	67.9	3.3	4.2	25.2	7.2
CONSE (Norouzi *et al.* (2013b))	45.6	34.3	38.8	0.4	88.6	0.8	1.6	72.2	3.1	6.8	39.9	11.6
SSE (Zhang and Saligrama (2015))	60.1	43.9	51.5	7.0	80.5	12.9	8.5	46.9	14.4	2.1	36.4	4.0
DeViSE (Frome *et al.* (2013))	54.2	50.0	56.5	13.4	68.7	22.4	23.8	53.0	32.8	16.9	27.4	20.9
SJE (Akata *et al.* (2015b))	65.6	53.9	53.7	11.3	74.6	19.6	23.5	59.2	33.6	14.7	30.5	19.8
ESZSL (Romera-Paredes and Torr (2015))	58.2	53.9	54.5	5.9	77.8	11.0	2.4	70.1	4.6	11.0	27.9	15.8
ALE (Akata *et al.* (2015a))	59.9	54.9	58.1	14.0	81.8	23.9	4.6	73.7	8.7	21.8	33.1	26.3
SYNC (Changpinyo *et al.* (2016))	54.0	55.6	56.3	10.0	**90.5**	18.0	7.4	66.3	13.3	7.9	**43.3**	13.4
SAE (Kodirov *et al.* (2017))	53.0	33.3	40.3	1.1	82.2	2.2	0.4	**80.9**	0.9	8.8	18.0	11.8
DEM (Zhang *et al.* (2017a))	68.4	51.7	61.9	30.5	86.4	45.1	11.1	75.1	19.4	20.5	34.3	25.6
TCN (Jiang *et al.* (2019))	70.3	59.5	61.5	49.4	76.5	60.0	52.6	52.0	52.3	31.2	37.3	34.0
GAZSL (Zhu *et al.* (2018))	68.2	55.8	61.3	19.2	86.5	31.4	23.9	60.6	34.3	21.7	34.5	26.7
F-GAN (Xian *et al.* (2018b))	68.2	57.3	60.8	**57.9**	61.4	59.6	43.7	57.7	49.7	42.6	36.6	39.4
cycle-CLSWGAN (Felix *et al.* (2018))	66.3	58.4	60.0	56.9	64.0	60.2	45.7	61.0	52.3	**49.4**	33.6	40.0
LisGAN (Li *et al.* (2019))	**70.6**	58.8	61.7	52.6	76.3	62.3	46.5	57.9	51.6	42.9	37.8	40.2
LsrGAN	66.4	**60.3**	**62.5**	54.6	74.6	**63.0**	**48.1**	59.1	**53.0**	44.8	37.7	**40.9**

WGAN. Hence, the superiority of LsrGAN suggests that the motivation behind this research work is realistic, and experiments are effective. In summary, LsrGAN achieves, 1.5 %, 3.9 %, 0.8 %, 0.44% improvement on CUB (Easy), CUB (Hard), NAB (Easy) and NAB (Hard) respectively for the Wikipedia-based datasets under ZSL. On the other hand, for the attribute-based ZSL, the LsrGAN attains 1.5% and 0.8% improvement on CUB and SUN, respectively. A little lower ZSL performance achieved on AWA is probably due to the high feature correlation between seen and unseen classes, e.g. Rat (unseen) and Mouse (seen). Notice that for ZSL and GZSL with attributes, I am using the classifier associated with the discriminator having all the classes. Although, in ZSL setting, I find the highest confidence over the unseen

Table 5.2: ZSL and GZSL results on CUB and NAB datasets with Wikipedia descriptions as semantic information on the two-split setting. We have used Top-1 % accuracy and Seen-Unseen AUC (%) for ZSL and GZSL, respectively.

Methods	Zero Shot Learning				Generalized Zero Shot Learning			
	CUB		NAB		CUB		NAB	
	Easy	Hard	Easy	Hard	Easy	Hard	Easy	Hard
WAC-Linear (Elhoseiny *et al.* (2013))	27.0	5.0	-	-	23.9	4.9	23.5	-
WAC-Kernal (Elhoseiny *et al.* (2016))	33.5	7.7	11.4	6.0	14.7	4.4	9.3	2.3
ESZSL (Romera-Paredes and Torr (2015))	28.5	7.4	24.3	6.3	18.5	4.5	9.2	2.9
ZSLNS (Qiao *et al.* (2016))	29.1	7.3	24.5	6.8	14.7	4.4	9.3	2.3
Sync-fast (Changpinyo *et al.* (2016))	28.0	8.6	18.4	3.8	13.1	4.0	2.7	3.5
ZSLPP (Elhoseiny *et al.* (2017))	37.2	9.7	30.3	8.1	30.4	6.1	12.6	3.5
GAZSL (Zhu *et al.* (2018))	43.7	10.3	35.6	8.6	35.4	8.7	20.4	5.8
LsrGAN	**45.2**	**14.2**	**36.4**	**9.04**	**39.5**	**12.1**	**23.2**	**6.4**

classes, the availability of seen classes affects the prediction capability of the classifier for ZSL. This could be the potential reason behind the substandard ZSL performance on AWA. However, it is worth noticing that the GZSL result for the same dataset is superior.

The primary focus behind this work is to elevate the GZSL performance, which is apparent from the right side of Table 5.1 and 5.2. Following (Xian *et al.* (2018a); Zhu *et al.* (2018); Xian *et al.* (2018b)), the harmonic mean and AUC score are reported for the attribute and Wikipedia-based datasets, respectively. The mentioned metrics help to showcase the approach's generalizability as the harmonic mean and AUC scores are only high when the performance on seen and unseen classes is high. From the results, it is evident that the LsrGAN outperforms the previous state of the art for the GZSL. In terms of numbers, LsrGAN achieves, 4.1%, 3.4% 2.8% and 0.6% gain on CUB (Easy), CUB (Hard), NAB (Easy) and NAB (Hard) respectively for the Wikipedia-based datasets and 0.7 %, 1.4% and 0.7 % improvement on attribute-based

AWA, CUB and SUN respectively. It is worth noticing that the LsrGAN improves the unseen Top-1 performance in the GZSL setting for the attribute-based CUB and SUN by 1.6% and 1.9% with the previous state of the art LisGAN (Li *et al.* (2019)).

The majority of the conventional approaches, including the generative models, overfit the seen classes as they ignore the utilization of the unseen semantic features during the training process. Consequently, these models suffer from the domain shift problem, which results in lower GZSL performance. Notice that most of the approaches mentioned in Table 5.1 achieve very high performance on seen classes compared to the unseen classes in the GZSL setting. For example, SYNC (Changpinyo *et al.* (2016)) has around 90% recognition capability on the seen classes, and it drops to only 10% (80% difference) for the unseen classes on AWA dataset. It is also evident from Tables 5.1 and 5.2 that the performance on the unseen categories drops drastically when the search space includes both seen and unseen classes in the GZSL setting. For instance, DAP (Lampert *et al.* (2013)) drops from 40 % to 1.7 %, GAZSL (Zhu *et al.* (2018)) drops from 55.8% to 23.9% and F-GAN (Xian *et al.* (2018b)) drops from 57.3% to 43.7 % on attribute-based CUB dataset. This indicates that the previous approaches are easy to overfit the seen classes. Although the generative models have achieved significant progress compared to the previous embedding methods, they still possess the overfitting issue towards seen classes by having substandard GZSL performance. On the contrary, LsrGAN incorporates the novel SR-Loss that enables the utilization of the unseen semantics in the training process itself, leading to explicit knowledge transfer from the similar seen classes to the unseen classes. Therefore, the proposed LsrGAN alleviates the overfitting concern and helps to achieve a state of the art GZSL performance. It is worth noticing that LsrGAN not only outperforms the generative zero-shot models having single GAN but also proves its worth against cycle-GAN (Felix *et al.* (2018)) in ZSL and GZSL

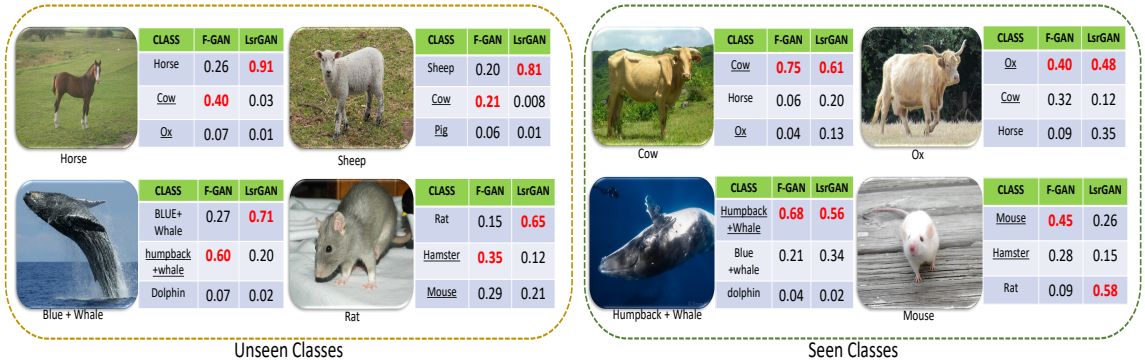


Figure 5.3: Average class confidence score (Avg SoftMax Probability) comparison for classifier trained with F-GAN and LsrGAN. Top 3 average guesses are mentioned here. The red marked label showcase the top 1 average guess. The class names with underline represent seen classes.

setting. As explained above, I owe the success of the LsrGAN model to the SR-loss for enabling explicit knowledge transfer from similar seen classes to the unseen classes. Lastly, to have fair comparisons, I have taken the performance numbers from (Xian *et al.* (2018a,b); Zhu *et al.* (2018)).

5.4.4 Effectiveness of SR-Loss

Utilizing the semantic relationship between seen and unseen classes to infer the visual characteristics of an unseen class is at the heart of the proposed SR-Loss. Contrary to other generative approaches, it enables explicit knowledge transfer in the generative model to make it learn from the unseen classes together with seen classes during the training process itself. As a result, the LsrGAN will become more robust towards the unseen classes leading to address the seen class overfitting concern. To demonstrate such ability, I have computed the average class confidence score (avg Softmax probabilities) of the classifier trained with the generated features from the LsrGAN or F-GAN model. Since the confidence scores are computed under the GZSL, the classifier's training set contains real visual features from the seen classes and generated features from LsrGAN or F-GAN of the unseen classes. To have a

fair comparison, I have used the same F-GAN's Softmax classifier in LsrGAN. Since LsrGAN learns from the seen and unseen classes during the training process itself, the Softmax classifier associated with it is not trained in an offline fashion like in the F-GAN model.

Fig. 5.3 depicts the confidence results on the AWA dataset under the GZSL setting. I have taken mainly four confusing seen and unseen classes for the comparison with classifier's top 3 guesses. It is evident from the figure that the classifier trained with the F-GAN has lower confidence for the unseen classes, and it mainly showcases very high confidence towards the similar seen classes even if the test image comes from the unseen classes. Also, during the seen class classification, the classifier's confidence values are mainly distributed among the seen classes in F-GAN. For instance, "mouse" has its confidence spread between mouse and hamster only. On the other hand, LsrGAN showcase decent confidence values for the seen and unseen, both leading to better GZSL performance. It is worth noticing that the LsrGAN fails for the "mouse" classification. However, the confidence is well spread among all the three categories showing it has considered an unseen class "rat" with other seen classes "mouse" and "Hamster". These observations reflect the fact that F-GAN has an overfitting issue towards the seen classes. On the other hand, the balanced performance of LsrGAN manifests that explicit knowledge transfer from SR-loss helps it to overcome the overfitting concern towards the seen classes. To bolster the claim further, I have also computed the avg. class confidence across all the seen and unseen classes for these two models on attribute-based AWA, CUB, and SUN datasets. Table 5.3 reports avg. confidence values for seen and unseen classes. Clearly, it shows the superiority of LsrGAN in terms of generalizability compared to F-GAN.

Table 5.3: Comparison of avg. class confidence score across all seen or unseen classes (SoftMax Probability) between F-GAN and LsrGAN for attribute-based AWA, CUB and SUN

	F-GAN (Xian *et al.* (2018b))		LrsGAN	
	Unseen	Seen	Unseen	Seen
AWA	0.29	0.86	0.69	0.83
CUB	0.33	0.65	0.60	0.64
SUN	0.32	0.35	0.65	0.39

(a) ϵ (CUB E) (b) λ_{sr} (AWA)

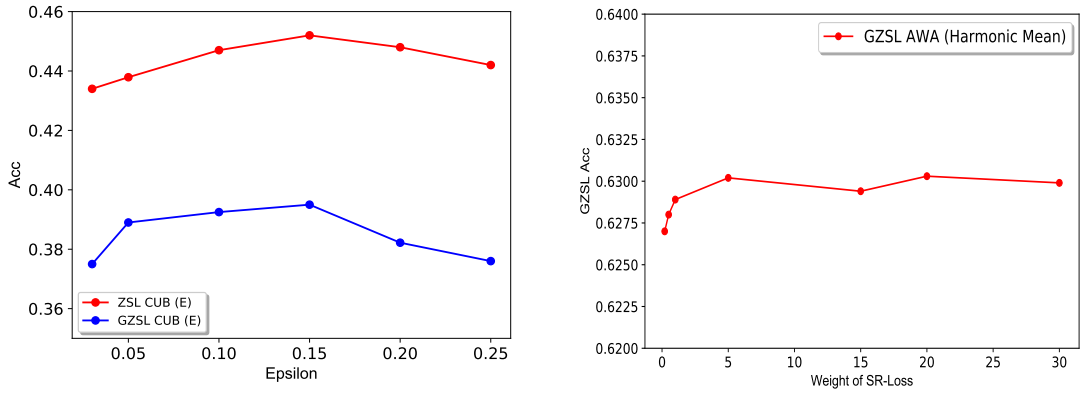
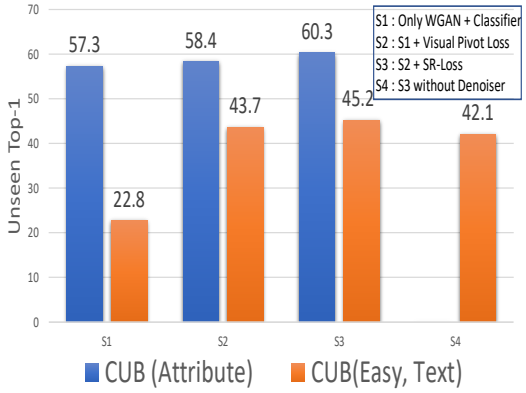


Figure 5.4: Parameter Sensitivity (a-b) of ϵ and λ_{sr} for SR-loss.

5.4.5 Model Analysis

Parameter Sensitivity : I have tuned the LsrGAN parameters by following the conventional grid search approach. Mainly the SR-Loss parameters ϵ, λ_{sr} and n_c are considered for the tuning. For the fair comparison, I have adopted other parameters λ_{vp}, λ_{gp} from (Xian *et al.* (2018b); Zhu *et al.* (2018)), also the λ_c is considered between $(0, 1]$, specifically, 0.01 for the majority of experiments. Fig. 5.4 (a)-(b) and Fig. 5.5 (b) show the parameter sensitivity for the SR-Loss. Notice that I estimate the ϵ value from the seen class visual and semantic relations. It can be seen that a lower and

(a) Ablation study



(b) n_c (CUB E)

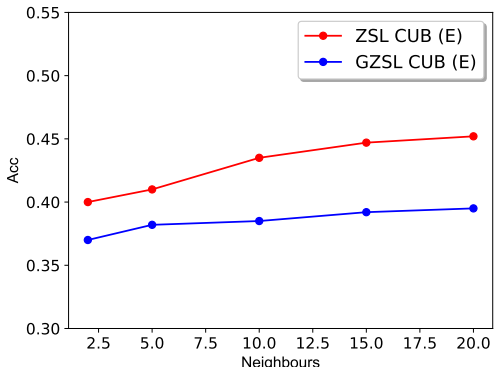


Figure 5.5: Ablation Study of LsrGAN (a), and Parameter Sensitivity of n_c (b) for SR-loss.

higher value of ϵ affects the performance. On the other hand, λ_{sr} and n_c maintain consistent performance after reaching a certain threshold value.

Training Stability : Since GANs are notoriously hard to train, and the proposed LsrGAN model not only uses GAN but also optimizes the similarity constraints from the SR-loss. Therefore, reporting the training stability for ZSL and GZSL for attribute and Wikipedia-based datasets. Specifically, for the ZSL, I have considered the unseen Top-1 accuracy and Epoch behavior in Fig. 5.6 (a) and Fig 5.7 (a). The harmonic mean of the seen and unseen Top-1 accuracy and Epoch behaviour is mentioned in Fig. 5.6 (b) and Fig. 5.7 (a) to showcase the training stability for the GZSL. Mainly we see the stable performance across all the datasets.

Ablation study : To showcase the effectiveness of the every component in the LsrGAN model. I have reported the Ablation study in Fig. 5.5 (a) under ZSL for both attribute and Wikipedia-based CUB. Primarily, I have used CUB (Easy) split for the Wikipedia-based dataset. The $S1$ - WGAN with a classifier is considered as a baseline model. $S2$ reflects the effect of the visual pivot regularizer in LsrGAN model. Finally, $S3$ showcase the performance of a complete LsrGAN with SR-loss.

(a) ZSL Text (b) GZSL Text

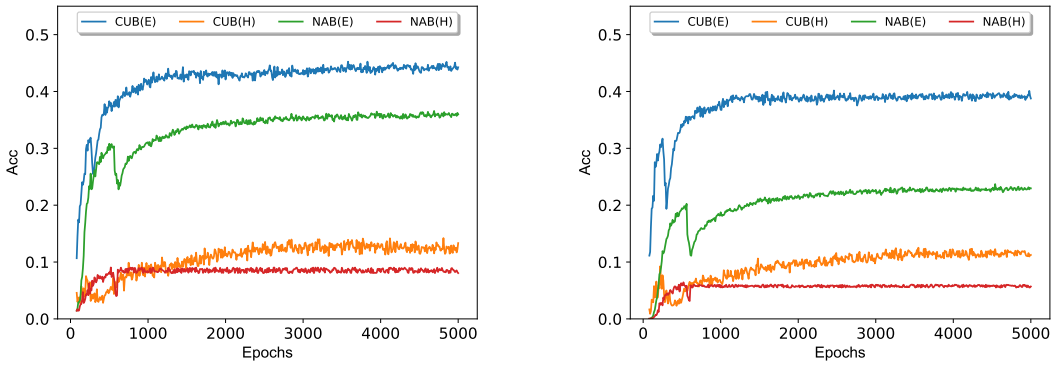


Figure 5.6: Training Stability for Wikipedia-based Datasets Under ZSL and GZSL

(a) ZSL-GZSL Attribute

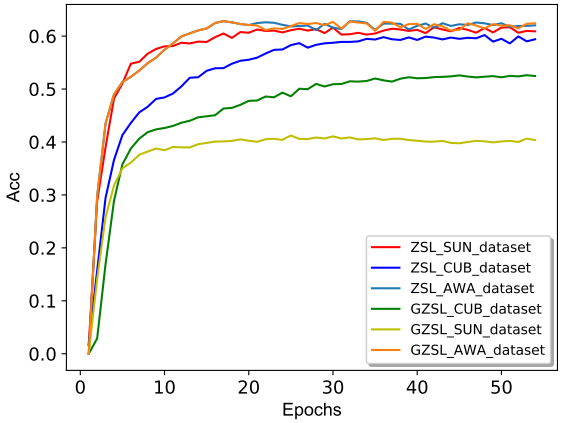


Figure 5.7: Training Stability for Attribute-based Datasets Under ZSL and GZSL

To highlight the effect of the denoiser used to process the Wikipedia-based features, I have also reported the LsrGAN without denoiser in $S4$. In summary, Fig. 5.5 (a) showcases the importance of each component used in LsrGAN model.

Chapter 6

CONCLUSION & FUTURE WORK

6.1 Conclusion

In this dissertation, I have proposed a two novel generative zero-shot models named Student- Teacher GAN and LsrGAN. The Student- Teacher based GAN uses the concepts of meta learning to mimic the zero-shot inference behaviour in the training process itself. On the other hand, the LsrGAN, Leverages the Semantic Relationship between seen and unseen classes to address the seen class overfitting concern in generative zero-shot models. Mainly, LsrGAN employs a novel Semantic Regularized Loss (SR-Loss) to perform explicit knowledge transfer from seen classes to unseen ones. The SR-Loss explores the semantic relationships between seen and unseen classes to guide the LsrGAN to generate visual features that mirror the same relationship. Extensive experiments conclude that the Student - Teacher based GAN is not improving the results to a greater extend. On the other hand, LsrGAN showcases a state of the art performance in ZSL and GZSL for all the seven benchmarks datasets, including attribute and Wikipedia description based datasets. This verifies that the propoesd LsrGAN effectively addresses the overfitting concern of the generative zero-shot models, and comes out as a more robust model for the GZSL.

6.2 Future Research Directions

Here, in this section will highlight some of the future research directions to extend my work.

6.2.1 Student-Teacher GAN Model

First of all, the Student-Teacher model is highly complex. Imaging training more than six GANs and fine-tuning them to get the desired performance; it is certainly hard to achieve. One way to address this concern is to train a single student incrementally and simultaneously make the Teacher network learn from it using the same Meta Learning-based episodic training procedure. I would also recommend exploring the recent Meta-Learning work such as "MAML" to devise an approach to train these Student-Teacher networks. Lastly, it is also vital how one splits the data into a train set and a held-out set for the student model. The splitting should have some common relationship between class categories like the EASY and HARD splits mentioned in the chapter 2.

6.2.2 LsrGAN Model

In my opinion, LsrGAN has many advantages compared to other generative models. Specifically, it only uses one GAN and hence easier to fine-tune. Although the constraint optimization is complex together with notorious GANs, the performance results state the GAN can handle these constraints well. I would only like to explore various distance metrics in SR-loss; also, it would worth build an attention model to automatically find the neighbors in the semantic and visual space and learn this model together with the LsrGAN. I believe it will have less dimensionality curse issues compared to cosine distance and Euclidian distance. Moreover, one can also devise a

network to determine the ideal value of ϵ for every category. I believe different values of ϵ can help to learn categories better since it is hard to generalize the value of ϵ for all the categories.

BIBLIOGRAPHY

Akata, Z., F. Perronnin, Z. Harchaoui and C. Schmid, "Label-embedding for image classification", IEEE transactions on pattern analysis and machine intelligence **38**, 7, 1425–1438 (2015a).

Akata, Z., 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 (2015b).

Al-Halah, Z., M. Tapaswi and R. Stiefelhagen, "Recovering the missing link: Predicting class-attribute associations for unsupervised zero-shot learning", in "Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition", pp. 5975–5984 (2016).

Arjovsky, M., S. Chintala and L. Bottou, "Wasserstein GAN", arXiv preprint arXiv:1701.07875 (2017).

Atzmon, Y. and G. Chechik, "Adaptive confidence smoothing for generalized zero-shot learning", arXiv pp. arXiv–1812 (2018).

Baldi, P., "Autoencoders, unsupervised learning, and deep architectures", in "Proceedings of ICML workshop on unsupervised and transfer learning", pp. 37–49 (2012).

Changpinyo, S., W.-L. Chao, B. Gong and F. Sha., "Synthesized classifiers for zero-shot learning", CVPR pp. 5327–5336 (2016).

Changpinyo, S., W.-L. Chao and F. Sha, "Predicting visual exemplars of unseen classes for zero-shot learning", in "Proceedings of the IEEE international conference on computer vision", pp. 3476–3485 (2017).

Chao, W.-L., S. Changpinyo, B. Gong and F. Sha, "An empirical study and analysis of generalized zero-shot learning for object recognition in the wild", in "European Conference on Computer Vision", pp. 52–68 (Springer, 2016).

Chen, Z., J. Li, Y. Luo, Z. Huang and Y. Yang, "Canzsl: Cycle-consistent adversarial networks for zero-shot learning from natural language", in "The IEEE Winter Conference on Applications of Computer Vision", pp. 874–883 (2020).

Devlin, J., M.-W. Chang, K. Lee and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding", arXiv preprint arXiv:1810.04805 (2018).

Ding, Z., M. Shao and Y. Fu, "Low-rank embedded ensemble semantic dictionary for zero-shot learning", in "Proceedings of the IEEE conference on computer vision and pattern recognition", pp. 2050–2058 (2017).

Elhoseiny, M., A. Elgammal and B. Saleh, "Write a classifier: Predicting visual classifiers from unstructured text", PAMI (2016).

Elhoseiny, M., B. Saleh and A. Elgammal, "Write a classifier: Zero-shot learning using purely textual descriptions", in "ICCV", (2013).

Elhoseiny, M., Y. Zhu, H. Zhang and A. Elgammal, "Link the head to the "beak": Zero shot learning from noisy text description at part precision", in "CVPR", (2017).

Felix, R., V. Kumar, I. Reid and G. Carnerio, "Multi-model cycle-consistent generalized zero-shot leanring", ECCV (2018).

Frome, A., G. S. Corrado, J. Shlens, S. Bengio, J. Dean, T. Mikolov and et al, "Devise: A deep visual-semantic embedding model", NIPS pp. 2121–2129 (2013).

Fu, Y., T. M. Hospedales, T. Xiang and S. Gong, "Transductive multi-view zero-shot learning", IEEE transactions on pattern analysis and machine intelligence **37**, 11, 2332–2345 (2015).

Goodfellow, I., J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, "Generative adversarial nets", NIPS (2014).

Gulrajani, I., F. Ahmed, M. Arjovsky, V. Dumoulin and A. C. Courville, "Improved training of wasserstein GANs", in "Advances in neural information processing systems", pp. 5767–5777 (2017).

He, K., X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition", in "Proceedings of the IEEE conference on computer vision and pattern recognition", pp. 770–778 (2016).

Hu, R. L., C. Xiong and R. Socher, "Correction networks: Meta-learning for zero-shot learning", (2018).

Huang, H., C. Wang, P. S. Yu and C.-D. Wang, "Generative dual adversarial network for generalized zero-shot learning", in "Proceedings of the IEEE conference on computer vision and pattern recognition", pp. 801–810 (2019).

Jayaraman, D. and K. Grauman, "Zero-shot recognition with unreliable attributes", in "Advances in neural information processing systems", pp. 3464–3472 (2014).

Jiang, H., R. Wang, S. Shan and X. Chen, "Transferable contrastive network for generalized zero-shot learning", in "Proceedings of the IEEE International Conference on Computer Vision", pp. 9765–9774 (2019).

Kankuekul, P., 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 (IEEE, 2012).

Kodirov, E., T. Xiang and S. Gong, "Semantic autoencoder for zero-shot learning", in "Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition", pp. 3174–3183 (2017).

Lampert, C. H., H. Nickisch and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization", IEEE transactions on pattern analysis and machine intelligence **36**, 3, 453–465 (2013).

Larochelle, H., D. Erhan and Y. Bengio, "Zero-data learning of new tasks", AAAI (2008).

Lei Ba, J., K. Swersky, S. Fidler *et al.*, "Predicting deep zero-shot convolutional neural networks using textual descriptions", in "Proceedings of the IEEE International Conference on Computer Vision", pp. 4247–4255 (2015).

Li, J., M. Jin, K. Lu, Z. Ding, L. Zhu and Z. Huang, "Leveraging the invariant side of generative zero-shot learning", CVPR (2019).

Lillo, W. E., M. H. Loh, S. Hui and S. H. Zak, "On solving constrained optimization problems with neural networks : A penalty method approach", IEEE Transactions on neural networks **4**, 6, 931–940 (1993).

Mikolov, T., 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", pp. 3111–3119 (2013).

Miller, G. A., "Wordnet: a lexical database for english", Communications of the ACM **38**, 11, 39–41 (1995).

Norouzi, M., T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado and J. Dean, "Zero-shot learning by convex combination of semantic embeddings", arXiv preprint arXiv:1312.5650 (2013a).

Norouzi, M., T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. S. Corrado and J. Dean, "Zero-shot learning by convex combination of semantic embeddings", arXiv preprint arXiv:1312.5650 (2013b).

Palatucci, M., D. Pomerleau, G. E. Hinton and T. M. Mitchell, "Zero-shot learning with semantic output codes", in "Advances in neural information processing systems", pp. 1410–1418 (2009).

Patterson, G. and J. Hays, "Sun attribute database: Discovering, annotating, and recognizing scene attributes", CVPR (2012).

Paul, A., N. C. Krishnan and P. Munjal, "Semantically aligned bias reducing zero shot learning", in "Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition", pp. 7056–7065 (2019).

Pennington, J., R. Socher and C. D. Manning, "Glove: Global vectors for word representation", in "Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)", pp. 1532–1543 (2014).

Qiao, R., L. Liu, C. Shen and A. v. d. Hengel, "Less is more: Zero-shot learning from online textual documents with noise suppression", in "CVPR", (2016).

Radovanović, M., A. Nanopoulos and M. Ivanović, "Hubs in space: Popular nearest neighbors in high-dimensional data", Journal of Machine Learning Research **11**, Sep, 2487–2531 (2010).

Rohrbach, M., M. Stark and B. Schiele, "Evaluating knowledge transfer and zero-shot learning in a large-scale setting", in "CVPR 2011", pp. 1641–1648 (IEEE, 2011).

Rohrbach, M., M. Stark, G. Szarvas, I. Gurevych and B. Schiele, "What helps where– and why? semantic relatedness for knowledge transfer", in "2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition", pp. 910–917 (IEEE, 2010).

Romera-Paredes, B. and P. Torr, "An embarrassingly simple approach to zero-shot learning", ICML pp. 2152–2161 (2015).

Scheirer, W. J., A. de Rezende Rocha, A. Sapkota and T. E. Boult, "Toward open set recognition", IEEE transactions on pattern analysis and machine intelligence **35**, 7, 1757–1772 (2012).

Schonfeld, E., S. Ebrahimi, S. Sinha, T. Darrell and Z. Akata, "Generalized zero- and few-shot learning via aligned variational autoencoders", in "Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition", pp. 8247–8255 (2019).

Socher, R., M. Ganjoo, C. D. Manning and A. Ng, "Zero-shot learning through cross-modal transfer", in "Advances in neural information processing systems", pp. 935–943 (2013).

V. Verma, A. M., G. Arora and P. Rai, "Generalized zero-shot learning via synthesized examples", CVPR (2018).

Van Horn, G., S. Branson, R. Farrell, S. Haber, J. Barry, P. Ipeirotis, P. Perona and S. Belongie, "Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection", in "Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition", pp. 595–604 (2015).

Welinder, P., S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie and P. Perona, "Caltech-ucsd birds 200", (2010).

Xian, Y., Z. Akata, G. Sharma, Q. Nguyen, M. Hein and B. Schiele, "Latent embeddings for zero-shot classification", in "Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition", pp. 69–77 (2016).

Xian, Y., C. H. Lampert, B. Schiele and Z. Akata, "Zero-shot learning - a comprehensive evaluation of the good, the bad and the ugly", TPAMI (2018a).

Xian, Y., T. Lorenz, B. Schiele and Z. Akata, "Feature generating networks for zero shot learning", CVPR (2018b).

Xu, X., F. Shen, Y. Yang, D. Zhang, H. Tao Shen and J. Song, "Matrix tri-factorization with manifold regularizations for zero-shot learning", in "Proceedings of the IEEE conference on computer vision and pattern recognition", pp. 3798–3807 (2017).

Yu, X. and Y. Aloimonos, "Attribute-based transfer learning for object categorization with zero or one training example", ECCV (2010).

Zhang, H., T. Xu, M. Elhoseiny, X. Huang, S. Zhang, A. Elgammal and D. Metaxas, "Spda-cnn: Unifying semantic part detection and abstraction for fine-grained recognition", in "Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition", pp. 1143–1152 (2016).

Zhang, L., T. Xiang and S. Gong, "Learning a deep embedding model for zero-shot learning", in "Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition", pp. 2021–2030 (2017a).

Zhang, L., T. Xiang and e. a. Shaogang Gong, "Learning a deep embedding model for zero-shot learning", **32** (2017b).

Zhang, Z. and V. Saligrama, "Zero-shot learning via semantic similarity embedding", ICCV pp. 4166–4174 (2015).

Zhang, Z. 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).

Zhu, J.-Y., T. Park, P. Isola and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks", in "Proceedings of the IEEE international conference on computer vision", pp. 2223–2232 (2017).

Zhu, Y., M. Elhoseiny, B. Liu, X. Peng and A. Elgammal, "A generative adversarial approach for zero-shot learning from noisy texts", in "Proceedings of the IEEE conference on computer vision and pattern recognition", pp. 1004–1013 (2018).

APPENDIX A

DATASET REPOSITORY

- For Attribute-based, CUB, AWA and SUN
- For Wikipedia-based CUB and NAB

APPENDIX B

PERMISSION STATEMENTS FROM CO-AUTHORS

Permission for including co-authored material in this dissertation was obtained from co-authors, Prof. Sethuraman Panchanathan and Prof. Hemanth Venkateswara.