Learning Internal Semantics with Expanded Categories for Generative Zero-Shot Learning

Xiaojie Zhao[1], Shidong Wang[2], and Haofeng Zhang[1(✉)]

[1] School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing 210094, China.
{zhaoxj,zhanghf}@njust.edu.cn
[2] School of Engineering, Newcastle University, Newcastle upon Tyne, NE1 7RU, UK.
shidong.wang@newcastle.ac.uk

Abstract. In recent years, generative Zero-Shot Learning (ZSL) has attracted much attention due to its better performance than traditional embedding methods. Most generative ZSL methods exploit category semantic plus Gaussian noise to generate visual features. However, there is a contradiction between the unity of category semantic and the diversity of visual features. The semantic of a single category cannot accurately correspond to different individuals in the same category. This is due to the different visual expression of the same category. Therefore, to solve the above mentioned problem we propose a novel semantic augmentation method, which expands a single semantic to multiple internal sub-semantics by learning expanded categories, so that the generated visual features are more in line with the real visual feature distribution. At the same time, according to the theory of Convergent Evolution, the sub-semantics of unseen classes are obtained on the basis of the expanded semantic of their similar seen classes. Four benchmark datasets are employed to verify the effectiveness of the proposed method. In addition, the category expansion is also applied to three generative methods, and the results demonstrate that category expansion can improve the performance of other generative methods. Code is available at: https://github.com/njzxj/EC-GZSL.

Keywords: Generative Zero-shot Learning · Category Expansion · Semantic Augmentation · Convergent Evolution.

1 Introduction

Deep learning has driven the rapid development of classification, retrieval, positioning and other fields. However, this development depends on a large number of manually labeled datasets, which are often labor-intensive and time-consuming. In order to mitigate this problem, Zero-Shot Learning (ZSL) [20,29] has been proposed to recognize unseen classes. ZSL makes the training model suitable for unseen classes that do not exist in the training set through category semantics. With the popularity of generative network in the field of image, an increasing number of generative ZSL methods emerge in recent years. Generative ZSL [40]

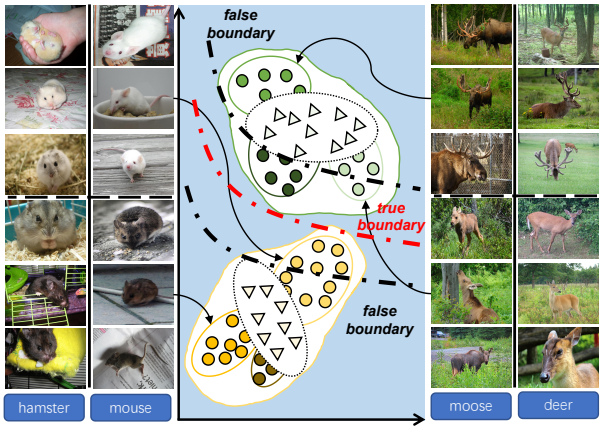


Fig. 1. The motivation of our method. Green and yellow represent moose and mouse respectively.

uses semantics and seen classes to train a generator, and then generates samples of unseen classes with the learned generator to make up for missing samples of unseen classes in the classifier training process. However, the semantic of each class is unique, which leads to the lack of diversity of visual features generated by traditional methods using a single semantic.

The semantic of a category is the summary of the attributes of all individuals in the category, and it is equivalent to a mathematical description of the characteristics of a specific category. From the macro point of view, this description is reasonable for the whole category, but from the micro point of view, it is unreasonable for individuals that it is a biased description. For example, there are three kinds of mouse hair colors: black, white and yellow, so these three color dimensions are marked in the semantic of mouse, and the dimension value representing these attributes is not 0. If we use the semantic that is not 0 in the white attribute dimension to represent the black mouse, then it is wrong. Similarly, the male deer has antlers while the female deer has no antlers, but the semantic of deer is not 0 in the dimension of antlers, so that of deer cannot be used to describe the female deer.

Different individuals in the same category have relativity in the performance of characters, that is, Biological Relative Character. The semantic of a category includes all the characters of the category, but it cannot accurately describe a single individual. Generative ZSL generally generates visual features through category semantics. Because category semantics cannot accurately describe individuals with different relative characters, there must be differences in the distribution of generated visual features and real visual features. We know that the more the generated samples match the real samples, the more beneficial it is to the training of the final classifier. In Figure 1, a category has three internal classes, and the visual feature distribution generated by a single semantic

cannot fully fit the distribution of real visual features. This can lead to inaccurate classification boundaries. Therefore, traditional methods using a single category semantic to generate visual features are unreasonable. That is to say, category semantics cannot reflect the relative characteristics of organisms, and the generated visual features do not accord with the real distribution of visual features.

In order to make the semantic description correct for specific individuals, so as to generate visual features in line with the real distribution, an obvious idea is to recombine the attributes of category semantics to obtain multiple extended category semantics, so that the new semantics can correctly refer to the performance of different relative characteristics of the same category. However, this reorganization is technically difficult if no additional manual annotation is introduced. In terms of solving this problem, we aim to diversify single semantics and obtain the semantics corresponding to the specific expression of relative characteristics. This process can be seen as a more detailed division of a class, and then obtain the semantics of each internal class. A simplified example is to obtain the mouse semantics expressed by different color traits. For example, the other color dimensions of white mouse semantics are 0. Of course, the actual situation is more complicated, because relative characteristics cannot only appear in color.

The problem is transformed into obtaining the semantics of the internal class, that is, obtaining the semantics that can represent white mouse in the mouse category. An effective method is to divide a single class into internal categories, that is, to expand categories for each class. Specifically, firstly, clustering the visual features of each category separately, so that we can obtain the visual prototypes of the internal categories. Then we use a trained mapper to map the visual prototype of the internal classes to the semantic space. In this way, we get the semantics of the internal category.

The method of obtaining the internal class semantics of seen classes is not applicable to unseen classes. In biology, there is the concept of Convergent Evolution [36], that is, different species change their overall or partial morphological structures in the same direction due to similar lifestyles. Figure 1 shows this. Hamster and mouse have the same characteristic expression in hair color. Moose and deer also have similar character expression on antlers. We believe that the characteristics of internal classes of similar species are also similar. Based on this assumption, we propose a method based on semantic similarity, which transfer the semantic expanded results of seen classes to obtain the internal class semantics of unseen classes.

Replace the original category semantics with the expanded semantics to achieve the purpose of generating more real visual features, that is, the generated visual features are more in line with the distribution of real visual features. At the same time, the semantic attributes mapped by visual features are cleverly used in the process of final classification. The reconstructed visual features are obtained from the mapped semantics through the trained generator, and the visual features before mapping are concatenated with reconstructed visual features for the training of the final classifier. This method effectively improves

the performance, because the reconstructed visual features eliminate irrelevant information for classification. In summary, our main contributions are as follows:

- Based on the principle of Relative Character, we propose a category expansion method to learning internal semantics of seen class. In addition, inspired by the thought of Convergent Evolution, the expanded semantics of unseen classes are also learned from those of seen classes.
- Concatenation of synthesized features and reconstructed features is employed to train the final classifier, which can effectively eliminate irrelevant information for classification, thereby improving the performance of the classifier.
- The proposed model is evaluated on four popular datasets and obtains competing results. Furthermore, for the category expansion part, we verified its effectiveness on three classical generative models.

2 Related Work

2.1 Zero-Shot Learning

The whole sample dataset is divided into seen and unseen parts in Zero-Shot Learning. The unseen classes are used in the training phase, and the unseen class is only used in the testing phase. According to whether the test sample contains seen classes, ZSL can be further categorized as conventional ZSL (CZSL) and generalized ZSL (GZSL) [3]. CZSL contains only unseen class samples in the test phase, while GZSL includes both unseen and seen class samples in the test phase. Since in more practical situations, the trained model needs to be applicable to both seen and unseen classes, GZSL has become the mainstream research point. Besides, according to whether to generate unseen visual features to convert ZSL into a fully supervised task, ZSL can also be divided into non-generative methods and generative methods.

Non-generative methods mainly exploits embedding strategies to associate visual features and semantics. Early methods train a projector to map visual features to semantic space [30,33]. However, later researchers consider that projecting visual features into semantic space will cause the serious hubness problem [45,2], so the way of mapping semantics to visual space is adopted to construct visual prototypes. For example, [4,18] project visual features and semantics to public space for classification. [10,1,25] uses a hybrid model based on mapping both semantic and visual features to hidden space. They mainly adopt the joint embedding method of multiple visual features and multiple text representations to connect the attributes with different areas of the image. In recent years, non-generative methods have begun to introduce attention mechanisms [17,42] to strengthen the semantics learning. Furthermore, knowledge graph has also been used to train classifier parameters [37,11], *e.g.*, MSDN [5] adopts the method of knowledge distillation to collaboratively learn attribute-based visual features and visual-based attribute features .

Generative ZSL trains a generator that can generate corresponding visual features according to its class semantics, and then uses the generator to generate

samples of unseen classes to make up for the missing samples of unseen classes. f-CLSWGAN [40] exploits the Generative Adversarial Networks (GAN) [12] plus a classifier to train the generative network. [28,13,14] optimizes the model on this basis by adding more constraints for feature alignment. Cycle-CLSWGAN [9] introduces cyclic consistency loss for feature generator. F-VAEGAN-D2 [41] employs Variational Auto-Encoder (VAE) and uses the distribution obtained by the encoder to replace the Gaussian noise as the input for GAN. Some also directly utilizes VAE to generate visual features [34,26,23]. Besides, IZF [35] adopts the generative flow model instead of generative adversarial network to circumvent the fixed format of Gaussian noise.

2.2 Single Attribute and Relative Character

There is only one semantic for a single category, but the visual features of the same category are diverse. Semantics can be regarded as mathematical description of categories, and they include all the possible characteristics of categories. Even in the same category, the expression of a visual semantic, that is, the theory of relative character in biology, is different. Relative Character refers to that the same species often have different performances in characters. Therefore, a single semantic cannot well represent each individual in a category. Most non generative methods map visual features to semantic space. In this case, the mapping is biased. Some non generative method [22,24,37] has put forward this problem and alleviated this problem to a certain extent.

Most generative methods generate visual features by means of a single semantic and Gaussian noise, so as to enrich visual features for unseen classes. However, the semantic description of visual features generated in this way is still less diverse, because they are all generated under the same semantic description. This method cannot generate visual features that conform to the real distribution, because the expression of different individual characters in the same category is different, so the corresponding semantics should also be different. In order to solve the problem that a single semantic cannot correctly correspond to the representation of different visual features, we propose a category expansion based feature generation method.

2.3 Convergent Evolution

Convergent Evolution means that different species show the same or similar characteristics under the influence of specific conditions. For example, mouse and hamster are two different categories, and they have similarities in the expression of hair color. Moose and deer are different species, and they have similarities in the expression of antlers. If we can expand the semantics of seen classes, we can expand the semantics of similar unseen classes according to the theory of Convergent Evolution.

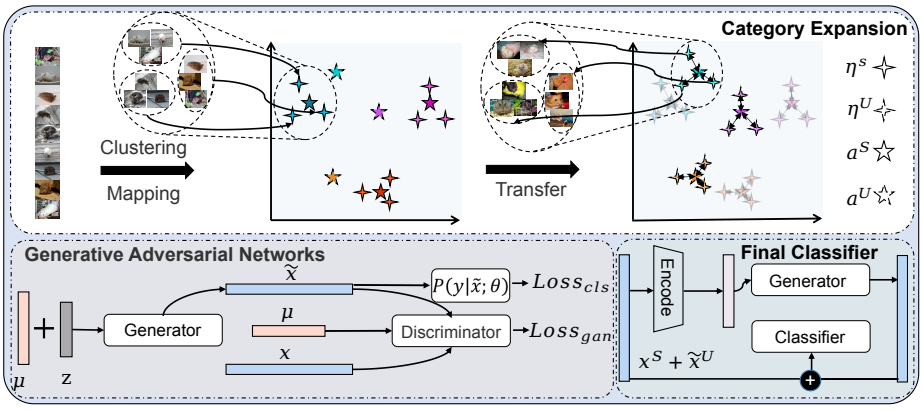


Fig. 2. The architecture of our proposed method. Category Expansion module shows how we can get the internal classes of seen and unseen classes.

3 Methodology

3.1 Problem Definition

Suppose there are S seen classes for training and U unseen classes for testing. We use C_s to represent seen classes and C_u to represent unseen classes, where $C_s \cap C_u = \varnothing$. $X^s = \left\{x_1^s, x_2^s, x_3^s, \cdots, x_{N_s}^s\right\} \subset \mathbb{R}^{d_x \times N_s}$ are the visual features of seen classes, where d_x is the dimension of visual features and N_s is the number of instances of seen classes. And the corresponding sample class labels are $Y^s = \left\{y_1^s, y_2^s, y_3^s, \cdots, y_{N_s}^s\right\}$. $X^u = \left\{x_1^u, x_2^u, x_3^u, ..., x_{N_u}^u\right\} \subset \mathbb{R}^{d_x \times N_u}$ is the visual feature of the sample in unseen classes, where N_u is the number of instances with unseen classes. And the corresponding sample class label is $Y^u = \left\{y_1^u, y_2^u, y_3^u, \cdots, y_{N_u}^u\right\}$. $A = \left\{a^1, a^2, ...a^s, a^{S+1}, ..., a^{S+U}\right\} \subset \mathbb{R}^{d_a \times (S+U)}$ represents category semantics, where d_a is the dimension of semantics. The first S vectors are seen classes and the last U are unseen classes. Our goal is to learn a classifier $f : x \rightarrow C$ to classify the visual feature x in the category space. For CZSL x only belongs to unseen classes and $C = C_s \cup C_u$, while x belongs to both seen and unseen classes and $C = C_u$ for GZSL.

3.2 Overall Idea

In order to diversify the generated visual features, traditional generative GZSL methods mostly use a single semantic plus Gaussian noise to generate synthesized unseen samples. However, the visual features obtained in this way do not match the feature distribution of real samples. This is because the category semantic corresponding to the generated visual features are single, while the real semantics corresponding to the real sample features are diverse. In order to make

the semantics corresponding to the generated visual features more diverse, we propose a method to augment the semantics by expanding categories of seen classes based on the theory of species Relative Character. In this method, the samples will be clustered in the visual space. Then, the clustering centers are mapped to the semantic space and will replace the original semantics. Next, in order to augment the semantics of unseen classes, we propose a method to extend the diversified semantics of seen classes to unseen classes based on the theory of Convergent Evolution. This method uses the similarity between seen classes and unseen class to augment the semantics of unseen class. Last, we use the augmented semantics to replace the original single semantic of the category for the training of Generative Adversarial Network, and introduce a simple reconstructed visual feature to improve the performance of the final classifier. Figure 2 shows the overall model framework.

3.3 Category Expansion

Category semantics are determined according to the character expression of the categories. It can be regarded as the attribute description of the category. For example, if a category has characters such as angle and claw, the corresponding semantic dimension will have numerical value. In order to uniformly describe a class, as long as any individual of the class has a specific character expression, its corresponding semantic dimension exists and is a fixed value. A typical example is that male deer have antlers and female deer have no antlers, but the dimension of antlers in deer semantics is fixed.

A single category semantic is the sum of the expression of all characters in the category, while different individuals in the same category have different performance of relative characters, which leads to the inability of a single semantic to generate true and accurate visual features. We cannot generate white mouse with the semantics of non-zero values in the dimensions corresponding to black, white and yellow. Of course, black mouse cannot be generated. Similarly, using the semantics of deer can generate male deer with antlers, but not female deer. Obviously, traditional generative methods that using this strategy are unreasonable. For a category, the visual features generated using a single semantic are inaccurate, such as mice of different colors, and incomplete, such as female deer without antlers. In Generative ZSL, the closer the generated visual features are to the real visual features, the better the performance of the generator, and it is more beneficial to downstream tasks. In order to solve this problem, we need to obtain the semantics of individuals expressing different characters in the same category. Therefore, we need to expand category, in other words, to obtain the internal semantics of different categories of mouse.

In order to obtain the semantics of different characters of the same category, that is, the semantics of internal categories, we use the method of clustering the samples of each category in visual space and then mapping them to the semantic space. We assume that the number of internal classes in a category is k, in other words, we assume that the number of clusters of each class in the visual space is k. Let $D_i = \left\{x_1^i, x_2^i, x_3^i, \cdots, x_{N_i}^i\right\}$ represent the features of the samples belonging

the ith class, and use the k-means algorithm to minimize the square error of the visual cluster division $C^i = \left\{C_1^i, C_2^i, C_3^i, \cdots, C_k^i\right\}$ of this class:

$$L = \sum_{j=1}^{k} \sum_{x^i \in C_j^i} \left\| x^i - \mu_j^i \right\|_2^2, \tag{1}$$

where $\mu_j^i = \frac{1}{\left|C_j^i\right|} \sum_{x^i \in C_j^i} x^i$ is the mean vector of cluster C_j^i. The clustering center of each internal class is regarded as the visual prototype of the internal class. We use μ^S to represent all μ_j^i for seen classes.

Then, we need a mapper to map visual prototypes to semantic space. The semantic information of visual features is extracted by a mapper E. We use $a' = E\left(x^s\right)$ to represent the semantic of extracted visual features. In order to make the mapped semantics correct, we use the category semantics to constrain the mapper. The loss function is as follows:

$$Loss_e = \frac{1}{N_S} \sum_{i=1}^{N_S} \left\| a_i' - a_i \right\|_F^2, \tag{2}$$

where a_i is the category semantics corresponding to the feature x_i^s. When the mapper E is obtained, we map the visual prototype μ^S of the seen category to the semantic space:

$$\eta^S = E\left(\mu^S\right), \tag{3}$$

where η^S represents the semantics of internal seen classes. In this way, we expand the category of each seen class and obtain the semantics expressed by different characters of the same class.

3.4 Augmentation Transfer

Because unseen class samples are not available in the training stage, we cannot obtain the semantics of their internal classes by using the method which used in seen classes. However, the hair color difference of mice is also reflected in hamsters, and the antler difference of deer is also reflected in moose. This phenomenon is called Convergent Evolution, that is, different species evolve into phenomena with similar morphological features or structures due to similar environment and other factors. Based on Convergent Evolution, we can transfer the semantics of seen classes through the similarity measurement between seen and unseen classes, so as to obtain the semantics of the expanded categories of unseen classes.

In order to migrate the differences between the internal classes of seen class to unseen class, we first need to obtain the differences between the internal classes of the seen class. We use $Z = \left\{z_1^1, \cdots, z_k^1, z_1^2, \cdots, z_k^2, \cdots, z_j^i, \cdots z_k^s\right\}$ to represent the distance between the jth internal class of the ith seen class and the original semantic a^i:

$$z_j^i = a^i - \eta_j^i. \tag{4}$$

Based on the theory of Convergent Evolution, similar categories have similar character expression. In order to transfer the expanded semantics, we need to obtain the seen class that is most similar to each unseen class. We calculate category similarity based on category semantics. For the semantic a^p of each unseen class, we calculate its semantic similarity with each seen class:

$$h_{pq} = \frac{a^p \cdot a^q}{\|a^p\| \|a^q\|} \left(S+1 \leq p \leq S+U, 1 \leq q \leq S\right), \tag{5}$$

where h_{pq} represents the semantic similarity between pth unseen class and qth seen class. We define the tags:

$$\tau_p = argmax_{q \in (1,2,\ldots,S)} h_{pq}. \tag{6}$$

We think that seen class τ_p and unseen class p are the most similar in character expression. Obviously, their internal class distribution also have similarities according to class semantics. Therefore, we can get the internal class semantics of unseen classes:

$$\eta^p = \{\eta_1^p, \eta_2^p, \cdots, \eta_k^p\} = \left\{a^p - \alpha h_{p\tau_p} z_1^{\tau_p}, \cdots, a^p - \alpha h_{p\tau_p} z_k^{\tau_p}\right\}, \tag{7}$$

where α is a hyper-parameter to reduce the deviation caused by the semantic transfer process.

3.5 Feature Generation

When we have got the semantics of the internal classes, we will use them to replace the original semantic of each sample. We use the replaced semantics to train the Generative Adversarial Networks. The internal class semantics η and Gaussian noise z are employed to generate visual feature through generator G:

$$\tilde{x} = G\left(\eta, z\right), \tag{8}$$

where $z \sim N\left(0,1\right)$ is the Gaussian noise. The generated visual features and the real visual features are discriminated by discriminator D, which is optimized with WGAN [27]. The total loss is as follows:

$$Loss_{gan} = \mathbb{E}\left[D\left(x,\eta\right)\right] - \mathbb{E}\left[D\left(\tilde{x},\eta\right)\right] - \lambda \mathbb{E}\left[\left(\left\|\nabla_{\hat{x}} D\left(\hat{x},\eta\right)\right\|_2 - 1\right)^2\right], \tag{9}$$

where η is the internal class semantics of the generated sample $\tilde{x}$, and it is also the internal class semantic corresponding to the real visual feature x. The last one is the gradient penalty, where $\hat{x} = \alpha x + \left(1-\alpha\right)\tilde{x}$ with $\alpha \sim U\left(0;1\right)$ and λ is the penalty coefficient.

At the same time, in order to make the generated samples more authentic, we follow f-CLSWGAN [40] to constrain the classification loss:

$$Loss_{cls} = -\mathbb{E}\left[logP\left(y|\tilde{x};\theta\right)\right], \tag{10}$$

where y is the real class of $\tilde{x}$, not the internal class. θ is the parameter of the pre-trained classifier on the seen class. It is worthy noted that we do not use internal category tags for classification. Then the total loss is:

$$Loss_{total} = Loss_{gan} + \beta Loss_{cls}, \tag{11}$$

where β is the balancing coefficient.

3.6 Remove Irrelevant Features

The trained mapper can map the visual features to the semantic space. This mapping eliminates irrelevant information which is useless or even harmful to the final classification in a certain extent. We know that the generator can generate visual features for the corresponding semantics. Therefore, the mapped semantics and the reconstructed visual features obtained by the generator are free of irrelevant information. Based on this, we get reconstructed visual features:

$$\check{x} = G\left(E\left(x\right); 0\right), \tag{12}$$

where x contains the features of seen class and the generated features of unseen class. It is noted that the Gaussian noise during generation is set to 0.

3.7 Classification

Now we have obtained the visual features of seen classes, the visual features of the generated unseen class and the reconstructed visual features of both. Considering the bias of the mapper to the seen class, in order to prevent the generated unseen reconstructed visual features from losing classification information in the process of reconstruction, we concatenate the reconstructed visual features with the original visual features for the training of the final classifier. Figure 2 shows the overall process. Let $\bar{x}$ represent the features after concatenating as $\bar{x} = x \oplus \check{x}$, where $\oplus$ represent the feature concatenation, the finally classifier loss is:

$$Loss_{final} = -\mathbb{E}\left[logP\left(y|\bar{x}, \theta_f\right)\right], \tag{13}$$

where θ_f is classifier parameters. Note that in the process of classification, we use the original category label. In other words, we classify each visual feature into corresponding category instead of internal category.

4 Experiments

4.1 Datasets and Setting

We evaluated our method on four datasets. **AWA2** [39] contains 37322 instances of 50 classes, and **aPY** [8] has 32 classes with a total of 15339 instances. The fine-grained dataset **CUB** [38] contains 11788 bird instances of 200 classes, and **SUN** [31] contains 14340 instances of 717 classes. For all datasets, we use 2048

dimensional visual features extracted with ResNet-101 [15]. It should be noted that we use the newly extracted 1024 dimensional semantic attribute for CUB [32]. There are three hyper-parameters. We set $\alpha = 0.5$ for coarse-grained datasets, $\alpha = 0.2$ for fine-grained datasets, and $\beta = 0.01$ for all. It is worthy noted that all hyper-parameters are obtained with cross validation. To be specific, we separate a certain number of the seen classes as the validational unseen. For example, we randomly divide 40 seen classes in AWA2 into 30 seen classes and 10 validational unseen classes multiple times, and select the hyper-parameters that can achieve the best mean performance for final training. Although this operation is a bit different from k-fold cross-validation, it is an effective way for ZSL hyper-parameter selection. In addition, to increase the generalization ability, L_2 regularization is added to train the generator.

4.2 Comparison with Baselines

Table 1 shows the performance comparison of our proposed method with other methods. It can be seen that on the coarse-grained datasets, the results we have obtained are higher than the models proposed in recent years, achieving the state-of-the-art performance, and our result is 70.3% for AWA2 and 45.4% for aPY. For fine-grained datasets, we obtained quite good results on CUB, which is 65.4%. For SUN, the result is not the best, but it remains at the average level.

It should be noted that the internal categories of unseen classes are obtained based on the principle of Convergent Evolution, which is biological. SUN and aPY do not belong to the biological category datasets. This shows that our proposed category expansion method is still effective on a non-biological dataset, which means that the internal categories between similar categories also have the same characteristic on non-biological datasets.

4.3 Verification on Other Generative Models

In order to verify that the Category Expansion we proposed is generally applicable, we use the semantics of expanded categories to replace the original semantics, and then apply them to other generative models. We verify this strategy on three classical generative models, including F-VAEGAN-D2 [41], RFF-GZSL [14] and CE-GZSL [13].

In order to fairly compare the impact of using the new semantics and original semantics on the performance of different models, we reproduce the above three models. We follow the parameters provided in the three articles, but because some parameters are not provided and the experimental platform is different, the reproduced results are a bit different from the original reported results.

For f-VAEGAN-D2, we use 1024 dimensional semantic attributes in the CUB dataset, and other settings follow the parameters given in the paper. For RFF-GZSL, we set batch size to 512. For CE-GZSL, because its batch size has a great impact on performance and has high requirements for GPU card, we cannot set the same batch size given in the paper, which makes the deviation of reproduction results too large. We set the batch size to 128.

Table 1. The results on four datasets. U represents the accuracy of the unseen class, S represents the accuracy of the seen class, and H represents the harmonic mean. The best value of each column is in bold, and '-' means not reported.

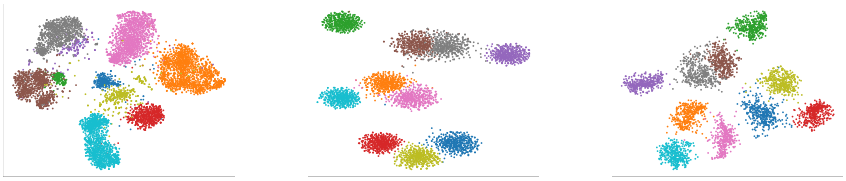
Method	AWA2			aPY			SUN			CUB		
	U	***S***	***H***	***U***	***S***	***H***	***U***	***S***	***H***	***U***	***S***	***H***
f-CLSWGAN [40]	-	-	-	32.9	61.7	42.9	42.6	36.6	39.4	43.7	57.7	49.7
RFF-GZSL [14]	-	-	-	-	-	-	45.7	38.6	41.9	52.6	56.6	54.6
cycle-CLSWGAN [9]	-	-	-	-	-	-	49.4	33.6	40.0	45.7	61.0	52.3
IZF [35]	60.6	77.5	68.0	-	-	-	**52.7**	57.0	**54.8**	52.7	**68.0**	59.4
GDAN [16]	32.1	67.5	43.5	30.4	**75.0**	43.4	38.1	89.9	53.4	39.3	66.7	49.5
CE-GZSL [13]	63.1	78.6	70.0	-	-	-	48.8	38.6	43.1	63.9	66.8	65.3
GCM-CF [44]	60.4	75.1	67.0	**37.1**	56.8	44.9	47.9	37.8	42.2	61.0	59.7	60.3
LisGAN [22]	-	-	-	-	-	-	42.9	37.8	40.2	46.5	57.9	51.6
FREE [6]	60.4	75.4	67.1	-	-	-	47.4	37.2	41.7	55.7	59.9	57.7
SE-GZSL [19]	**80.7**	59.9	68.8	-	-	-	40.7	45.8	43.1	60.3	53.1	56.4
HSVA [7]	59.3	76.6	66.8	-	-	-	48.6	39.0	43.3	52.7	58.3	55.3
E-PGN [43]	52.6	**86.5**	64.6	-	-	-	-	-	-	52.0	61.1	56.2
Disentangled-VAE [23]	56.9	80.2	66.6	-	-	-	36.6	47.6	41.4	54.1	58.2	54.4
Ours	67.0	73.9	**70.3**	33.5	70.3	**45.4**	48.5	36.6	41.7	**68.3**	62.8	**65.4**

Table 2. The results of expanded categories on three baseline models. The upper part is the result obtained by using the original semantics, and the lower part is the result obtained using the internal class semantics. 'EC' stands for Expanded Categories.

Method	AWA2			aPY			SUN			CUB		
	U	***S***	***H***	***U***	***S***	***H***	***U***	***S***	***H***	***U***	***S***	***H***
f-VAEGAN-D2	57.5	**68.7**	62.6	**32.9**	61.7	42.9	44.0	**39.8**	41.8	64.0	**67.0**	65.5
f-VAEGAN-D2+EC	**59.8**	68.5	**63.8**	31.7	**67.2**	**43.1**	**49.0**	37.3	**42.4**	**69.0**	64.9	**66.9**
CE-GZSL	57.0	74.9	64.7	9.85	**88.4**	17.3	40.9	**35.4**	37.9	66.3	**66.6**	66.4
CE-GZSL+EC	**62.5**	**75.1**	**68.2**	**35.0**	57.3	**43.5**	**49.7**	32.3	**39.5**	**67.6**	66.3	**66.9**
RFF-GZSL	54.1	**77.7**	63.8	21.0	**87.5**	33.8	42.6	**37.8**	40.0	66.3	63.1	64.7
RFF-GZSL+EC	**60.1**	72.2	**65.6**	**31.6**	71.8	**43.9**	**47.4**	36.4	**41.2**	**67.3**	**63.3**	**65.2**

However, the above parameter settings do not affect our verification because we follow the method of fixed variables. After replacing the original semantics, the parameters of the model are not changed, and the experimental results are compared only on the basis of modifying semantics.

Table 2 shows our experimental results. It can be seen that for coarse-grained datasets AWA2 [21] and aPY, the performance has been significantly improved after replacing semantics. For fine-grained datasets, although the performance is also improved, the effect is not as obvious as that of coarse-grained datasets. The classification on fine-grained datasets is meticulous, if each class is divided into new internal classes, the semantic difference of internal classes is also limited. For this reason, we think the result is reasonable. The experimental results fully prove that our proposed Category Expansion method can be applied to different generation models as a general method and improve the performance of the model.



(a) Real features distribution. (b) Single attribute generation. (c) Semantic diversity generation.

Fig. 3. t-SNE illustration of visual features generated by traditional single attribute generation method and attribute diversity method.

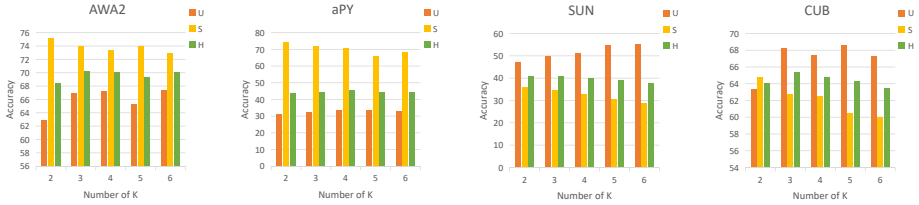


Fig. 4. Results under different number of internal classes.

4.4 Feature Generation

In order to verify that the generated visual features are more consistent with the real visual feature distribution, we visualize the generated visual features. Figure 3 (a) shows the distribution of real unseen visual features under t-SNE. Figure 3 (b) shows the unseen visual features generated by traditional single semantics, to be noted that here we use the classical f-CLSWGAN model. Figure 3 (c) shows the visual features generated by our proposed method. It can be seen that for the visual features generated by a single semantic, each category is close to a regular ellipse. The visual features generated by our category expansion method are more irregular, which is in line with the irregular distribution of real visual feature shown in Figure 3 (a).

4.5 Number of Internal Classes

Different clustering centers also have different impact on final performance. Figure 4 shows the experimental results for different numbers of internal classes. We can see that the best results are $k = 3$ on AWA2, $k = 4$ on aPY, $k = 3$ on CUB and $k = 3$ on SUN. It can seen that after the accuracy H reaches the highest point, the influence of the value of k on the coarse-grained datasets begin to decrease and tend to be stable. But it has a negative impact on fine-grained datasets. Coarse-grained datasets have higher tolerance for the division of internal classes, because coarse-grained datasets have more space for the division

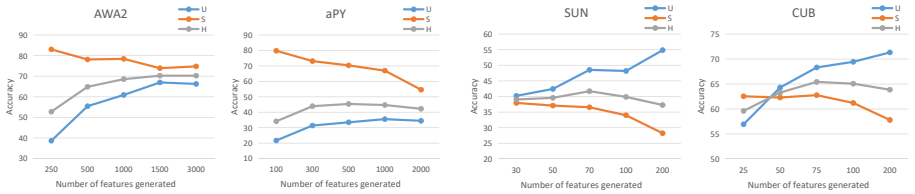


Fig. 5. Results of generating different numbers of unseen visual features.

of internal classes, while the fine-grained datasets have less space for class division, and the wrong internal class division will have a certain impact on the performance of the final classifier.

4.6 Number of Features Generated

The number of unseen visual features generated has an impact on the final experimental results. Figure 5 shows the comparison of results under different numbers of visual features generated. We can see that with the increase of the generated number, the accuracy of seen classes shows a downward trend, and the accuracy of unseen classes shows an upward trend. When the number of visual features generated is 3000 on AWA2, 500 on aPY, 70 on SUN and 75 on CUB, the highest accuracy is achieved due to the balance between the number of unseen and seen classes.

5 Conclusion

In this paper, we have discussed the problem of generating diverse feature with a single semantic in generative ZSL. The visual features generated by traditional semantics do not accord with the distribution of real visual features. Therefore, we have proposed a category expansion method based on Relative Character, and extend the results of semantic augmentation of seen classes to unseen classes based on Convergent Evolution. At the same time, we have employed a simple and efficient way to eliminate irrelevant information of visual features. Our method has achieved good performance on four benchmark datasets. On this basis, we have tested the semantic augmentation module as a general method on three generative ZSL methods, and verified that this semantic augmentation is generally applicable and can improve the performance of generative ZSL.

Acknowledgements This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grants No. 61872187, No. 62072246 and No. 62077023, in part by the Natural Science Foundation of Jiangsu Province under Grant No. BK20201306, and in part by the "111" Program under Grant No. B13022.

References

1. Akata, Z., Malinowski, M., Fritz, M., Schiele, B.: Multi-cue zero-shot learning with strong supervision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 59–68 (2016)
2. Annadani, Y., Biswas, S.: Preserving semantic relations for zero-shot learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7603–7612 (2018)
3. Chao, W.L., Changpinyo, S., Gong, B., Sha, F.: An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In: European conference on computer vision. pp. 52–68. Springer (2016)
4. Chen, L., Zhang, H., Xiao, J., Liu, W., Chang, S.F.: Zero-shot visual recognition using semantics-preserving adversarial embedding networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1043–1052 (2018)
5. Chen, S., Hong, Z., Xie, G.S., Wang, W., Peng, Q., Wang, K., Zhao, J., You, X.: Msdn: Mutually semantic distillation network for zero-shot learning. arXiv preprint arXiv:2203.03137 (2022)
6. Chen, S., Wang, W., Xia, B., Peng, Q., You, X., Zheng, F., Shao, L.: Free: Feature refinement for generalized zero-shot learning. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 122–131 (2021)
7. Chen, S., Xie, G., Liu, Y., Peng, Q., Sun, B., Li, H., You, X., Shao, L.: Hsva: Hierarchical semantic-visual adaptation for zero-shot learning. Advances in Neural Information Processing Systems **34** (2021)
8. Farhadi, A., Endres, I., Hoiem, D., Forsyth, D.: Describing objects by their attributes. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 1778–1785. IEEE (2009)
9. Felix, R., Reid, I., Carneiro, G., et al.: Multi-modal cycle-consistent generalized zero-shot learning. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 21–37 (2018)
10. Fu, Z., Xiang, T., Kodirov, E., Gong, S.: Zero-shot object recognition by semantic manifold distance. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2635–2644 (2015)
11. Geng, Y., Chen, J., Ye, Z., Yuan, Z., Zhang, W., Chen, H.: Explainable zero-shot learning via attentive graph convolutional network and knowledge graphs. Semantic Web (Preprint), 1–28 (2020)
12. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. Advances in neural information processing systems **27** (2014)
13. Han, Z., Fu, Z., Chen, S., Yang, J.: Contrastive embedding for generalized zero-shot learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2371–2381 (2021)
14. Han, Z., Fu, Z., Yang, J.: Learning the redundancy-free features for generalized zero-shot object recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12865–12874 (2020)
15. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
16. Huang, H., Wang, C., Yu, P.S., Wang, C.D.: Generative dual adversarial network for generalized zero-shot learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 801–810 (2019)

17. Huynh, D., Elhamifar, E.: Fine-grained generalized zero-shot learning via dense attribute-based attention. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4483–4493 (2020)
18. Jiang, H., Wang, R., Shan, S., Chen, X.: Learning class prototypes via structure alignment for zero-shot recognition. In: Proceedings of the European conference on computer vision (ECCV). pp. 118–134 (2018)
19. Kim, J., Shim, K., Shim, B.: Semantic feature extraction for generalized zero-shot learning. arXiv preprint arXiv:2112.14478 (2021)
20. Lampert, C.H., Nickisch, H., Harmeling, S.: Learning to detect unseen object classes by between-class attribute transfer. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 951–958. IEEE (2009)
21. Lampert, C.H., Nickisch, H., Harmeling, S.: Attribute-based classification for zero-shot visual object categorization. IEEE transactions on pattern analysis and machine intelligence **36**(3), 453–465 (2013)
22. Li, J., Jing, M., Lu, K., Ding, Z., Zhu, L., Huang, Z.: Leveraging the invariant side of generative zero-shot learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7402–7411 (2019)
23. Li, X., Xu, Z., Wei, K., Deng, C.: Generalized zero-shot learning via disentangled representation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 1966–1974 (2021)
24. Liu, Y., Guo, J., Cai, D., He, X.: Attribute attention for semantic disambiguation in zero-shot learning. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 6698–6707 (2019)
25. Long, Y., Liu, L., Shao, L., Shen, F., Ding, G., Han, J.: From zero-shot learning to conventional supervised classification: Unseen visual data synthesis. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1627–1636 (2017)
26. Ma, P., Hu, X.: A variational autoencoder with deep embedding model for generalized zero-shot learning. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 11733–11740 (2020)
27. Martin Arjovsky, Soumith Chintala, L.B.: Wasserstein gan. Proceedings of ICML 2017 (2017)
28. Ni, J., Zhang, S., Xie, H.: Dual adversarial semantics-consistent network for generalized zero-shot learning. Advances in Neural Information Processing Systems **32** (2019)
29. Palatucci, M., Pomerleau, D., Hinton, G.E., Mitchell, T.M.: Zero-shot learning with semantic output codes. Advances in neural information processing systems **22** (2009)
30. Palatucci, M., Pomerleau, D., Hinton, G.E., Mitchell, T.M.: Zero-shot learning with semantic output codes. Advances in neural information processing systems **22** (2009)
31. Patterson, G., Xu, C., Su, H., Hays, J.: The sun attribute database: Beyond categories for deeper scene understanding. International Journal of Computer Vision **108**(1), 59–81 (2014)
32. Reed, S., Akata, Z., Lee, H., Schiele, B.: Learning deep representations of fine-grained visual descriptions. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 49–58 (2016)
33. Romera-Paredes, B., Torr, P.: An embarrassingly simple approach to zero-shot learning. In: International conference on machine learning. pp. 2152–2161. PMLR (2015)

34. Schonfeld, E., Ebrahimi, S., Sinha, S., Darrell, T., Akata, Z.: Generalized zero- and few-shot learning via aligned variational autoencoders. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8247–8255 (2019)
35. Shen, Y., Qin, J., Huang, L., Liu, L., Zhu, F., Shao, L.: Invertible zero-shot recognition flows. In: European Conference on Computer Vision. pp. 614–631. Springer (2020)
36. Stern, D.L.: The genetic causes of convergent evolution. Nature Reviews Genetics **14**(11), 751–764 (2013)
37. Wang, X., Ye, Y., Gupta, A.: Zero-shot recognition via semantic embeddings and knowledge graphs. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6857–6866 (2018)
38. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., Perona, P.: Caltech-ucsd birds 200 (2010)
39. Xian, Y., Lampert, C.H., Schiele, B., Akata, Z.: Zero-shot learningâĂŤa comprehensive evaluation of the good, the bad and the ugly. IEEE transactions on pattern analysis and machine intelligence **41**(9), 2251–2265 (2018)
40. Xian, Y., Lorenz, T., Schiele, B., Akata, Z.: Feature generating networks for zero-shot learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5542–5551 (2018)
41. Xian, Y., Sharma, S., Schiele, B., Akata, Z.: f-vaegan-d2: A feature generating framework for any-shot learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10275–10284 (2019)
42. Xie, G.S., Liu, L., Jin, X., Zhu, F., Zhang, Z., Qin, J., Yao, Y., Shao, L.: Attentive region embedding network for zero-shot learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9384–9393 (2019)
43. Yu, Y., Ji, Z., Han, J., Zhang, Z.: Episode-based prototype generating network for zero-shot learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 14035–14044 (2020)
44. Yue, Z., Wang, T., Sun, Q., Hua, X.S., Zhang, H.: Counterfactual zero-shot and open-set visual recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15404–15414 (2021)
45. Zhang, L., Xiang, T., Gong, S.: Learning a deep embedding model for zero-shot learning. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2021–2030 (2017)