CONTINUAL ZERO-SHOT LEARNING THROUGH SEMANTICALLY GUIDED GENERATIVE RANDOM WALKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Learning new knowledge, not forgetting previous ones, and adapting it to future tasks occur simultaneously throughout a human's lifetime. However, this learning procedure is mostly studied individually in deep learning either from the perspective of lifetime learning without forgetting (continual learning) or adaptation to recognize unseen tasks (zero-shot learning, ZSL). Continual ZSL (CZSL), the desired and more natural learning setting, has been introduced in recent years, and mostly developed in transductive setting, which is unrealistic in practice. In this paper, we focus on inductive continual generalized zero-shot learning (CGZSL) by generative approach, where no unseen class information is provided during the training. The heart of the success of previous generative-based approaches is that learn quality representations from seen classes to improve the generative understanding of the unseen visual space. Motivated by this, we first introduce generalization bound tools and provide the first theoretical explanation for the benefits of generative modeling to ZSL and CZSL tasks. Second, we develop a pure Inductive Continual Generalized Zero-Shot Learner using our theoretical analysis to guide the improvement of the generation quality. The learner employs a novel semantically-guided Generative Random Walk (GRW) loss, where we encourage high transition probability, computed by random walk, from seen space to a realistic generative unseen space. We also demonstrate that our learner continually improves the unseen class representation quality, achieving state-of-the-art performance on AWA1, AWA2, CUB, and SUN datasets and surpassing existing CGZSL methods by around 3-7% on different datasets. Code is available here `https://anonymous.4open.science/r/cgzsl-76E7/main.py`

1 INTRODUCTION

Researchers have invested a lot of effort in designing human-like AI learners. Zero-shot learning (ZSL), as one of these endeavors, aims at identifying instances of unseen classes. This can be achieved by either the transductive methods that leverage semantic information to characterize unseen classes or the inductive approaches that do not require any knowledge of unseen classes. To develop a robust technique, recent ZSL works propose to evaluate models in the generalized setting where test samples are from both seen and unseen classes, noted as Generalized ZSL (GZSL) (Pourpanah et al., 2022).

Human zero-shot learning skill, on the other hand, expands over their lifetime, where the distributions of seen and unseen tasks temporally change. With the ever-growing knowledge over more seen tasks, people's ability to recognize unseen tasks and distinguish them from seen ones improves over time. In the meantime, the continual learning community also strives to design models with enhanced zero-shot future task transferability (Lin et al., 2021; Douillard et al., 2021). Towards bridging the gap between continual learning and zero-shot learning in a more realistic setting, Skorokhodov & Elhoseiny (2021) extended ZSL to continual ZSL (CZSL). In CZSL, the model is continually trained and evaluated by zero-shot learning metrics, where classes up until the current task are regarded as seen classes and those in the future are regarded as unseen classes. Since the unseen world changes in a dynamic and unexpected manner, it is unrealistic to use the knowledge about unseen classes in CZSL. However, most existing methods struggle to work well without semantic information in the CZSL setting.

Table 1: Showing Seen-Unseen AUC results of ZSL experiments on noisy text description based datasets **CUB** and **NAB**(Easy and Hard Splits)

Metric	Seen-Unseen AUC (%)			
Dataset	CUB		NAB	
Split-Mode	Easy	Hard	Easy	Hard
ZSLNS (Qiao et al., 2016)	14.7	4.4	9.3	2.3
SynC$_{fast}$ (Changpinyo et al., 2016)	13.1	4.0	2.7	3.5
ZSLPP (Elhoseiny et al., 2017)	30.4	6.1	12.6	3.5
FeatGen (**?**)	34.1	7.4	21.3	5.6
LsrGAN (*tr*) (Vyas et al., 2020)	39.5	12.1	23.2	6.4
+GRW	39.9$^{+0.4}$	13.3$^{+1.2}$	24.5$^{+1.3}$	6.7$^{+0.3}$
GAZSL (*in*) (Zhu et al., 2018)	35.4	8.7	20.4	5.8
+ CIZSL (Elhoseiny & Elfeki, 2019)	39.2	11.9	24.5	6.4
+ GRW	**40.7** $^{+5.3}$	**13.7**$^{+5.0}$	**25.8**$^{+5.4}$	**7.4** $^{+1.6}$

Table 2: Showing Seen Unseen harmonic results of ZSL experiments on attribute based datasets **AwA2**, **aPY** and **SUN**.

	Seen-Unseen H		
	AwA2	aPY	SUN
SYNC Changpinyo et al. (2016)	18.0	13.3	13.4
SAE (Kodirov et al., 2017)	2.2	0.9	11.8
DEM (Zhang et al., 2016)	25.1	19.4	25.6
FeatGen (**?**)	17.6	21.4	24.9
cycle-(U)WGAN (Felix et al., 2018)	19.2	23.6	24.4
LsrGAN (*tr*) (Vyas et al., 2020)	48.7*	31.5*	44.8
+ GRW	**49.2**$^{+0.5}$	**32.7**$^{+1.2}$	**46.1**$^{+1.3}$
GAZSL (*in*) (Zhu et al., 2018)	15.4	24.0	26.7
+ CIZSL (Elhoseiny & Elfeki, 2019)	24.6	25.7	27.8
+ GRW	39.0$^{+23.6}$	27.2$^{+3.2}$	27.9$^{+1.2}$

Generative models, especially GANs, have gained much progress in producing photorealistic images by learning high dimensional probability distributions. This promising ability has motivated various researchers to adapt GAN to ZSL to generate missing data of unseen classes (Guo & Viktor, 2004). The generative model can reduce model prediction bias towards seen data, contributing to competitive and strong performance (Li et al., 2019; Vyas et al., 2020; Narayan et al., 2020). In spite of all these empirically developed generative ZSL approaches, theoretical reasons for why zero-shot learning benefits from generative structures are rarely explored. However, there are firm foundations on the theoretical study of related topics, such as embedding-based zero-shot learning (Rostami et al., 2022), domain adaptation (Ben-David et al., 2010), and continual learning (Wang et al., 2022). Recent analysis on training the generative model (Chang et al., 2019) with synthetic data also provides a possible route towards the desired theoretical explanation. All of these considerations lead us to develop a generalization bound tool to understand the learning mechanism in generative-based CZSL.

In our analysis, we quantified the factors that it is critical for a CZSL learner to generate realistic visual generations of unseen classes to 1) reduce the distance between the generated and actual unseen visual space and 2) prevent the model from shifting so much that it becomes difficult to learn future tasks discriminatively. However, the lack of ground truth semantic descriptions/attributes of unseen classes makes it challenging to generate realistic samples of unseen classes. A similar problem has been tackled in novel style artworks generation, where GANs' training is augmented with a loss to encourage deviation from existing art style classes (Elgammal et al., 2017; Sbai et al., 2018; Hertzmann, 2018; Jha et al., 2021; Hertzmann, 2020). Inspired by the improved feature representation achieved by generative models in novel art generation and its connection to unseen samples in GZSL, we propose a purely inductive semantically guided **Generative Random Walk (GRW)** loss in CGZSL.

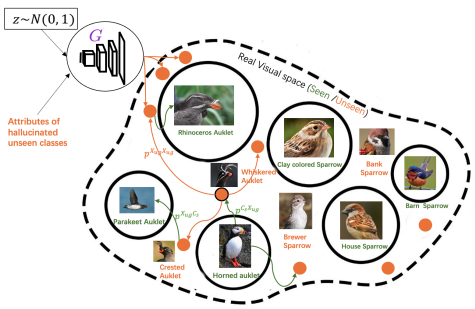

Figure 1: The generative random walk process starts from each seen class (in green) and go through generated examples of hallucinated unseen classes (in orange).

As illustrated in Figure 1, GRW start the transition from seen class and perform a random walk through generated examples of hallucinated unseen classes for R steps, detailed later (Section 5.2.2). GRW then encourages high transition probability to the realistic unseen space by deviating from the seen visual space and avoiding the less real space. Then, the landing representations are realistic but distinguishable from seen classes, and this quality enhances generative understanding of unseen classes[1].The design of hallucinated semantic descriptions of unseen classes instead of real ones is what makes GRW purely inductive. To validate this, exploratory experiments were conducted with GRW loss when integrated with existing generative GZSL methods in non-continual setup. As shown in Table 1 and Table 9, the method with GRW improves up to 5.4% Seen-Unseen AUC (Chao et al., 2016) results in the hardest textual description dataset, and even up to promisingly 23% in the attribute-based dataset. We will later show its effectiveness in a CZSL setting as part of our approach.

[1]The ability to produce generations of unseen classes that are distinguishable from each other and seen ones

Motivated by the promising empirical and theoretical generative approach in GZSL, we develop generalization bound tools to analyze the use of random walk in continual zero-shot learning in Section 4, formally describe our theoretical guided inductive continual generalized zero-shot learning method, ICGZSL, in Section 5, and show experimental results in Section 6. Our **contribution** includes 1) We give the first theoretical analysis, to our knowledge, of (continual) generalized zero-shot learning; 2) Guided by the analysis, we develop methods in pure inductive continual zero-shot learning setting; 3) Our model achieves SOTA results in standard continual zero-shot learning tasks.

2 Related Works

Generative Approach in Zero-shot Learning. Zero-Shot learning can be divided into embedding-based and generative-based methods. Embedding-based approaches project visual and semantic information into a shared embedding space, where classification is based on class projection and sample representation similarity. (Frome et al., 2013; Akata et al., 2015; Romera-Paredes & Torr, 2015; Liu et al., 2020). Generative-based approaches (Xian et al., 2018b; Vyas et al., 2020; Kuchibhotla et al., 2022) convert the zero-shot learning problem to a traditional supervised learning problem by using the generator to produce visual samples of unseen classes during training.

Inductive generalized zero-shot learning using generative approaches. There are varying degrees of accessibility to unseen information in zero-shot learning. Transductive methods use unlabeled samples and attributes of unseen classes during the training (Paul et al., 2019; Rahman et al., 2019). Semantically transductive methods only use attributes of unseen classes in training (Xian et al., 2018b; Wang et al., 2021). In the inductive setting, any usage of unseen information is not allowed (Zhu et al., 2018; Elhoseiny & Elfeki, 2019; Liu et al., 2019; Xian et al., 2019). This can result in a bias towards seen classes (Pourpanah et al., 2022). Generative methods produce unseen samples utilizing only seen class information during training to solve this. Liu et al. (2019) proposes to combine both GAN and VAE with an unconditional discriminator to generate features of unseen samples. Elhoseiny & Elfeki (2019) relate ZSL to human creativity (Martindale, 1990) to generate images that deviate from seen classes during training. Chandhok et al. (2021) used unlabeled samples from out-of-distribution data to gather knowledge about unseen data. In contrast, we investigate the relationship between the unseen generated samples and the seen samples, which leads to the GRW loss.

Continual Learning. The majority of continual learning works try to tackle the problem of catastrophic forgetting that the data representation has biased towards the most recent task in sequential learning. There are three primary groups of used techniques, i.e., regularization based methods (Li & Hoiem, 2017; Aljundi et al., 2018), structure-based methods (Rajasegaran et al., 2019; Ebrahimi et al., 2020), and replay based methods (Shin et al., 2017; Xiang et al., 2019). Recent research also explores forward transfer in continual learning, with the belief that as knowledge accumulates, higher next-task transferability measured by zero-shot assessment should be attained. Their evaluation space either includes the next task (Lin et al., 2021) or the whole class space (Douillard et al., 2021). However, compared to our setting, Lin et al. (2021) did not evaluate the model in a generalized manner, and (Douillard et al., 2021) only paid attention to the seen accuracy.

Continual Zero-shot learning. Chaudhry et al. (2019) introduced A-GEM for continual learning and applied the learner to deal with zero-shot tasks sequentially, constituting the initial work of CZSL. Skorokhodov & Elhoseiny (2021) proposed the inductive CZSL scenario for this work and found that a class-based normalization approach can improve continual zero-shot learning performance. Both Gautam et al. (2021) and Ghosh (2021a) explore the CZSL problem but use unseen class descriptions to train a classifier before inference. Recently Kuchibhotla et al. (2022) provided a generative adversarial approach with a cosine similarity-based classifier supporting dynamic addition of classes. Although their classifier does not need to be trained by unseen samples, they still use unseen class descriptions for the seen-unseen deviation, making it semantically transductive. This motivates us to explore the inductive method for the seen-unseen deviation and unseen realism.

3 Problem Setup and Notations

We formally describe our problem and notation following Chang et al. (2019); Skorokhodov & Elhoseiny (2021). A labelled dataset is defined as a tuple $\boldsymbol{D} = \{(\boldsymbol{x}, \boldsymbol{a}, y) | y = f(\boldsymbol{x}), (\boldsymbol{x}, \boldsymbol{a}, y) \sim \mathcal{D}\}$, where $\mathcal{D}$ represents the data distribution, composed of the tuple of data point of extracted image

features and its corresponding attribute $(\boldsymbol{x}, \boldsymbol{a}) \in \mathbb{R}^{d_x+d_a}$ ($\boldsymbol{x} \in \mathbb{R}^{d_x}, \boldsymbol{a} \in \mathbb{R}^{d_a}$), and a label y. Here d_x is the dimension of the visual feature space, and d_a is the dimension of the attribute space. Each distribution has a specific labeling function f. Our goal is to learn a model on top of $\boldsymbol{D}$ to estimate f.

In inductive generalized zero-shot learning, we hope our model learned on seen dataset $\boldsymbol{D}_s$ can be generalized to unseen distribution $\mathcal{D}_u$ without using any unseen class information. Moreover, we assume the seen class n and unseen class are disjoint, that is $\mathcal{D}_s \cap \mathcal{D}_u = \phi$. During the training time, the model $\hat{f}$ is trained on the seen dataset $\boldsymbol{D}_s$ as well as the synthesized dataset $\boldsymbol{D}_{ug}$. $\boldsymbol{D}_{ug}$ is generated by conditioning on unseen attribute $\boldsymbol{a}_{ug}$ and prior $\mathcal{Z} \sim \mathcal{N}(0, 1)$. The labelling function f_{ug} of the generated dataset is a look-up table of the generated features $\boldsymbol{x} \in \boldsymbol{X}_{ug}$ and the corresponding attribute condition $\boldsymbol{a}_{ug}$. During the inference time, the model is evaluated on test examples of both seen dataset $\boldsymbol{D}_s$ and unseen dataset $\boldsymbol{D}_u$. In continual zero-shot learning, we solve the zero-shot learning problem sequentially, and the unseen distribution converts dynamically into seen distribution. This procedure is illustrated in the bottom part of Figure 2

Notation. We use several notational conventions in this paper that are stated here. 1) We use the subscript $\cdot_s(\cdot_{sg}), \cdot_u(\cdot_{ug})$ to specify the variables in seen real (seen generated) space or unseen real (unseen generated) space; 2) We use variable with hat $\hat{\cdot}$ for the values and model empirically computed; 3) We use superscripted $\cdot^t$ or $\cdot^{1:t}$ to specify the variable is for task t or for task $1:t$. In practice, $\cdot^{1:t}$ refers to the current variable plus previous ones in the buffer. ; 4) $\boldsymbol{D}$ is for empirical sample set, and $\mathcal{D}$ is for distribution; 5) We use N_s, N_u for the number of seen or unseen class.

4 Theoretical Analysis

4.1 Essential Factors in Inductive Zero-Shot Learning

We analyze the major components in our approach that influence continual zero-shot learning performance by generalization bound. Given the whole training distribution, a learning algorithm can output a hypothesis h to estimate the ground truth labeling function f. Due to the limitation in the volume of training data, our learning algorithm outputs $\hat{h}$ instead to estimate $f_s \cup f_u$, which can be measured by risk (Kearns & Vazirani, 1994). We define the actual risk $\epsilon(h, f) = \mathbb{E}_{(\boldsymbol{x},\boldsymbol{a})\sim\mathcal{D}}[\mathbb{1}_{f(\boldsymbol{x})\neq h(\boldsymbol{x},\boldsymbol{a})}]$, and the empirical risk $\hat{\epsilon}(h, f) = \frac{1}{|\boldsymbol{D}|}\sum_{(\boldsymbol{x},\boldsymbol{a})\in\boldsymbol{D}}[\mathbb{1}_{f(\boldsymbol{x})\neq h(\boldsymbol{x},\boldsymbol{a})}]$.

We start our analysis by proposing a distance measure of the generated unseen distribution and real unseen distribution, as well as its empirical counterpart by

Definition 4.1 (Generative distance)**.** Given two feature distributions $\mathcal{D}_{ug}$ and $\mathcal{D}_u$, the ground truth labelling function f_{ug}, f_u, and the optimal hypothesis $h^* = \arg\min_{h\in H} \epsilon(h, f_{ug}) + \epsilon(h, f_u)$. The $h^*\Delta f$-distance between $\mathcal{D}_{ug}$ and $\mathcal{D}_u$ is defined as

$$d_{h^*}(\mathcal{D}_{ug}, \mathcal{D}_u) = |\mathbb{P}_{(\boldsymbol{x},\boldsymbol{a})\sim\mathcal{D}_{ug}}[f_{ug}(\boldsymbol{x}) \neq h^*(\boldsymbol{x}, \boldsymbol{a})] - \mathbb{P}_{(\boldsymbol{x},\boldsymbol{a})\sim\mathcal{D}_u}[f_u(\boldsymbol{x}) \neq h^*(\boldsymbol{x}, \boldsymbol{a})]| \ ,$$

and its empirical counterpart when we only have the limited empirical generated and real unseen samples is

$$\bar{d}_{GDB}(\boldsymbol{D}_{ug}, \boldsymbol{D}_u) = |\hat{\epsilon}(\hat{h}^*, f_u) - \hat{\epsilon}(\hat{h}^*, f_{ug})| \ ,$$

where $\hat{h}^* = \arg\min_{h\in H} \hat{\epsilon}_s(h, f_s) + \hat{\epsilon}_{ug}(h, f_{ug})$ is the optimal hypothesis of during the training.

Note that this proposed $\bar{d}_{GDB}$ enjoys the property of a pseudo-metric, and it represents the distribution distance computed by empirical samples depending on the hypothesis space, i.e. the type of model. And these two distances can be used in general problems irrespective to unseen generated and unseen real distribution.

Theorem 4.2 (Generalization bound of generative ZSL)**.** *Given the ZSL procedure described in section 3, with confidence* $1-\delta$ *the risk on the unseen dataset is bounded by*

$$\epsilon_u \leq \hat{\epsilon}(h, f_s) + \frac{1}{2}d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_s, \mathcal{D}_u) + \bar{\lambda} + \frac{1}{2}\bar{d}_{GDB}(\boldsymbol{D}_u, \boldsymbol{D}_{ug}) \ , \tag{1}$$

where $\hat{h}^* = \arg\min_{h\in H} \hat{\epsilon}(h, f_s) + \hat{\epsilon}(h, f_{ug})$, $\bar{\lambda} = \hat{\epsilon}(\hat{h}^*, f_s) + \hat{\epsilon}(\hat{h}^*, f_{ug})$.

The detailed derivation of this theorem is in Appendix A.1. Note that $d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_s, \mathcal{D}_u)$ is fixed when we are facing a specific problem, but $\bar{d}_{GDB}(\boldsymbol{D}_u, \boldsymbol{D}_{ug}), \bar{\lambda}$, and $\hat{\epsilon}_s(h, f_s)$ can be improved by our algorithm. We further extend the generalization bounds of $\epsilon(h, f_s)$ to the continual setting following the theorem proposed by Wang et al. (2022); see Appendix A.2.

4.2 REDUCING THE BOUND USING MARKOV CHAIN AND CONNECTION TO THE LEARNING ALGORITHM

In Equation 1, $\hat{\epsilon}(h, f_s)$ and $\bar{\lambda}$ can be reduced by adversarial training, which will be illustrated in Section 5.2.2. $\bar{d}_{GDB}(\boldsymbol{D}_{ug}, \boldsymbol{D}_u)$ can be reduced by decreasing the difference between $\boldsymbol{D}_u$ and $\boldsymbol{D}_{ug}$. Assuming we can establish $\boldsymbol{a}_{ug}$ as a compact support of $\boldsymbol{a}_u$, this can be further achieved by generating quality unseen generations to increase $\mathbb{P}[\boldsymbol{D}_u \subset \boldsymbol{D}_{ug}]$, where the probability is taken over all the possible generations. To quantify the probability value $\mathbb{P}[\boldsymbol{D}_u \subset \boldsymbol{D}_{ug}]$, we view the generations as nodes in a Markov chain, and define the transition probability between two states as how often one sample is classified as another. Then $\mathbb{P}[\boldsymbol{D}_u \subset \boldsymbol{D}_{ug}]$ can be bounded by the self transition probability by the generalization bound. When the self-transition probability is the same in two set of generations, we prefer the one with higher diversity quantified by DDP. Detailed explanations are illustrated in Appendix A.3. Here we give the informal statement.

Statement 4.3. *Finding unseen generated samples to "carefully" increase the determinant and the diagonal entries of the transition matrix of the above described Markov Chain can reduce $\bar{d}_{GDB}$.*

Connection of the analysis to the learning algorithm. The analysis gives guidelines for the development of the algorithm. However, some notions are intractable to compute in practice. We apply the following two adjustments to our algorithm. Firstly, we represent transition matrix among unseen classes (noted as $\boldsymbol{B} \in \mathbb{R}^{N_u \times N_u}$) in seen class space by a congruent transformation $\boldsymbol{A}\boldsymbol{B}\boldsymbol{A}^\top$, where $\boldsymbol{A} \in \mathbb{R}^{N_s \times N_u}$ is the transition probability matrix from seen to unseen. Secondly, generating compact support of unseen class attributes whose transition matrix is diagonal requires a huge number of generations. To reduce this number, we encourage the generated samples of hallucinated unseen classes to have "relatable deviation" to seen classes instead, where the transition matrix $\boldsymbol{B}$ may not be strictly diagonal but encouraged to be. And to still keep the diversity, i.e., low values of non-diagonal entries, we design the sampling strategy of hallucinated unseen classes to explore more unseen classes during training, as detailed later in Sec. 5.2.1.

To quantify the transition probability, we adapt the random walk framework (Häusser et al., 2017; Ayyad et al., 2021) originally used in the semi-supervised few-shot learning setting to the generative zero-shot learning setting with few changes. Compared to the semi-supervised version of the loss, we use generated examples instead of unlabeled samples and try to push the generated samples away from seen class centers. This is the opposite of a semi-supervised case where they use an attraction signal instead of a deviation signal (see Appendix B.6 for more details). Then the transitions going from the seen centers to unseen class generations for R steps and back to seen centers is $\boldsymbol{A}\boldsymbol{B}^R\boldsymbol{A}^\top \in \mathbb{R}^{N_s \times N_s}$. We encourage "relatable deviation" of unseen class generations from seen classes by having $\boldsymbol{A}\boldsymbol{B}^R\boldsymbol{A}^\top$ close to uniform, detailed later in Equation. 22. This makes the generations not attracted by any seen classes, yet by design, also transfer knowledge between seen classes to enable knowledge transfer.

5 GENERATIVE-BASED INDUCTIVE CZSL APPROACH

5.1 PRELIMINARY: GENERATIVE-BASED CONTINUAL ZERO-SHOT LEARNING

Our model contains a generator $G(\boldsymbol{a}, \boldsymbol{z}) \in \mathbb{R}^{d_x}$ and a discriminator $D(\boldsymbol{a}) \in \mathbb{R}^{d_x}$ (See Figure 2). The generator takes the concatenated semantic information (denoted by $\boldsymbol{a}$) and the prior (denoted by $\boldsymbol{z}$) sampled from a standard normal distribution $\mathcal{Z}$ as input and outputs visual features. Discriminator projects semantic information $\boldsymbol{a}$ into visual space. The conditional adversarial training can be illustrated by the discriminator loss and generator loss as:

$$\begin{aligned} \mathcal{L}_D &= -\mathcal{L}_{\text{real-fake}} + \lambda_{\text{cls}}\mathcal{L}_{\text{classification}} + \lambda_{\text{rd}}\mathcal{R}_D, \\ \mathcal{L}_G &= \mathcal{L}_{\text{real-fake}} + \lambda_{\text{cls}}\mathcal{L}_{\text{classification}} + \mathcal{L}_{\text{inductive}} + \lambda_{\text{rg}}\mathcal{R}_G \ . \end{aligned} \tag{2}$$

$\mathcal{L}_{\text{real-fake}}$ is the standard GAN loss taken on current task real and generated seen classes following Goodfellow et al. (2014), and $\mathcal{L}_{\text{classification}}$ is cosine similarity based entropy loss taken over the seen classes of all the tasks, shown in the following

$$\begin{aligned} \mathcal{L}_{\text{real-fake}} =& \mathbb{E}_{(\boldsymbol{x},\boldsymbol{a})\sim\boldsymbol{D}_s^t}\big[\log\langle \boldsymbol{x}, D(\boldsymbol{a})\rangle\big] - \mathbb{E}_{\boldsymbol{z}\sim\mathcal{Z},(\boldsymbol{a},\boldsymbol{x})\sim\boldsymbol{D}_s^t}\big[\log\langle G(z,\boldsymbol{a}), D(\boldsymbol{a})\rangle\big] \\ \mathcal{L}_{\text{classification}} =& \mathbb{E}_{(\boldsymbol{x},y)\sim\boldsymbol{D}_s^{1:t}}\big[L_e(\langle \boldsymbol{x}, D(\mathcal{A}_s^{1:t})\rangle, y)\big] + \mathbb{E}_{\boldsymbol{z}\sim\mathcal{Z}(\boldsymbol{x},\boldsymbol{a},y)\sim\boldsymbol{D}_s^{1:t}}\big[L_e(\langle G(\boldsymbol{z},\boldsymbol{a}), D(\mathcal{A}_s^{1:t})\rangle, y)\big] \end{aligned}$$

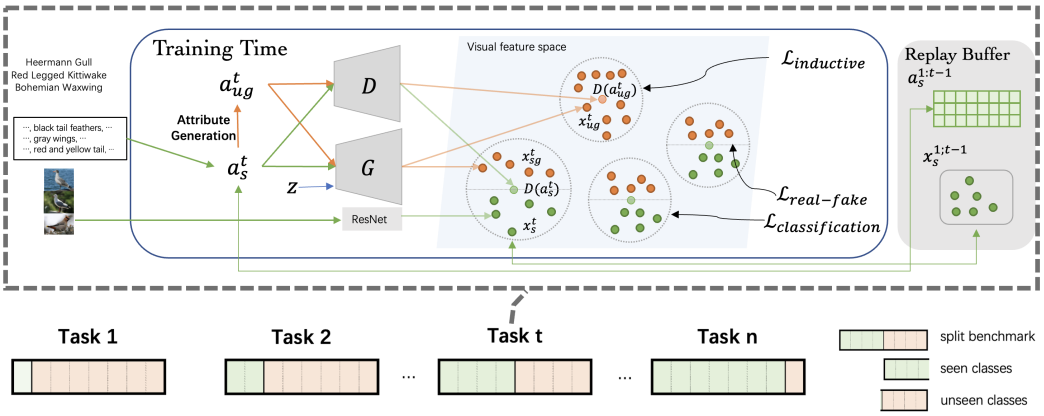

Figure 2: The bottom illustrates a continual zero-shot setting, where unseen classes dynamically become seen. The upper part displays our CZSL learner. It generates unseen $\boldsymbol{a}_{ug}^t$ at each time step t, based on the seen attributes $\boldsymbol{a}_s^t$. The discriminator embeds the attributes into the visual feature space, and the generator generates seen and unseen features $\boldsymbol{x}_{ug}^t, \boldsymbol{x}_s^t$ conditioning on the corresponding attributes. Inductive loss applied to the visual feature space encourages the unseen features to be real and characterized. Features of previous tasks are stored in the buffer of fixed size.

where $\langle \cdot, \cdot \rangle$ is the cosine similarity, $\mathcal{A}_s^{1:t}$ is the matrix of attributes of seen class up to the current task and L_e is the cross entropy loss. In practice, the $\boldsymbol{D}_s^{1:t}$ consists of current samples and previous samples in the buffer.

$\mathcal{L}_{\text{inductive}}$ is the main loss to improve inductive continual zero-shot learning, which will be described in detail in section 5.2. $\mathcal{R}_D, \mathcal{R}_G$ are regularization terms for discriminator and generator respectively that follow Kuchibhotla et al. (2022). $\lambda_{c,rd,i,rg}$ are hyperparameters. Following the standard GAN training procedure, the discriminator, and the generators are optimized alternatively. See Appendix B.1 and Algorithm 1 for more details about our baseline algorithm.

5.2 Inductive Unseen Attribute and Sample Generation

5.2.1 Attribute and Sample Generation of Unseen Classes

Interpolation-based method. With the assumption that the attributes are distributed uniformly in the attribute space, which can be compact-supported by the seen attribute, we use an interpolation-based method to estimate unseen attribute at every mini-batch, introduced by Elhoseiny & Elfeki (2019). Unseen attributes are generated by $\boldsymbol{a}_{ug} = \alpha \boldsymbol{a}_{s_1} + (1-\alpha)\boldsymbol{a}_{s_2}$, where $\alpha \sim \mathcal{U}(0.2, 0.8)$, $\boldsymbol{a}_{s_1}$ and $\boldsymbol{a}_{s_2}$ are two random seen attributes. The sample interval is chosen as $(0.2, 0.8)$ to avoid interpolated attributes being too close to seen attributes.

Dictionary-based method. With the assumption that attributes are sparsely distributed in the attribute space, we suggest learning a sparse attribute dictionary in $\mathbb{R}^{N_s^t \times d_a}$ space during training and randomly picking an attribute from it. The use of a learnable dictionary permits the attributes to change more freely in accordance with the loss function. The dictionary is randomly initialized by interpolation of seen attributes. This is more useful for classification at a finer level.

See Table 3 and appendix B.3 for the analysis and visualization of the above two assumptions. Conditioning on the generated attributes, the unseen samples can be generated by the Generator, $\boldsymbol{x}_{ug} = G(\boldsymbol{a}_{ug}, \boldsymbol{z})$, where $\boldsymbol{a}_{ug}$ is obtained by either of the two aforementioned methods. For each unseen attribute, we only generate one unseen sample to encourage the diversity of unseen attributes.

5.2.2 Improve Generation Quality by Inductive Loss

As there are no real examples of unseen classes for adversarial training, we need to construct additional learning signals to encourage the unseen generations to be realistic and characterized. As introduced in Section 4.2, we perform a random walk to compute the transition probability using $\boldsymbol{D}_s^t, \boldsymbol{D}_{sg}^t$ and $\boldsymbol{D}_{ug}^t$. The random walk starts from each generative seen class center $\boldsymbol{C} \in \mathbb{R}^{N_s^{1:t} \times d_x}$

computed by the mean of generated seen samples from the corresponding class attributes, where $N_s^{1:t}$ are the number of seen classes till step t. Then we take R steps of transitions within unseen samples $\boldsymbol{D}_{ug}$ with the final landing probability over seen classes so far:

$$P^{\boldsymbol{C}\boldsymbol{D}_{ug}\boldsymbol{C}}(R) = \sigma(\boldsymbol{A})\sigma(\boldsymbol{B})^R\sigma(\boldsymbol{A})^\top \in \mathbb{R}^{N_s^{1:t}\times N_s^{1:t}} \tag{3}$$

where $\sigma(\cdot)$ is the soft-max activation, and as mentioned in Section 4, $\boldsymbol{A} \in \mathbb{R}^{N_s^{1:t}\times N_u^t}$ is the similarity between seen centers and unseen samples, $\boldsymbol{B} \in \mathbb{R}^{N_s^{1:t}\times N_s^{1:t}}$ is similarities among unseen samples (see Appendix B.2 for the details). Here N_u^t is the number of unseen classes at step t. We hope the probability $P^{\boldsymbol{C}\boldsymbol{D}_{ug}\boldsymbol{C}}(R)$ to be uniformly distributed over all the seen classes, and the probability $P^{\boldsymbol{C}\to\boldsymbol{D}_{ug}} = \sigma(\boldsymbol{A})\sigma(\boldsymbol{B}) \in \mathbb{R}^{N_s^{1:t}\times N_u^t}$ to be uniformly distributed over all the generated examples to encourage as many generations to be visited in the random walk, and hence influence the learning. Hence, our *Generative Random Walk* (GRW) loss is defined by

$$L_{\text{GRW}} = \sum_{r=0}^{R} \gamma^r L_e(P^{\boldsymbol{C}\boldsymbol{D}_{ug}\boldsymbol{C}}(r), \boldsymbol{U}_1) + L_e(P^{\boldsymbol{C}\to\boldsymbol{D}_{ug}}, \boldsymbol{U}_2) \ , \tag{4}$$

where $L_e(\cdot,\cdot)$ is the cross-entropy loss, $\boldsymbol{U}_1 \in \mathbb{R}^{N_s^{1:t}\times N_s^{1:t}}$ and $\boldsymbol{U}_2 \in \mathbb{R}^{N_s^{1:t}\times N_u^t}$ are uniform matrices with values of $1/N_s^{1:t}$ and $1/N_u^t$ respectively, R is the random walk steps, and γ is exponential decay.

In addition, we empirically found that the GRW loss can also work as a regularizer to encourage the consistency of generated seen visual space as well, which we defined as

$$\mathcal{R}_{\text{GRW}} = \sum_{r=0}^{R} \gamma^r L_e(P^{\boldsymbol{C}\boldsymbol{D}_{sg}\boldsymbol{C}}(r), \boldsymbol{I}) + L_e(P^{\boldsymbol{C}\to\boldsymbol{D}_{sg}}, \boldsymbol{I}) \ , \tag{5}$$

where $\boldsymbol{I} \in \mathbb{R}^{N_s^{1:t}\times N_s^{1:t}}$ are identity matrix, and $\boldsymbol{D}_{sg}$ represent the matrix for generated seen samples. We numerically show that the random walk based penalty can reduce $\bar{d}_{GDB}$ (Def 4.1) by the relationship between $\bar{d}_{GDB}$ and L_{GRW}. Details are shown in Appendix B

We also adapt the loss proposed in (Elhoseiny & Elfeki, 2019) to directly prevent the generated unseen samples from being classified into seen classes, i.e.,

$$L_{\text{creativity}} = \mathbb{E}_{\boldsymbol{z}\sim\mathcal{Z},\boldsymbol{a}_{ug}\sim\boldsymbol{D}_{ug}} D_{\text{KL}}\Big(\big\langle G(\boldsymbol{z},\boldsymbol{a}_{ug}), D(\mathcal{A}_s^{1:t})\big\rangle \| \boldsymbol{u}\Big) \ , \tag{6}$$

where $D_{\text{KL}}(\cdot\|\cdot)$ is the KL divergence, $\mathcal{A}_s^{1:t} \in \mathbb{R}^{N_s^{1:t}\times d_a}$ is the matrix of attributes vectors of seen classes until task, $\boldsymbol{a}_{ug}$ is generated unseen attributes according to Section 5.2.1, $\big\langle G(\boldsymbol{z},\boldsymbol{a}_{ug}), D(\mathcal{A}_s^{1:t})\big\rangle \in \mathbb{R}^{N_s^{1:t}}$ are the logits over seen classes so far for a given $G(\boldsymbol{z},\boldsymbol{a}_{ug})$, $\boldsymbol{u}$ is the uniform distribution in $\mathbb{R}^{N_s^{1:t}}$, where $\boldsymbol{u}_{[i]} = 1/N_s^{1:t}$

Inductive loss Combining Equation 4,5 and 6 our final inductive loss is

$$\mathcal{L}_{\text{inductive}} = \lambda_{\text{c}} L_{\text{creativity}} + \lambda_{\text{i}} L_{\text{GRW}} + \lambda_{\text{i}} \mathcal{R}_{\text{GRW}} \tag{7}$$

5.3 Replay-based Knowledge Retention

As discussed in Section 2, there are mainly three knowledge retention methods for sequential tasks. Skorokhodov & Elhoseiny (2021) found the replay-based method is the most desirable continual learning tool. Some existing method (Ghosh, 2021a;b; Kuchibhotla et al., 2022) tend to use the generative replay method proposed by Gautam et al. (2021), where the correctly predicted seen generated features from the previous task are stored in buffers. Nonetheless, the buffer size increases significantly over task since a fixed number of samples for each class is stored. Additionally, if the model struggles to make accurate predictions for certain classes, samples from these classes are absent in the buffer. (see Appendix B.5)

We empirically find that the class balanced experience replay method (Prabhu et al., 2020) can be extremely helpful. At every task, we save the class attribute in $\mathcal{A}^{1:t}$, class center matrix $\boldsymbol{C}$, and modify the buffer with current features noted as $\boldsymbol{D}_s^{1:t}$ such that the buffer is balanced across all the seen classes. Note that we store real features in the buffer instead of the raw data, which is a fair replacement of the saved generated features but significantly improves the performance. We study the impact of using both real and generative replay in our experiments in Appendix D.5.

Table 3: Our proposed learner ICGZSL achieves SOTA results when comparing with recent inductive (*in*) method, and is even shows competitive results in mHA (D.2) with recent semantic transductive methods (*tr*)

Dataset	AWA1			AWA2			CUB			SUN		
Metric	mSA	mUA	mHA	mSA	mUA	mHA	mSA	mUA	mHA	mSA	mUA	mHA
DVGR (*tr*)	65.1	28.5	38.0	73.5	28.8	40.6	44.9	14.6	21.7	22.4	10.7	14.5
A-CGZSL (*tr*)	71.0	24.3	35.8	70.2	25.9	37.2	34.3	12.4	17.4	17.2	6.3	9.7
BD-CGZSL (*tr*)	62.9	29.9	39.0	68.1	33.9	42.9	19.8	17.2	17.8	27.5	15.9	20.0
CN-CZSL (*in*)	-	-	-	33.6	6.4	10.8	44.3	14.8	22.7	22.2	8.2	12.5
BD-CGZSL-in(in)	62.1	31.5	40.5	67.7	32.9	42.3	37.8	9.1	14.4	34.9	14.9	20.8
ours + interpolation	67.0	34.2	**43.4**	71.1	34.9	44.5	42.2	22.7	28.4	36.0	21.6	26.8
ours + dictionary	67.1	33.5	41.6	70.2	35.1	**44.6**	42.4	23.6	**28.8**	36.5	21.8	**27.1**

6 EXPERIMENT

6.1 CONTINUAL ZERO-SHOT LEARNING EXPERIMENTS

Data Stream and Benchmarks: We adopt the continual zero-shot learning proposed in Skorokhodov & Elhoseiny (2021). In this setting, a T-split dataset $D^{1:T}$ forms $T-1$ tasks. At time step t, the split $D^{1:t}$ is defined as seen set of tasks, and the split $D^{t+1:T}$ is an unseen set of tasks; see Figure 2. We conduct experiments on four widely used CGZSL benchmarks for a fair comparison: AWA1 (Lampert et al., 2009), AWA2 (Yu et al., 2020), Caltech UCSD Birds 200-2011 (CUB) (Wah et al., 2011), and SUN (Patterson & Hays, 2012). We follow Skorokhodov & Elhoseiny (2021); Kuchibhotla et al. (2022) for the class split for continual zero-shot learning setting; see Appendix 10 for the details.

Baselines, backbone, and training: We use the method in Kuchibhotla et al. (2022) as the main baseline and compare it with recent CGZSL methods in the setting we mentioned above, including transductive method DVGR (Ghosh, 2021b), A-CGZSL (Ghosh, 2021a), BD-CGZSL (Kuchibhotla et al., 2022), and inductive method CN-CZSL (Skorokhodov & Elhoseiny, 2021). 'BD-CGZSL-in' denotes our modified inductive version of Kuchibhotla et al. (2022) by naively getting rid of the use of unseen attribute. We use vanilla GAN's Generator and Discriminator, both are two-layer linear networks. Image features are extracted by ResNet-101 pre-trained on Imagenet-1k in advance. The attributes from Xian et al. (2018a) and extracted features are used as our model input. We use the replay buffer size of 5k. We run all experiments for 50 epochs and 64 batch sizes with the Adam optimizer. We use a learning rate of 0.005 and a weight decay of 0.00001. Results reported in Table 3 are based on one NVIDIA Tesla P100 GPU. We tuned our hyperparameters, random walk steps R, coefficient of inductive loss terms λ_i with a validation set (Chaudhry et al., 2019). See the appendix D.4 for the hyperparameters we use and ablations.

Metrics: Following (Skorokhodov & Elhoseiny, 2021), we use the mean seen accuracy mSA, mean unseen accuracy mUA and mean harmonic seen/unseen accuracy mHA to measure the continual zero-shot learning ability. We also use the backward transfer, adapted from the continual learning literature Chaudhry et al. (2019); Aljundi et al. (2018); see Appendix D and D.2 for more details.

Results Our accuracy results are shown in Table 3, and the task-wise mHA is shown in Figure 3. In coarse-grained datasets AWA1 and AWA2, our proposed learner achieves 43.4% and 44.6% in mHA, respectively, surpassing all the current inductive and transductive methods. In the fine-grained datasets and longer sequential tasks (CUB, SUN), our method achieves 28.8% and 27.1%, surpassing all the current CZSL methods. We observe that even though other methods have comparable mSA, they have far lower mUA than ours. We believe that our method achieves this improved knowledge transfer ability from seen visual space to unseen visual space through the proposed inductive learning signals (i.e., $\mathcal{L}_{\text{inductive}}$). Table 4 shows the forgetting of different continual zero-shot learner. Our model exhibits a good backward transfer capability, especially on longer task sequences where BWT is more needed. We have the highest BWT, 0.19, on CUB. On SUN, forgetting (negative BWT) appears on most other models, but our method can still retain knowledge from the past. The results imply that the analysis tools we created allow us to determine which aspects are crucial for zero-shot learning, and the design of tools for continual learning enhances our capacity to retain information.

6.2 ABLATION STUDY

Effect of random walk-based penalty. To better understand the effect of our novel random walk-based penalties L_{GRW} and $\mathcal{R}_{GRW}$, we conducted experiments with and without them; see

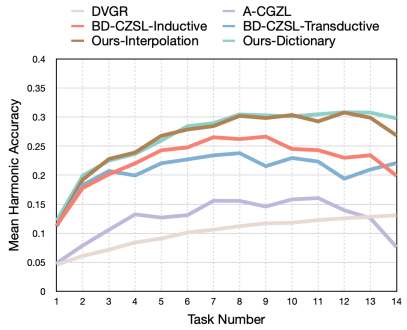

Figure 3: Average Harmonic accuracy up until each task on SUN dataset. Our method outperforms both transductive and inductive methods

Table 4: Backward-transfer of different CGZSL methods. Higher results mean less forgetting. Our proposed learner achieves competitive results in backward transfer in coarse-grained datasets and SOTA in fine-grained datasets

Dataset		AWA1	AWA2	CUB	SUN
DVGR	tr	0.09	0.10	-0.07	-0.20
A-CGZSL	tr	0.11	0.05	0.10	0.005
BD-CGZSL	tr	0.18	0.14	0.13	-0.02
CN-ZSL	in	-	-	-0.04	-0.02
BD-CGZSL-in	in	**0.18**	**0.15**	0.14	-0.03
ours + interpolation	in	0.12	0.10	**0.19**	**0.01**
ours + dictionary	in	0.11	0.11	**0.19**	**0.01**

Table 5: Effect of the random walk-based penalty with mH measure on CUB dataset.

	interpolation	dictionary
with $\mathcal{R}_{\text{GRW}}$ + L_{GRW}	**28.4**	**28.8**
- $L_{\text{creativity}}$	27.72	27.66
w/o $\mathcal{R}_{\text{GRW}}$, L_{GRW}	19.07	20.75
- $L_{\text{creativity}}$	14.43	14.43
with L_{GRW}	26.73	27.39

Table 6: Comparison of generative replay and real replay methods on CUB dataset. Dictionary-based attribute generation is used

	Buffer Size	Ours		BD-CGZSL
		mHA	BWT	mHA
generative	28.5k	0.14	21.06	17.76
real	10k	0.17	28.44	27.79
real	5k	0.19	28.8	26.55
real	2.5k	0.08	26.99	26.77

Table 5. The improvements are mainly from L_{GRW}, and $\mathcal{R}_{GRW}$ contributes around an additional 1%. $L_{\text{creativity}}$ is also part of the inductive loss. We show in Table 5 that removing $L_{\text{creativity}}$ while using our GRW losses have an insignificant effect on the performance; see Table 11 for more details.

Effect of experience replay. As mentioned in section 5.3, we found that real feature replay has advantages over generative feature replay. We make a comparison here on the CUB dataset between them. It shows that with around $1/10$ of the generative replay buffer size, the method with real replay can surpass that with generative replay in harmonic accuracy. With around $1/5$ of the generative replay buffer size, the method with real replay can have comparable BWT with the generative replay-based method. Moreover, the method with real replay is not sensitive to the buffer size. DVGR, A-CGZSL, and BD-CGZSL tend to use generative replay, and only CN-CGZSL uses real replay. We add the last column here in the Table 6 to show that our proposed replay method is also helpful for mHA of other methods.

7 Conclusion and Discussion

We studied continual zero-shot learning in this paper, where we believe inductive limitation needs more focus and exploration toward more realistic learning systems. We started our exploration by developing the first framework for the theoretical analysis of generative zero-shot learning. It helps us distinguish the influential factors, $\bar{d}_{GDB}$ when the unseen information is not accessible during training in continual zero-shot learning. We also proposed a continual zero-shot learner, ICGZSL, to reduce the $\bar{d}_{GDB}$ bound. To empirically evaluate our learner, we conducted experiments on four popular continual zero-shot learning benchmarks, AWA1, AWA2, CUB, and SUN. We increased around 3% harmonic accuracy in the small dataset and around 7% in the more extensive dataset compared to the previous inductive and transductive methods. We demonstrated that unseen semantic information is not essential with well-analyzed seen distribution and method.

Although $\bar{d}_{GDB}$ is satisfactory for the numerical analysis of a method, a more strict version of the relationship between d^{h^*} and $\bar{d}_{GDB}$ can be developed, and the multi-class classification condition should also be considered. Moreover, the distance can be written in those distance measure that has a relationship with GAN performance, such as the Wasserstein distance. Meanwhile, similar to most continual learning learners, there is still a performance gap to bridge between sequential tasks (Table 3) and non-sequential tasks (Table 9).

REFERENCES

Yaser S Abu-Mostafa, Malik Magdon-Ismail, and Hsuan-Tien Lin. *Learning from data*, volume 4. AMLBook New York, 2012.

Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(7):1425–1438, 2016.

Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory Aware Synapses: Learning what (not) to forget. In *European Conference on Computer Vision*, 2018.

Ahmed Ayyad, Yuchen Li, Nassir Navab, Shadi Albarqouni, and Mohamed Elhoseiny. Semi-supervised few-shot learning with prototypical random walks. In *AAAI Workshop on Meta-Learning and MetaDL Challenge*, pp. 45–57, 2021.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1-2):151–175, 2010.

Shivam Chandhok, Sanath Narayan, Hisham Cholakkal, Rao Muhammad Anwer, Vineeth N Balasubramanian, Fahad Shahbaz Khan, and Ling Shao. Structured latent embeddings for recognizing unseen classes in unseen domains. In *British Machine Vision Conference*, 2021.

Fu-Chieh Chang, Hao-Jen Wang, Chun-Nan Chou, and Edward Y. Chang. G2R Bound: A Generalization Bound for Supervised Learning from GAN-Synthetic Data. *arXiv:1905.12313*, 2019.

Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Synthesized classifiers for zero-shot learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5327–5336, 2016.

Wei-Lun Chao, Soravit Changpinyo, Boqing Gong, and Fei 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, 2016.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient Lifelong Learning with A-GEM. In *International Conference on Learning Representations*, 2019.

Arthur Douillard, Eduardo Valle, Charles Ollion, Thomas Robert, and Matthieu Cord. Insights from the future for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3482–3491, 2021.

Sayna Ebrahimi, Franziska Meier, Roberto Calandra, Trevor Darrell, and Marcus Rohrbach. Adversarial continual learning. In *European Conference on Computer Vision*, pp. 386–402, 2020.

Mohamed Elfeki, Camille Couprie, Morgane Riviere, and Mohamed Elhoseiny. Gdpp: Learning diverse generations using determinantal point processes. In *International conference on machine learning*, pp. 1774–1783, 2019.

Ahmed Elgammal, Bingchen Liu, Mohamed Elhoseiny, and Marian Mazzone. Can: Creative adversarial networks, generating "art" by learning about styles and deviating from style norms. In *International Conference on Computational Creativity*, 2017.

Mohamed Elhoseiny and Mohamed Elfeki. Creativity inspired zero-shot learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5784–5793, 2019.

Mohamed Elhoseiny, Yizhe Zhu, Han Zhang, and Ahmed Elgammal. Link the head to the "beak": Zero shot learning from noisy text description at part precision. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

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

Rafael Felix, Vijay BG Kumar, Ian Reid, and Gustavo Carneiro. Multi-modal cycle-consistent generalized zero-shot learning. In *European Conference on Computer Vision*, pp. 21–37, 2018.

Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *Advances in Neural Information Processing Systems*, volume 26, 2013.

Chandan Gautam, Sethupathy Parameswaran, Ashish Mishra, and Suresh Sundaram. Generative Replay-based Continual Zero-Shot Learning. *arXiv:2101.08894*, 2021.

Subhankar Ghosh. Adversarial training of variational auto-encoders for continual zero-shot learning. In *International Joint Conference on Neural Networks*, 2021a.

Subhankar Ghosh. Dynamic VAEs with Generative Replay for Continual Zero-shot Learning. In *CVPR workshop*, 2021b.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, volume 27, 2014.

Hongyu Guo and Herna L. Viktor. Learning from imbalanced data sets with boosting and data generation: the databoost-im approach. *ACM Sigkdd Explorations Newsletter*, 6(1):30–39, 2004.

Aaron Hertzmann. Can computers create art? In *Arts*, volume 7, pp. 18, 2018.

Aaron Hertzmann. Visual indeterminacy in gan art. *Leonardo*, 53(4):424–428, 2020.

Philip Häusser, Alexander Mordvintsev, and Daniel Cremers. Learning by association — a versatile semi-supervised training method for neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 626–635, 2017.

Divyansh Jha, Hanna H. Chang, and Mohamed Elhoseiny. Wölfflin's affective generative analysis of visual art. *The International Conference on Computational Creativity*, 2021.

Byungkon Kang. Fast determinantal point process sampling with application to clustering. *Advances in Neural Information Processing Systems*, 26, 2013.

Michael J Kearns and Umesh Vazirani. *An introduction to computational learning theory*. MIT press, 1994.

Elyor Kodirov, Tao Xiang, and Shaogang Gong. Semantic autoencoder for zero-shot learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3174–3183, 2017.

Hari Chandana Kuchibhotla, Sumitra S Malagi, Shivam Chandhok, and Vineeth N Balasubramanian. Unseen classes at a later time? no problem. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

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

Jingjing Li, Mengmeng Jing, Ke Lu, Zhengming Ding, Lei Zhu, and Zi Huang. Leveraging the Invariant Side of Generative Zero-Shot Learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7394–7403, 2019.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2017.

Zhiqiu Lin, Jia Shi, Deepak Pathak, and Deva Ramanan. The clear benchmark: Continual learning on real-world imagery. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

Shaoteng Liu, Jingjing Chen, Liangming Pan, Chong-Wah Ngo, Tat-Seng Chua, and Yu-Gang Jiang. Hyperbolic visual embedding learning for zero-shot recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9273–9281, 2020.

Yang Liu, Jishun Guo, Deng Cai, and Xiaofei He. Attribute attention for semantic disambiguation in zero-shot learning. In *IEEE/CVF International Conference on Computer Vision*, 2019.

Colin Martindale. *The clockwork muse: The predictability of artistic change.* Basic Books, 1990.

Sanath Narayan, Akshita Gupta, Fahad Shahbaz Khan, Cees GM Snoek, and Ling Shao. Latent embedding feedback and discriminative features for zero-shot classification. In *European Conference on Computer Vision*, pp. 479–495, 2020.

Genevieve Patterson and James Hays. SUN attribute database: Discovering, annotating, and recognizing scene attributes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2751–2758, 2012.

Akanksha Paul, Narayanan C Krishnan, and Prateek Munjal. Semantically aligned bias reducing zero shot learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7056–7065, 2019.

Farhad Pourpanah, Moloud Abdar, Yuxuan Luo, Xinlei Zhou, Ran Wang, Chee Lim, and Xi-Zhao Wang. A review of generalized zero-shot learning methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP, 07 2022.

Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. Gdumb: A simple approach that questions our progress in continual learning. In *European conference on computer vision*, pp. 524–540, 2020.

Ruizhi Qiao, Lingqiao Liu, Chunhua Shen, and Anton van den Hengel. Less is more: Zero-shot learning from online textual documents with noise suppression. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Shafin Rahman, Salman Khan, and Nick Barnes. Transductive learning for zero-shot object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6082–6091, 2019.

Jathushan Rajasegaran, Munawar Hayat, Salman Khan, Fahad Shahbaz Khan, and Ling Shao. Random path selection for incremental learning. *Advances in Neural Information Processing Systems*, 2019.

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

Mohammad Rostami, Soheil Kolouri, Zak Murez, Yuri Owekcho, Eric Eaton, and Kuyngnam Kim. Zero-Shot Image Classification Using Coupled Dictionary Embedding. *Machine Learning with Applications*, 8:100278, 2022.

Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

Othman Sbai, Mohamed Elhoseiny, Antoine Bordes, Yann LeCun, and Camille Couprie. Design: Design inspiration from generative networks. In *ECCV workshop*, 2018.

Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in Neural Information Processing Systems*, 30, 2017.

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

Grant Van Horn, Steve Branson, Ryan Farrell, Scott Haber, Jessie Barry, Panos Ipeirotis, Pietro Perona, and Serge Belongie. Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2015.

Twan van Laarhoven and Elena Marchiori. Unsupervised domain adaptation with random walks on target labelings. *arXiv:1706.05335*, 2017.

Maunil Vyas, Hemanth Venkateswara, and Sethuraman Panchanathan. Leveraging seen and unseen semantic relationships for generative zero-shot learning. In *European Conference on Computer Vision*, pp. 70–86, 2020.

C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

Boyu Wang, Jorge Mendez, Changjian Shui, Fan Zhou, Di Wu, Christian Gagné, and Eric Eaton. Gap Minimization for Knowledge Sharing and Transfer. *arXiv:2201.11231*, 2022.

Wenlin Wang, Hongteng Xu, Guoyin Wang, Wenqi Wang, and Lawrence Carin. Zero-Shot Recognition via Optimal Transport. In *IEEE Winter Conference on Applications of Computer Vision*, pp. 3470–3480, 2021.

Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. Latent embeddings for zero-shot classification. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 69–77, 2016.

Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018a.

Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature Generating Networks for Zero-Shot Learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5542–5551, 2018b.

Yongqin Xian, Saurabh Sharma, Bernt Schiele, and Zeynep Akata. F-vaegan-d2: A feature generating framework for any-shot learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

Ye Xiang, Ying Fu, Pan Ji, and Hua Huang. Incremental learning using conditional adversarial networks. In *IEEE/CVF International Conference on Computer Vision*, pp. 6619–6628, 2019.

Kai Yi and Mohamed Elhoseiny. Domain-Aware Continual Zero-Shot Learning. *arXiv:2112.12989*, 2021.

Lu Yu, Bartlomiej Twardowski, Xialei Liu, Luis Herranz, Kai Wang, Yongmei Cheng, Shangling Jui, and Joost van de Weijer. Semantic Drift Compensation for Class-Incremental Learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6980–6989, 2020.

Li Zhang, Tao Xiang, and Shaogang Gong. Learning a deep embedding model for zero-shot learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Yizhe Zhu, Mohamed Elhoseiny, Bingchen Liu, Xi Peng, and Ahmed Elgammal. A generative adversarial approach for zero-shot learning from noisy texts. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

A DERIVATION OF THEOREMS IN SECTION 4

A.1 THEOREM 3.2

Definition A.1 ($\mathcal{H}\Delta\mathcal{H}$-distance in Ben-David et al. (2010))**.** Given two feature distributions $\mathcal{D}_g$ and $\mathcal{D}_r$, and the hypothesis class $\mathcal{H}$, the $\mathcal{H}\Delta\mathcal{H}$-distance between $\mathcal{D}_g$ and $\mathcal{D}_r$ is defined as

$$d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_g, \mathcal{D}_r) = 2 \sup_{h,h' \in \mathcal{H}} |\mathbb{P}_{\boldsymbol{x}\sim\mathcal{D}_g}[h(\boldsymbol{x}) \neq h'(\boldsymbol{x})] - \mathbb{P}_{\boldsymbol{x}\sim\mathcal{D}_r} h(\boldsymbol{x}) \neq h'(\boldsymbol{x})| \ .$$

Note that the following inequality related to $d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_g, \mathcal{D}_r)$ holds for any h and h^*

$$\begin{aligned} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_g, \mathcal{D}_r) &= 2 \sup_{h,h' \in \mathcal{H}} |\mathbb{P}_{\boldsymbol{x}\sim\mathcal{D}_g}[h(\boldsymbol{x}) \neq h'(\boldsymbol{x})] - \mathbb{P}_{\boldsymbol{x}\sim\mathcal{D}_r} h(\boldsymbol{x}) \neq h'(\boldsymbol{x})| \\ &\geq 2|\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_g}[\mathbb{1}_{h(\boldsymbol{x})\neq h^*(\boldsymbol{x})}] - \mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}_r}[\mathbb{1}_{h(\boldsymbol{x})\neq h^*(\boldsymbol{x})}]| \\ &= 2|\epsilon_g(h, h^*) - \epsilon_r(h, h^*)| \ . \end{aligned} \tag{8}$$

Notation Statement in the Appendix To involve risk between hypotheses and between hypothesis and ground truth models, we use $\epsilon_{\cdot}(h, f)$ or $\epsilon_{\cdot}(h, h^*)$ to specify which space the risk is computed on.

Lemma A.2 (Abu-Mostafa et al. (2012))**.** *For a fixed hypothesis, the actual risk can be estimated from the empirical error with probability* $1 - \delta$

$$\epsilon(h, f) \leq \hat{\epsilon}(h, f) + \sqrt{\frac{1}{2m} \log \frac{2}{\delta}} \ , \tag{9}$$

where $\epsilon(h, f)$ *is the actual risk,* $\hat{\epsilon}(h, f)$ *is the empirical risk, and* m *is the number of testing samples.*

Proposition A.3 (Bound $d_{h^*}(\mathcal{D}_u, \mathcal{D}_{ug})$ by $\bar{d}_{GDB}(\boldsymbol{D}_u, \boldsymbol{D}_{ug})$)**.** *The distribution distance* $d_{h^*}(\mathcal{D}_u, \mathcal{D}_{ug})$ *can be bounded by it empirical counterpart by*

$$d_{h^*}(\mathcal{D}_u, \mathcal{D}_{ug}) \leq \bar{d}_{GDB}(\boldsymbol{D}_u, \boldsymbol{D}_{ug}) + C(\frac{1}{m}, \frac{1}{\delta}) \ , \tag{10}$$

where $C(\frac{1}{m}, \frac{1}{\delta})$ *is a constant term depending on the training sample size* m *and confidence* $1 - \delta$*. Here* $\mathcal{D}_{\cdot}$ *represent the distribution, and* $\boldsymbol{D}_{\cdot}$ *represents the dataset sampled from the corresponding distribution.*

Proof. Similar to Equation 8, we can write our generative distance as

$$d_{h^*}(\mathcal{D}_u, \mathcal{D}_{ug}) = 2|\epsilon_{ug}(h^*, f) - \epsilon_u(h^*, f)| \ . \tag{11}$$

Combining Lemma A.2, we have

$$\begin{aligned} \frac{1}{2} d_{h^*}(\mathcal{D}_u, \mathcal{D}_{ug}) &= |\epsilon_{ug}(h^*, f) - \epsilon_u(h^*, f)| \\ &\leq |\hat{\epsilon}_{ug}(h^*, f) - \hat{\epsilon}_u(h^*, f)| + |(\hat{\epsilon}_{ug}(h^*, f) + \hat{\epsilon}_u(h^*, f)) - (\epsilon_{ug}(h^*, f) + \epsilon_u(h^*, f))| \\ &\lesssim \frac{1}{2} \bar{d}_{GDB}(\boldsymbol{D}_u, \boldsymbol{D}_{ug}) + C(\frac{1}{m}, \frac{1}{\delta}) \ , \end{aligned} \tag{12}$$

where $h^* = \arg\min_{h' \in H} \epsilon_s(h', f_s) + \epsilon_{ug}(h', f_{ug})$, and $\hat{h}^* = \arg\min_{h' \in H} \hat{\epsilon}_s(h', f_s) + \hat{\epsilon}_{ug}(h', f_{ug})$. Following the discussion of Chang et al. (2019), we assume the optimal hypothesis $\hat{h}^*$ we can achieve is very close to the global minimum when the training sample is large, then we can estimate h^* in Equation 12 by $\hat{h}^*$. $C(\frac{1}{m}, \frac{1}{\delta})$ is obtained from Lemma A.2 □

Proof of theorem 3.2 Given the zero-shot learning procedure described in section 3.1, a model is trained on $\boldsymbol{D}_s$ and generate data $\boldsymbol{D}_{ug}$ by hallucinated unseen attribute $\hat{a}_{ug}$, then its risk on the unseen dataset is bounded by

$$\epsilon_u(h, f_u) \leq \hat{\epsilon}_s(h, f_s) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_s, \mathcal{D}_u) + \bar{\lambda} + \frac{1}{2} \bar{d}_{GDB}(\boldsymbol{D}_u, \boldsymbol{D}_{ug}) \ , \tag{13}$$

where $\hat{h}^* = \arg\min_{h' \in H} \hat{\epsilon}_s(h', f_s) + \hat{\epsilon}_{ug}(h', f_{ug})$, $\bar{\lambda} = \hat{\epsilon}_s(\hat{h}^*, f_s) + \hat{\epsilon}_{ug}(\hat{h}^*, f_{ug})$.

Proof. Let $h^* = \arg\min_{h' \in H} \epsilon_s(h', f_s) + \epsilon_{ug}(h', f_{ug})$, and $\lambda = \epsilon_s(h^*, f_s) + \epsilon_{ug}(h^*, f_{ug})$, It follows that

$$\begin{aligned}
\epsilon_u(h, f_u) &= \epsilon_s(h, f_s) + \epsilon_u(h, h^*) - \epsilon_s(h, h^*) + \epsilon_{ug}(h^*, f_{ug}) + \epsilon_s(h^*, f_s) - \epsilon_{ug}(h^*, f) + \epsilon_u(h^*, f) \\
&\quad - \epsilon_s(h, f_s) - \epsilon_u(h, h^*) + \epsilon_s(h, h^*) - \epsilon_s(h^*, f_s) - \epsilon_u(h^*, f) + \epsilon_u(h, f_u) \\
&\leq \epsilon_s(h, f_s) + |\epsilon_u(h, h^*) - \epsilon_s(h, h^*)| + |\epsilon_{ug}(h^*, f_{ug}) + \epsilon_s(h^*, f_s)| + |\epsilon_{ug}(h^*, f) - \epsilon_u(h^*, f)| \\
&\quad - \epsilon_s(h, f_s) + \epsilon_s(h, h^*) - \epsilon_s(h^*, f_s) - \epsilon_u(h, h^*) + \epsilon_u(h, f_u) - \epsilon_u(h^*, f) \\
&\leq \epsilon_s(h, f_s) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_s, \mathcal{D}_u) + \lambda + d_{h^*}(\mathcal{D}_{ug}, \mathcal{D}_u) \\
&\quad - \epsilon_s(h, f_s) + \epsilon_s(h, h^*) - \epsilon_s(h^*, f_s) - \epsilon_u(h, h^*) + \epsilon_u(h, f_u) - \epsilon_u(h^*, f) \ ,
\end{aligned} \tag{14}$$

Note that for any distribution

$$\begin{aligned}
|\epsilon_{\mathcal{D}}(h, f_{\mathcal{D}}) - \epsilon_{\mathcal{D}}(h, h^*)| &= |\mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}}[\mathbb{1}_{h \neq f_{\mathcal{D}}}] - \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}}[\mathbb{1}_{h \neq h^*}]| \\
&= |\mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}}[\mathbb{1}_{h \neq f_{\mathcal{D}}} - \mathbb{1}_{h \neq h^*}]| \\
&\leq \mathbb{E}_{\boldsymbol{x} \sim \mathcal{D}}[\mathbb{1}_{h^* \neq f_{\mathcal{D}}}] = \epsilon_{\mathcal{D}}(h^*, f_{\mathcal{D}}) \ ,
\end{aligned} \tag{15}$$

where the inequality holds by the triangle inequality of the characteristic function, i.e. $\mathbb{1}[a \neq b] \geq \mathbb{1}[a \neq c] - \mathbb{1}[b \neq c]$ for $\forall a, b, c \in \mathbb{R}$. Equation (15) shows that the fourth line in Equation (14) is less than or equal to zero.

Combining Equation 15, the Equation 14 can be written as

$$\epsilon_u(h, f_u) \leq \epsilon_s(h, f_s) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_s, \mathcal{D}_u) + \lambda + d_{h^*}(\mathcal{D}_{ug}, \mathcal{D}_u) \ , \tag{16}$$

However, Equation (16) involves unknown risk and unsolvable distribution. We combine the expected risk and the actual observed risk by Lemma A.2. Let $\hat{h}^* = \arg\min_{h' \in H} \hat{\epsilon}_s(h', f_s) + \hat{\epsilon}_{ug}(h', f_{ug})$ be the optimal hypothesis on the training set, and $\bar{\lambda} = \hat{\epsilon}_s(\hat{h}^*, f_s) + \hat{\epsilon}_{ug}(\hat{h}^*, f_{ug})$, we have $\lambda \leq \hat{\lambda}$. Together with Lemma A.2 and Proposition A.3, we have

$$\epsilon_u(h, f_u) \leq \hat{\epsilon}_s(h, f_s) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}_s, \mathcal{D}_u) + \bar{\lambda} + \frac{1}{2} \bar{d}_{GDB}(\boldsymbol{D}_u, \boldsymbol{D}_{ug}) \ . \tag{17}$$

□

A.2 Theorem 3.2 in Continual Learning Setting

We use real instant replay to improve the model's continual learning performance. In other words, our classification loss function at task t can be written as

$$\mathcal{L}_{\text{classification}} = \gamma^{<t} \mathcal{L}^{<t}_{\text{classification}} + \gamma^t \mathcal{L}^t_{\text{classification}} \ ,$$

where $\gamma^{<t}$ is the portion of replayed data, $\mathcal{L}^{<t}_{\text{classification}}$ is the classification loss on the replayed data, and γ^t is the portion of current data, $\mathcal{L}^t_{\text{classification}}$ is the classification loss on the current data. We have $\gamma^{<t} + \gamma^t = 1$.

Definition A.4 ($\mathcal{Y}$-Discrepancy)**.** Let $\mathcal{H}$ be a hypothesis class mapping $\mathcal{X}$ to $\mathcal{Y}$, and let $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ define a loss function over $\mathcal{Y}$. The $\mathcal{Y}$-Discrepancy between two distributions $\mathcal{X}_g$ and $\mathcal{X}_r$ is defined as

$$\text{dist}_{\mathcal{Y}}(\mathcal{X}_g, \mathcal{X}_r) = \sup_{h \in \mathcal{H}} |\mathcal{L}_{\mathcal{X}_g}(h) - \mathcal{L}_{\mathcal{X}_r}(h)| \tag{18}$$

Lemma A.5 (Theorem 6 in Wang et al. (2022))**.** *Let h^* be the optimal solution of the problem we described above. Assume the loss function is ρ-Lipschitz continuous For any $\delta \in (0, 1)$, with probability at least $1 - \delta$, we have*

$$\mathcal{L}^t_s(h^*) \leq \mathcal{L}^{<t}_{\textit{classification}}(h^*) + \gamma^{<t} \textit{dist}_{\mathcal{Y}}(\mathcal{D}^{<t}_s, \mathcal{D}^t_s) + \beta + (\Delta + \beta + \frac{B}{N}) \sqrt{\frac{N \log \frac{1}{\delta}}{2}} \ ,$$

where $\mathcal{L}^t_s(h^*)$ is the expected loss of seen classes at task t, $\mathcal{D}^t_s$ is the seen class distribution of task t, $N = N^{<t} + N^t$ with $N^{<t}$ the number of replayed data, N^t current seen data, B is the upper

bound of the loss function, $\beta = \max\{\beta_1^{<t}, \ldots, \beta_{N^{<t}}^{<t}, \beta_1^t, \ldots, \beta_{N^t}^t\}$, $\Delta = \sum \beta_*^*$, *with the stability coefficients upper bounded by*

$$\beta_i^* \leq \frac{\rho^2 R^2}{N\lambda} \ ,$$

Kindly refer to Wang et al. (2022) for more details.

Instead of connecting the expected version of theorem 3.2 (Equation 16) to its corresponding empirical counterpart, $\epsilon_s(h, f_s)$ also has relationship with the seen loss of previous tasks as implied by Lemma A.5. This reveals that the knowledge from currently seen classes should be transferred to the test time and next training time.

A.3 Explanation of Statement 3.3

Let $\boldsymbol{D}_{ug} \sim \mathcal{D}_{ug}$ be the generated unseen set we are training on, where $\mathcal{D}_{ug}$ is the empirical distribution of all possible generations. In unsupervised domain adaptation, van Laarhoven & Marchiori (2017) uses random walk to select label set for the samples who has small generalization error. Proposition 3.2 of van Laarhoven & Marchiori (2017) demonstrates that the self transition probability of a Markov chain represents an upper bound on the margin linear classifier's generalization error. This concept is adapted to connect our GDB bound connected to the Markov Chain in below. In our sample generation procedure, we generate only one sample from each class. Our discussion of this section will be based on this. We have $\bar{d}_{GDB}(\boldsymbol{D}_u, \boldsymbol{D}_{ug}) \propto -\sum_{i \in I_u} \mathbb{P}(\boldsymbol{a}_{u[i]} \in \boldsymbol{D}_{ug})$, where the probability is taken over $\mathcal{D}_{ug}$, and I_u is the index set of unseen real attributes. This is because the difference of the risk will be reduced if the generations contain as much points close to ground truth unseen ones as possible. Consider the Markov chain with single step transition probabilities p_{ij} of jumping from node i to node j. Each node represents a generated sample. Let

$$p_{ij} = \mathbb{P}[h(\boldsymbol{x}_i) = y_j] \ , \tag{19}$$

where h is the hypothesis trained on $\boldsymbol{D}_{ug}$, and the h output predictions on the current generation's classification space depending on the quality of h, and the probability is taken over $\mathcal{D}_{ug}$. We assume the training achieves error ϵ, then $h(\boldsymbol{x}_i) = y_i$ with probability $(1-\delta)$ if the training set contains class with attribute $\boldsymbol{a}_i$. It is not hard to prove that $\mathbb{P}(\boldsymbol{a}_{u[i]} \in \boldsymbol{D}_{ug}) \geq p_{ii}(1-\delta)(1-\epsilon)$ by the generalization bound, since if $\boldsymbol{a}_{u[i]} \notin \boldsymbol{D}_{ug}$, $y_{u[i]}$ is not in the current generation's classification space. It follows that

$$\bar{d}_{GDB}(\boldsymbol{D}_u, \boldsymbol{D}_{ug}) \propto -\sum_i \mathbb{P}(\boldsymbol{a}_{u[i]} \subseteq \boldsymbol{D}_{ug}) \leq -\sum_i p_{ii}(1-\delta)(1-\epsilon)$$

Then we can release the bound $\bar{d}_{GDB}(\boldsymbol{D}_u, \boldsymbol{D}_{ug})$ by increasing $\sum_i p_{ii}$. Note that $\mathbb{P}(\boldsymbol{a}_{u[i]} \in \boldsymbol{D}_{ug})$ can be replaced by $\mathbb{P}(\min_{\boldsymbol{a}_{ug[j]} \in \boldsymbol{D}_{ug}} |\boldsymbol{a}_{u[i]} - \boldsymbol{a}_{ug[j]}| < \varepsilon)$ with the robustness assumption of the model.

When two generations have the same $\sum_i p_{ii}$, we prefer the one having higher diversity. The diversity of the generated set $\boldsymbol{D}_{ug}$ can be quantified from the perspective of determinantal point process. As mentioned in Kang (2013) and Elfeki et al. (2019), Determinantal Point Process (DPP) is a framework for representing a probability distribution that models diversity. More specifically, a DPP over the set $\mathcal{V}$ with $|\mathcal{V}| = N$, given a positive-definite similarity matrix $L \in \mathbb{R}^{N \times N}$, is a probability distribution P_L over any $S \subseteq \mathcal{V}$ in the following form

$$P_L[S] \propto \det(L_s) \ ,$$

where L_s is the similarity kernel of the subset S[2]. Since the point process according to this probability distribution naturally capture the notion of diversity, we hope to generate a subset with high $P_L[\boldsymbol{D}_{ug}]$ where the $\mathcal{V}$ is viewed as $\mathcal{D}_{ug}$ and the transition matrix is viewed as the similarity kernel. One way to generate a set of unseen samples with high $\det(L_{\boldsymbol{D}_{ug}})$ is to encourage the diagonality of the transition matrix which can be achieved by promoting orthogonality of the generated samples. Moreover, since actually f_{ug} is a look-up table, low $\sum_{j \neq i} p_{ji}$ can be explained as the large dis-similarity of the generated unseen samples from different class.

[2]The feature representation of the similarity space is typically normalized so the highest eigen value is 1, and hence the determinant (multiplication of the eigen values) is < 1

B MORE DETAILS OF SECTION 5

Algorithm 1 shows the overall training process. The Discriminator and Generator are alternatively optimized. During the training of the Generator (line 11 - 22), we propose to generate unseen attributes (line 12 for interpolation based method and line 12,13 for dictionary based method) and encourage the generations to be realistic and deviate from the seen generations (line 19). After the training of each task, we propose to store the current semantic information and real features in the buffer.

B.1 REGULARIZATION TERMS $\mathcal{R}_D$ AND $\mathcal{R}_G$ LOSS FUNCTION 2

We closely follow Kuchibhotla et al. (2022) for the regularization terms of the Generator and Discriminator. The regularization term on discriminator encourages the semantic embedding to be close to the class center, i.e., at task t

$$\mathcal{R}_D^t = \|D(\mathcal{A}_s^{1:t}) - \boldsymbol{C}_s^{1:t}\|_{\mathsf{F}}^2 \ ,$$

where $\mathcal{A}_s^{1:t}$ is the attribute matrix and $\boldsymbol{C}_s^{1:t}$ is the class mean matrix computed by seen features up until current task. $\|\cdot\|_{\mathsf{F}}$ is the Frobenius norm. The regularization terms on the generator encourage the seen generations to be close to the seen class centers and have moderately distanced to their semantic neighborhoods. $\mathcal{R}_G$ is defined as

$$\mathcal{R}_G = L_{\text{nuclear}} + L_{\text{sal}} \ .$$

L_{nuclear} is the Nuclear loss, defined as

$$L_{\text{nuclear}} = \|\boldsymbol{C}_s^t - \boldsymbol{C}_{sg}^t\|_{\mathsf{F}}^2 \ ,$$

where $\boldsymbol{C}_s^t$ is the class mean matrix computed by seen features of current task, and $\boldsymbol{C}_{sg}^t$ is the class mean matrix computed by generated seen features of current task. L_{sal} is the incremental bidirectional semantic alignment loss defined as

$$\begin{aligned} L_{\text{sal}} = \frac{1}{N_s^t} \sum_{i=1}^{N_s^t} \sum_{j \in \boldsymbol{I}_i} & \| \max\{0, \langle \boldsymbol{C}_{s[j]}, \boldsymbol{C}_{sg[i]} \rangle - (\langle {}^t_{s[i]}, {}^t_{s[j]} \rangle + \varepsilon)\} \|^2 \\ & + \| \max\{0, (\langle {}^t_{s[i]}, {}^t_{s[j]} \rangle - \varepsilon) - \langle \boldsymbol{C}_{s[j]}, \boldsymbol{C}_{sg[i]} \rangle\} \|^2 \ , \end{aligned}$$

where N_s^t is the number of current seen classes at task t, $\boldsymbol{I}_i$ is the neighbor set of class i, ε is the margin error, $\langle \cdot, \cdot \rangle$ is the cosine similarity.

B.2 DETAILS OF GRW LOSS

We provide more details for the terms in GRW loss in Section 5.2.2. The transition probability matrix from seen class centers to unseen samples is defined as

$$P^{\boldsymbol{C} \to \boldsymbol{D}_{ug}} = \sigma(\langle \boldsymbol{C}, \boldsymbol{D}_{ug}^\top \rangle) \ , \tag{20}$$

where $\langle \cdot, \cdot \rangle$ is a similarity measure, and $\sigma(\cdot)$ is a softmax operator applied on rows. In practice, we use negative Euclidean distance for similarity, that is, suppose $\boldsymbol{x}_{ug}$ is the i-th row of $\boldsymbol{D}_{ug}$ and $\boldsymbol{c}$ is the j-th class center,

$$\langle \boldsymbol{C}, \boldsymbol{D}_{ug}^\top \rangle_{i,j} = -\|\boldsymbol{x}_{ug} - \boldsymbol{c}\|^2 \ .$$

Similarly, the transition probability matrix from unseen samples to unseen samples and to seen class centers are defined as

$$P^{\boldsymbol{D}_{ug} \to \boldsymbol{D}_{ug}} = \sigma(\langle \boldsymbol{D}_{ug}, \boldsymbol{D}_{ug}^\top \rangle), \quad P^{\boldsymbol{D}_{ug} \to \boldsymbol{C}} = \sigma(\langle \boldsymbol{D}_{ug}, \boldsymbol{C}^\top \rangle) \ . \tag{21}$$

Then the random walk staring from each seen class center and taking R step transitions within unseen samples has probability

$$P^{\boldsymbol{C}\boldsymbol{D}_{ug}\boldsymbol{C}}(R) = P^{\boldsymbol{C} \to \boldsymbol{D}_{ug}} \cdot (P^{\boldsymbol{D}_{ug} \to \boldsymbol{D}_{ug}})^R \cdot P^{\boldsymbol{D}_{ug} \to \boldsymbol{C}} \tag{22}$$

Figure 4: Attribute distribution T-SNE visualizations of AWA1 dataset in different task with interpolation method

B.3 Visualization of Attribute Distribution

Our analysis assumes the generated unseen attributes can compactly support the real unseen attributes. Here we use the T-SNE embedding method to visualize the distribution of them. Figure 4 shows the distribution of seen attributes, unseen attributes, and interpolated attributes in different tasks. We only plotted partial of the generations of every task here. The actual number of generated attributes equals the number of training samples. As the task progresses, the learner sees more and more seen classes, and the generated attributes get closer and closer to the unseen attribute. In the areas where the unseen attribute distribution is sparse, that is the blanks in the figure, the generated attributes are also very sparse. Therefore, the generated invisible attribute tends to describe the possible visual space, has deviation from the visible attribute, and produces compact support for the unseen attributes. This is exactly what we assumed.

B.4 Numerical Verification of GRW Loss

We mentioned in the statement 3.3 that the GRW loss could reduce $\bar{d}_{GDB}$. Here we plot the figure to indicate the relationship between the GRW loss and the bound $\bar{d}_{GDB}$. We use the model at different epochs for different $\hat{h}^*$, and use difference between generated unseen accuracy and test unseen accuracy to represent $\bar{d}_{GDB} = \|\hat{\epsilon}_u - \hat{\epsilon}_{ug}\|$ at a random select task. Figure 5 shows that the $\bar{d}_{GDB}$ has a strong positive correlation, especially when the loss is getting lower. This indicates that we can reduce the bound, i.e., the distance between generated unseen space and true unseen space, by minimizing the GRW loss.

B.5 Comparison of Replay Methods

We mentioned in Section 4.3 that the generative replay method suffers from the increasing buffer and unbalanced class problems. This problem is severe in early tasks. We plot the number of the buffer features of every class at task 2 of the SUN dataset in Figure 6. Our proposed real replay method stores a similar number of features in each class, while the generative replay method stores nearly

Figure 5: Relationship between $\bar{d}_{GDB}$ and the GRW loss in CUB dataset

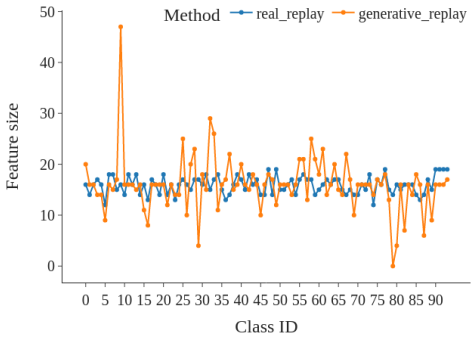

Figure 6: Comparison of replayed number of features per class in diffrent replay method at task 2 in SUN dataset

no features in some classes and doubled number of features in some other classes. However, those classes with less stored features are precisely those where the model does not perform well because it only stores the correctly classified generated data in the generative replay method. This may make the model perform worse in these classes in future tasks.

B.6 Relation to Other Work using Random Walk

We adapt random walk modeling (Ayyad et al., 2021) with three key changes

1. In Ayyad et al. (2021); Häusser et al. (2017), class prototypes/centers are represented by the few examples provided for each class. In our setting, these class prototypes represent seen classes that in our case we want to deviate from seen classes and enable knowledge transfer to unseen classes. To facilitate such transfer through attributes/semantic descriptions, we define seen class centers $\boldsymbol{C}$ in a semantically guided way computed by as the mean of generated seen samples from their corresponding attributes ($[\boldsymbol{C}]_i =$ mean $G(z, a_i)$ for class i defined by attribute vector a_i).
2. Ayyad et al. (2021); Häusser et al. (2017) use unlabeled data points to calculate the random walk, where we use generated examples.
3. These generated examples, in our case, are from unseen classes instead of unlabeled examples of the seen classes as in the few-shot learning problem. Hence, Ayyad et al. (2021); Häusser et al. (2017)'s loss is to attract unlabeled samples to labeled samples, but our goal is the opposite to rather push unseen samples away from seen samples. Häusser et al. (2017) encourage global consistency by using a random walk from labeled data to unlabeled data (their $\boldsymbol{A}$ matrix) and back to labeled samples ($\boldsymbol{A}^\top$ matrix). It promotes identity distribution of paths ($\boldsymbol{A}\boldsymbol{A}^\top \rightarrow I$) where the starting and ending points are of the same class. Ayyad et al. (2021) investigates a more general case in which the number of random walk steps between unlabeled classes ($\boldsymbol{B}$ matrix) is greater than one ($\boldsymbol{A}\boldsymbol{B}^R\boldsymbol{A}^\top$). Since we assume that none of the generated unseen samples belong to the seen classes, and as a deviation signal, we encourage that all of the paths of the random walk from seen to generated examples of unseen classes and back to seen classes (our $\boldsymbol{A}\boldsymbol{B}^R\boldsymbol{A}^\top$) to be uniform instead of identity.

van Laarhoven & Marchiori (2017) focuses on unsupervised domain adaptation, which involves doing a random walk over all potential labeling circumstances on unlabeled target data in order to identify stationary labeling distribution. Labeling stability is defined from the perspective of a generalization bound which can be attained through a stationary Markov chain. We borrow the idea of using the Markov chain to estimate the relationship between different labeling to find a stationary one that can reduce the generalization bound. We employ the Markov chain to estimate the relationship between different unseen generations and discover a diverse one that can reduce the generalization bound. L_GRW loss encourages the random walk to find a highly diverse unseen generation which will reduce the generalization bound.

Algorithm 1: Training procedure of ICGZSL

Input : Total task number T, training epoch E, random walk length R, decay rate of random walk γ, and coefficients $\lambda_{c,rd,i,rg}$, learning rate $\alpha_{G,D,Dic}$, buffer size B
Data : $X_s^{1:T}$, $y_s^{1:T}$, $a_s^{1:T}$
Initialize : Generator, Discriminator

for $t = 1 : T$ **do**
  Get train loader by concatenating train set t with buffer data;
  for $e = 1 : E$ **do**
    Get X_s^t, y_s^t sampled from train loader. Get $a_s^{1:t}$ from current train set and buffer ;
    begin Train Discriminator
      Generate samples conditioning on seen attributes $X_{sg}^t = G(z, a_s^t)$;
      Compute real-fake loss $\mathcal{L}_{\text{real-fake}}$ in equation (**??**) using real seen samples X_s^t, generated seen samples X_{sg}^t, and current task attribute a_s^t;
      Compute classification loss $\mathcal{L}_{\text{classification}}$ in equation **??** using real seen samples X_s^t, generated seen samples X_{sg}^t, and attributes $a_s^{1:t}$;
      Compute $\mathcal{L}_D$ in equation 2 and update $\theta_D \leftarrow \theta_D - \alpha_D \nabla \mathcal{L}_D$;
    end
    begin Train Generator
      Generate a_{ug}^t by interpolation between two random a_s^t ;
      if *Use dictionary based method* **then**
        Initialize the dictionary with the interpolated attribute and get θ_{Dic}
      end
      Generate samples conditioning on unseen attributes $X_{ug}^t = G(z, a_{ug}^t)$;
      Compute the second part of real-fake loss $\mathcal{L}_{\text{real-fake}}$ in equation (**??**) using generated unseen samples X_{sg}^t and current task attribute a_s^t;
      Compute the second part of classification loss $\mathcal{L}_{\text{classification}}$ in equation **??** using generated unseen samples X_{sg}^t and attributes $a_s^{1:t}$;
      Compute tehe inductive loss in $\mathcal{L}_{\text{inductive}}$ using $C_s^t = \text{mean}(X_s^t)$, generated seen samples X_{sg}^t, and unseen generated samples X_{ug}^t Compute $\mathcal{L}_G$ in equation 2 and update $\theta_G \leftarrow \theta_G - \alpha_G \nabla \mathcal{L}_G$;
      if *Use dictionary based method* **then**
        $\theta_{Dic} \leftarrow \theta_{Dic} - \alpha_{Dic} \nabla \mathcal{L}_D$
      end
    end
  end
  begin Replay data by section 4.2.3
    Save a_s^t to the buffer;
    Save current real features with size $B/N_s^{1:t}$ per class, reduce previous features to size $B/N_s^{1:t}$
  end
end

C ZERO-SHOT LEARNING EXPERIMENTS

C.1 TEXT BASED ZERO-SHOT LEARNING EXPERIMENTS

Text-based ZSL is more challenging because the descriptions are at the class level and are extracted from Wikipedia, which is noisier.

Benchmarks We perform experiments on existing zero-shot learning (ZSL) benchmarks, CUB (Wah et al., 2011) and NAB (Van Horn et al., 2015), with text descriptions as semantic class descriptions. Caltech UCSD Birds-2011 (CUB) contains 200 classes with 11, 788 images, and North America Birds (NAB) has 1011 classes with 48, 562 images. To evaluate the generalization capability of class-level text zero-shot recognition, we split the two benchmarks into four subsets (CUB Easy, CUB Hard, NAB Easy, and NAB Hard). Hard splits are constructed such that the unseen bird classes from super-categories do not overlap with seen classes (Chao et al., 2016; Zhu et al., 2018; Elhoseiny & Elfeki, 2019).

Baseline and training We add the proposed GRW loss ($L_{GRW} + \mathcal{R}_{GRW}$) to the inductive zero-shot learning method GAZSL (Zhu et al., 2018) and compare it with other inductive zero-shot learning methods. For the text representation function $\psi(\cdot)$, we used the TF-IDF(Salton & Buckley, 1988) representation of the input text followed by an FC noise suppression layer. We use the random walk length $R = 10$ for our experiments, and show that longer random walk process is more helpful in ablation study. We launch every ZSL experiment on a single NVIDIA V100 GPU.

Evaluation and metrics During test, the visual features of unseen classes are synthesized by the generator conditioned on a given unseen text description a_u, i.e. $x_u = G(s_u, z)$. We generate 60 different synthetic unseen visual features for each unseen class and apply a simple nearest neighbor classifier on top of them. We use two metrics: standard zero-shot recognition with the Top-1 unseen class accuracy and Seen-Unseen Generalized Zero-shot performance with Area under Seen-Unseen curve (Chao et al., 2016).

Results Our proposed approach improves over older methods on all datasets and achieves SOTA on both Easy and SCE(hard) splits, as shown in Table 1 in the introduction section. We show improvements in 0.8-1.8% Top-1 accuracy and 1-1.8% in AUC. GAZSL (Zhu et al., 2018) + GRW also has an improvement of around 2% over other inductive loss (GAZSL (Zhu et al., 2018) + CIZSL (Elhoseiny & Elfeki, 2019)).

GRW Loss for Transductive ZSL To better understand how the GRW improves the consistency of generated seen features space and generated unseen features space, we conduct experiments on semantic transductive zero-shot learning settings. The improvements are solely from the GRW loss with the ground truth semantic information. We choose LsrGAN (Vyas et al., 2020) as the baseline model. Our loss can also improve LsrGAN on text-based datasets on most metrics ranging from 0.3%-3.6%. However, as we expected, the improvement in the purely inductive/more realistic setting is more significant.

Ablation Table 7 shows the results of our ablation study on the random walk length. We find that the longer random walk performs better, giving higher accuracy and AUC scores for both easy and hard splits for CUB Dataset. With a longer random walk process, the model could have a more holistic view of the generated visual representation that enables better deviation of unseen classes from seen classes.

GRW loss contains two parts, L_{GRW} and $\mathcal{R}_{GRW}$. Table 8 shows the results of our ablation study on the $\mathcal{R}_{GRW}$ in zero-shot learning. We perform experiments both with $\mathcal{R}_{GRW}$ and without $\mathcal{R}_{GRW}$. Training failed with NaN gradients in 5% of the times without $\mathcal{R}_{GRW}$ but 0% with $\mathcal{R}_{GRW}$; thus, it is important for the training stability.

C.2 ATTRIBUTE BASED ZERO-SHOT LEARNING EXPERIMENTS

Benchmarks We perform these experiments on the AwA2 (Lampert et al., 2009), aPY (Farhadi et al., 2009), and SUN (Patterson & Hays, 2012) datasets.

Table 7: Ablation studies on CUB Dataset (text). Each row shows either baseline deviation losses and or GRW losses with different length on GAZSL (Zhu et al., 2018)

Setting	CUB-Easy		CUB-Hard	
	Top-1 Acc (%)	SU-AUC (%)	Top1-Acc (%)	SU-AUC (%)
+ GRW (R=1)	45.41	39.62	13.79	12.58
+ GRW (R=3)	45.11	39.25	14.21	13.22
+ GRW (R=5)	45.40	40.51	14.00	13.07
+ GRW (R=10)	**45.43**	**40.68**	**15.51**	**13.70**

Table 8: Ablation study using Zero-Shot recognition on **CUB** & **NAB** datasets with two split settings. We experiment with and without the $\mathcal{R}_{GRW}$ (second and last row). The first loss is the baseline method.

Metric	Top-1 Accuracy (%)				Seen-Unseen AUC (%)			
Dataset	CUB		NAB		CUB		NAB	
Split-Mode	Easy	Hard	Easy	Hard	Easy	Hard	Easy	Hard
GAZSL (Zhu et al., 2018)	43.7	10.3	35.6	8.6	35.4	8.7	20.4	5.8
GAZSL (Zhu et al., 2018) + GRW	**45.4**	**15.5**	**38.4**	10.1	**40.7**	**13.7**	**25.8**	**7.4**
GAZSL (Zhu et al., 2018) + only L_{GRW}	45.3	14.8	38.2	**10.3**	40.1	12.8	25.8	7.4

Baseline, training, and evaluation We perform experiments on the widely used GBU (Xian et al., 2018a) setup, where we use class attributes as semantic descriptors. The evaluation process and training devices are the same as text-based experiments. We use seen accuracy, unseen accuracy, harmonic mean of seen and unseen accuracy, and top-1 accuracy as the evaluation metrics.

Results In Table 9, we see that GRW outperforms all of the existing methods on the seen-unseen harmonic mean for AwA2, aPY, and SUN datasets. In the case of the AwA2 dataset, it outperforms all the compared methods by a significant margin, i.e., 15.1% in harmonic mean, and is also competent with existing methods in Top-1 accuracy while improving 4.8%. GAZSL (Zhu et al., 2018)+GRW has an average relative improvement over GAZSL (Zhu et al., 2018)+CIZSL (Elhoseiny & Elfeki, 2019) and GAZSL (Zhu et al., 2018) of 24.92% and 61.35% in harmonic mean.

Table 9: Zero-Shot Recognition on class-level attributes of **AwA2**, **aPY** and **SUN** datasets, showing that GRW loss can improve the performance on attribute-based datasets.

	Top-1 Accuracy(%)			Seen-Unseen H		
	AwA2	aPY	SUN	AwA2	aPY	SUN
SJE (Akata et al., 2015)	61.9	35.2	53.7	14.4	6.9	19.8
LATEM (Xian et al., 2016)	55.8	35.2	55.3	20.0	0.2	19.5
ALE (Akata et al., 2016)	62.5	39.7	58.1	23.9	8.7	26.3
SYNC (Changpinyo et al., 2016)	46.6	23.9	56.3	18.0	13.3	13.4
SAE (Kodirov et al., 2017)	54.1	8.3	40.3	2.2	0.9	11.8
DEM (Zhang et al., 2016)	67.1	35.0	61.9	25.1	19.4	25.6
FeatGen (**?**)	54.3	42.6	60.8	17.6	21.4	24.9
cycle-(U)WGAN (Felix et al., 2018)	56.2	44.6	60.3	19.2	23.6	24.4
LsrGAN (*tr*) (Vyas et al., 2020)	60.1	34.6	62.5	48.7	31.5	44.8
+ **GRaWD**	63.7$^{+3.6}$	35.5$^{+0.9}$	**64.2**$^{+1.7}$	**49.2**$^{+0.5}$	**32.7**$^{+1.2}$	**46.1**$^{+1.3}$
GAZSL (Zhu et al., 2018)	58.9	41.1	61.3	15.4	24.0	26.7
+ CIZSL (Elhoseiny & Elfeki, 2019)	67.8	42.1	63.7	24.6	25.7	27.8
+ **GRaWD**	**68.4**$^{+9.5}$	**43.3**$^{+2.2}$	62.1$^{+0.8}$	39.0$^{+23.6}$	27.2$^{+3.2}$	27.9$^{+1.2}$

Table 10: Seen and Unseen classes in different dataset

	AWA1	AWA2	CUB	SUN
Total classes	50	50	200	705
Number of tasks	5	5	20	15
Initial seen classes	10	10	10	47
Covered class	10	10	10	47

Table 11: Ablation studies on how the $L_{\text{creativity}}$ influence our method on CUB dataset

	without $L_{\text{creativity}}$		with $L_{\text{creativity}}$	
	Inter.	Dic.	Inter.	Dic.
w/o GRW loss	14.43	14.43	19.07	20.75
w/ GRW loss	27.72	27.66	28.4	28.8

D CONTINUAL ZERO-SHOT LEARNING EXPERIMENTS

D.1 DATASET AND CONTINUAL ZERO-SHOT LEARNING SETUP

We display the seen and unseen class conversions in each task for each dataset in the Table 10 to provide a better understanding of the specific implementation of CZSL on different datasets. Covered class means the number of unseen class converted to seen class per task.

D.2 METRICS

We use the mean seen accuracy, mean unseen accuracy and mean harmonic seen/unseen accuracy (Skorokhodov & Elhoseiny, 2021) to measure the zero-shot learning ability. These metrics are defined as follows,

$$\text{mSA} = \frac{1}{T}\sum_{t=1}^{T} S_t(\hat{D}^{\leq t}),\ \text{mHA} = \frac{1}{T-1}\sum_{t=1}^{T-1} H(S_t(\hat{D}^{\leq t}), U_t(\hat{D}^{\geq t})),\ \text{mUA} = \frac{1}{T-1}\sum_{t=1}^{T-1} U_t(\hat{D}^{\geq t})\ ,$$

where $H(\cdot,\cdot)$ is the harmonic mean and S_t, U_t are seen and unseen per-class accuracy using the model trained after time t. We also use the backward transfer (Chaudhry et al., 2019; Yi & Elhoseiny, 2021; Skorokhodov & Elhoseiny, 2021) to measure the continual learning ability, which is defined in Skorokhodov & Elhoseiny (2021)

$$\text{BWT} = \frac{1}{T-1}\sum_{t=1}^{T-1}(S_T(\hat{D}^{\leq t}) - S_t(\hat{D}^{\leq t}))\ .$$

Note that this should only be conducted on seen set, since part of the early unseen set become seen set later. The BWT on unseen set cannot reflect the knowledge retain ability of the model.

D.3 MORE ABLATIONS

Influence of $L_{\text{creativity}}$ $L_{\text{creativity}}$ serves as our baseline, borrowed from Elhoseiny & Elfeki (2019). We do an ablation here on how $L_{\text{creativity}}$ influence each component of our random walk loss in table 11. The results show that $L_{\text{creativity}}$ improves the generated unseen as a baseline method alone without our proposed GRW loss. With GRW loss, $L_{\text{creativity}}$ serves as an auxiliary term to further encourage characterization of the generations.

Random seed We experiment with multiple random seeds on the CUB dataset and show the averaged mH (line) and standard deviation (shadow) in Figure 7. The random seed mainly affects the generation part of GZSL learners. The generated data is used directly or indirectly to train the classifier of the unseen class. Figure 7 shows that previous models are sensitive to random seeds, but our model is not. Previous models use the generated data as replay data or directly train the classifier, while ours avoids these. Our method uses a non-parametric classifier, a similarity-based classifier. During training, we pay more attention to improving the generalization ability of our embedder (discriminator) by encouraging the consistency between the generated visual space and the true visual space. Plus, we store the real data in the buffer. These all make our model more stable. Although we only reported the results of one seed (2222) in Table 4, the figure shows that the effect of different seeds on the results is not significant.

We also report mean and standard deviation of multiple runs of our methods in each dataset in Table 12, 13. It shows that experiments on all the dataset with both attribute generation methods have

Table 12: Our method in continual zero shot learning setting with interpolated attributes. Mean and variance calculated on three runs with different random seeds.

	mSA		mUA		mHA	
	Mean	Std	Mean	Std	Mean	Std
AWA1	65.87	1.19	33.77	1.00	42.69	0.57
AWA2	70.52	0.46	34.52	0.90	44.45	0.79
CUB	42.11	0.88	22.10	0.67	27.80	0.53
SUN	36.29	0.18	21.07	0.33	26.44	0.20

Table 13: Our method in inductive continual zero shot learning with learnable dictionary of attributes. Mean and variance calculated using three runs with different random seeds

	mSA		mUA		mHA	
	Mean	Std	Mean	Std	Mean	Std
AWA1	66.35	0.28	32.75	0.94	41.90	0.91
AWA2	70.55	0.51	33.88	0.60	43.49	0.88
CUB	42.22	0.30	22.78	0.91	28.09	0.68
SUN	36.63	0.12	21.39	0.47	26.79	0.37

Table 14: The hyperparameter for Table 3

	AWA1		AWA2		CUB		SUN	
	Interpolation	Dictionary	Interpolation	Dictionary	Interpolation	Dictionary	Interpolation	Dictionary
λ_c	10	1	1	10	1	1	1	1
λ_i	0.5	2	1	5	2	2	5	1
R	3	3	3	3	5	5	5	5

relatively small variance. Although interpolation based method has lower mean harmonic accuracy on fine-grained dataset CUB and SUN, it is shown to be more stable with less variance than dictionary based method.

D.4 Hyperparameters in GRW loss

Hyperparameter for Table 3 We use the validation set to tune the hyperparameter random walk step R, coefficient of $L_{\text{creativity}}$ λ_c, and coefficient of $L_{\text{inductive}}$ λ_i. The hyperparameter used to report Table 3 is shown in Table 14

Walk length R and decay rate γ We do an ablation study on the random walk length R and decay rate γ of the GRW loss in continual zero-shot learning experiments. Table 15 shows our method with different random walk lengths in AWA1 dataset and CUB dataset. In the dataset AWA1, moderate lengths give the highest mHA while in the CUB dataset higher random walk lengths provide the best mHA. It shows that the more challenging the dataset, the more random walk length is needed. Unlike ZSL experiments, in CZSL experiments, knowledge is not only transferred to the unseen class space

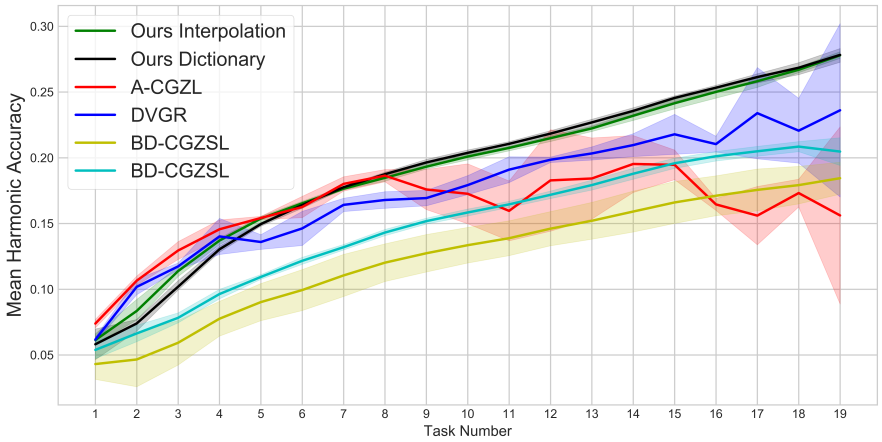

Figure 7: Mean harmonic accuracy at end of each task with 5 different random seeds on CUB (Wah et al., 2011). Lines show the averaged mH, and shadows show the standard deviation.

Table 15: Our method with different random walk length R

(a) Experiments on CUB dataset

		Ours Interpolation			Ours dictionary		
		mSA	mUA	mHA	mSA	mUA	mHA
R	1	41.7	21.2	27.1	43.6	22.7	28.4
	3	42.3	22.1	27.7	42.1	20.6	26.6
	5	42.2	22.7	**28.4**	42.4	23.6	**28.8**

(b) Experiments on AWA1 dataset

		Ours Interpolation			Ours dictionary		
		mSA	mUA	mHA	mSA	mUA	mHA
R	1	65.4	33.9	42.7	66.6	32.7	41.5
	3	67.0	34.2	**43.4**	67.1	33.5	**42.8**
	5	65.8	33.0	42.1	66.8	32.7	41.7

Table 16: Our method with different decay rate γ on CUB dataset

(a) Experiments on CUB dataset

		Ours-interpolation			Ours-dictionary		
		mSA	mUA	mH	mSA	mUA	mH
γ	0.7	40.97	21.78	27.26	42.22	22.03	27.47
	1	40.95	21.21	27.05	42.62	21.6	27.43

(b) Experiments on AWA1 dataset

		Ours-interpolation			Ours-dictionary		
		mSA	mUA	mH	mSA	mUA	mH
γ	0.7	66.8	33.42	42.87	66.93	32.41	41.51
	1	66.07	32.31	41.69	66.34	32.87	41.76

Table 17: Our method with different inductive coefficients λ_i

(a) Experiments on CUB dataset

		Ours-interpolation			Ours-dictionary		
		mSA	mUA	mH	mSA	mUA	mH
λ_i	0.01	41.81	20.93	27.01	42.8	23.07	28.51
	0.1	42.32	21.27	27.11	42.73	21.98	27.85
	1	40.97	21.78	27.26	42.22	22.03	27.47

(b) Experiments on AWA1 dataset

		Ours-interpolation			Ours-dictionary		
		mSA	mUA	mH	mSA	mUA	mH
λ_i	0.1	66.81	32.82	42.15	66.32	32.11	41.15
	1	66.8	33.42	42.87	66.93	32.41	41.51
	10	66.38	33.77	42.92	66.47	31.81	40.89

but also to the next task. Long walk length could give the model a more holistic view of the current task but may harm the transformation to the next task. Therefore tuning the number of random walk steps is required for new datasets.

Decay rate γ works as a scale factor to the GRW loss to prevent a specific area in the probability matrix from being too close to one, resulting in exponential growth in the multiplication results when compared to other areas. Compared to the non-decay case when $\gamma = 1$ in Table 16, the decayed case has noticeable improvements in unseen accuracy, resulting in better harmonic accuracy.

Inductive weight λ_i We also do an ablation study on the inductive coefficient λ_i in Table 17. This factor mainly affects the proportion of inductive loss in the overall loss. We found that our model is not sensitive to this hyperparameter. Whether on the larger dataset CUB or the smaller dataset AWA1, the difference of mH of different λ_i on our model does not exceed 1%. Therefore, our model does not need too much parameter tuning process.

D.5 Continual zero-shot learning with other common settings

Although our main research problem is inductive setting, and we think real replay is needed, we still have an open attitude to other settings and migrate our model naively to their setting. We show experiment results in these settings in Table 18 and compare them with other methods.

We mentioned earlier that the generative replay method has unbalanced storage and buffer overload problems, but many models still use generative replay. When data privacy concerns are encountered, the generative replay method may be an alternative to real replay method. When using the generative replay, our model outperforms most existing methods. Our problem analysis cannot be applied in this setting since we believe the replayed feature should have a balanced number in each class.

Our primary focus is on the inductive setting, but we also provide results in the transductive setting and with generative replay. In the transductive setting, we use the ground truth unseen attributes to

Table 18: Comparison of our inductive loss in other common CZSL settings

	replay method	zsl setting	AWA1			AWA2			CUB			SUN		
			mSA	mUA	mHA	mSA	mUA	mHA	mSA	mUA	mHA	mSA	mUA	mHA
CN-ZSL	real	in	-	-	-	33.55	6.44	10.77	44.31	14.8	22.7	22.18	8.24	12.46
Ours-interpolation	real	in	62.9	32.77	**42.03**	67.41	35.4	**45.06**	40.17	21.78	27.26	36.29	21.05	26.51
Ours-dictionary	real	in	63.43	32	41.15	68.02	33.22	42.89	41.45	22.03	**27.47**	36.54	21.31	**26.76**
DVGR	generative	tr	65.1	28.5	38	73.5	28.8	40.6	44.87	14.55	21.66	22.36	10.67	14.54
A-CGZSL	generative	tr	70.16	25.93	37.19	70.16	25.93	37.19	34.25	12.42	17.41	17.2	6.31	9.68
BD-CGZSL	generative	tr	67.55	36.04	**47.88**	71.37	38.76	**51.6**	31	23.97	**26.01**	30.08	20.07	23.72
Ours-interpolation	generative	tr	62.43	33.03	42.01	66.84	34.01	43.77	32.53	16.66	21.65	-	-	-
Ours-dictionary	generative	tr	62.34	31.5	40.18	68.07	34.45	44.17	30	16.18	20.55	-	-	-
BD-CGZSL-in	generative	in	62.12	31.51	40.46	67.68	32.88	42.33	37.76	9.089	14.43	34.93	14.86	20.8
Ours-interpolation	generative	in	61.43	34.04	**42.18**	67.34	35.29	**44.95**	29.78	16.86	**21.06**	30.9	18.4	**22.99**
Ours-dictionary	generative	in	62.26	30.88	39.68	67.44	33.68	43.24	28.34	16.94	20.57	30.13	18.56	22.85

generate the visual features, and our loss works on these generations. Our method is comparable with other transductive methods, even without carefully designing how to use the semantic information.

Through these knots, we believe that our model has the possibility of being migrated to other settings and is valuable for further explorations in other settings.