A Review of Generalized Zero-Shot Learning Methods

Pourpanah, Farhad; Abdar, Moloud; Luo, Yuxuan; Zhou, Xinlei; Wang, Ran; Lim, Chee Peng; Wang, Xi-Zhao; Wu, Q. M. Jonathan

Published in:
IEEE Transactions on Pattern Analysis and Machine Intelligence

Published: 01/04/2023

Document Version:
Final Published version, also known as Publisher’s PDF, Publisher’s Final version or Version of Record

License:
CC BY-NC-ND

Publication record in CityU Scholars:
Go to record

Published version (DOI):
10.1109/TPAMI.2022.3191696

Publication details:

Citing this paper
Please note that where the full-text provided on CityU Scholars is the Post-print version (also known as Accepted Author Manuscript, Peer-reviewed or Author Final version), it may differ from the Final Published version. When citing, ensure that you check and use the publisher's definitive version for pagination and other details.

General rights
Copyright for the publications made accessible via the CityU Scholars portal is retained by the author(s) and/or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights. Users may not further distribute the material or use it for any profit-making activity or commercial gain.

Publisher permission
Permission for previously published items are in accordance with publisher's copyright policies sourced from the SHERPA RoMEO database. Links to full text versions (either Published or Post-print) are only available if corresponding publishers allow open access.

Take down policy
Contact lbscholars@cityu.edu.hk if you believe that this document breaches copyright and provide us with details. We will remove access to the work immediately and investigate your claim.

Download date: 11/03/2024
A Review of Generalized Zero-Shot Learning Methods

Farhad Pourpanah, Member, IEEE, Moloud Abdar, Yuxuan Luo, Xinlei Zhou, Ran Wang, Member, IEEE, Chee Peng Lim, Xi-Zhao Wang, Fellow, IEEE, and Q. M. Jonathan Wu, Senior Member, IEEE

Abstract—Generalized zero-shot learning (GZSL) aims to train a model for classifying data samples under the condition that some output classes are unknown during supervised learning. To address this challenging task, GZSL leverages semantic information of the seen (source) and unseen (target) classes to bridge the gap between both seen and unseen classes. Since its introduction, many GZSL models have been formulated. In this review paper, we present a comprehensive review on GZSL. First, we provide an overview of GZSL including the problems and challenges. Then, we introduce a hierarchical categorization for the GZSL methods and discuss the representative methods in each category. In addition, we discuss the available benchmark data sets and applications of GZSL, along with a discussion on the research gaps and directions for future investigations.

Index Terms—Generalized zero-shot learning, deep learning, semantic embedding, generative adversarial networks, variational auto-encoders

1 INTRODUCTION

With recent advances in image processing and computer vision, deep learning (DL) models have achieved extensive popularity due to their capability for providing an end-to-end solution from feature extraction to classification. Despite their success, traditional DL models require training on a massive amount of labeled data for each class, along with a large number of samples. In this respect, it is a challenging issue to collect large-scale labelled samples. As an example, ImageNet [1], which is a large data set, contains 14 million images with 21,814 classes in which many classes contain only few images. In addition, standard DL models can only recognize samples belonging to the classes that have been seen during the training phase, and they are not able to handle samples from unseen classes [2]. While in many real-world scenarios, there may not be a significant amount of labeled samples for all classes. On one hand, fine-grained annotation of a large number of samples is laborious and it requires an expert domain knowledge. On the other hand, many categories are lack of sufficient labeled samples, e.g., endangered birds, or being observed in progress, e.g., COVID-19, or not covered during training but appear in the test phase [3], [4], [5], [6].

Several techniques for various learning configurations have been developed. One-shot [7] and few-shot [8] learning techniques can learn from classes with a few learning samples. These techniques use the knowledge obtained from data samples of other classes and formulate a classification model for handling classes with few samples. While the open set recognition (OSR) [9] techniques can identify whether a test sample belongs to an unseen class, they are not able to predict an exact class label. Out-of-distribution [10] techniques attempt to identify test samples that are different from the training samples. However, none of the above-mentioned techniques can classify samples from unseen classes. In contrast, human can recognize around 30,000 categories [11], in which we do not need to learn all these categories in advance. As an example, a child can easily recognize zebra, if he/she has seen horses previously, and have the knowledge that a zebra looks like a horse with black and white strips. Zero-shot learning (ZSL) [12], [13] techniques offer a good solution to address such challenge.

ZSL aims to train a model that can classify objects of unseen classes (target domain) via transferring knowledge obtained from other seen classes (source domain) with the help of semantic information. The semantic information embeds the names of both seen and unseen classes in high-
et al. [20] introduced an effective calibration technique, called calibrated stacking, to balance the trade-off between recognizing samples from the seen and unseen classes, which allows learning knowledge about the unseen classes. Since then, the number of proposed techniques under the GZSL setting has been increased radically.

1.1 Contributions

Since its introduction, GZSL has attracted the attention of many researchers. Although several comprehensive reviews of ZSL models can be found in the literature [3], [24], [26], [27]. The main differences between our review paper with previous ZSL survey literature [3], [24], [26], [27] are as follows. In [3], the mainly focus is on ZSL, and only a few GZSL methods have been reviewed. In [24], the impacts of various ZSL and GZSL methods in different case studies are investigated. Several SOTA ZSL and GZSL methods have been selected and evaluated using different data sets. However, the work [24] is more focused on empirical research rather than a review paper on ZSL and GZSL methods. The study in [26] is focused on ZSL, with only a brief discussion (a few paragraphs) on GZSL. Rezaei and Shahidi [27] studied the importance of ZSL methods for COVID-19 diagnosis (medical application). Unlike the aforementioned review paper, in this manuscript, we focus on GZSL, rather than ZSL, methods. However, none of them include an in-depth survey and analysis of GZSL. To fill this gap, we aim to provide a comprehensive review of GZSL in this paper, including the problem formulation, challenging issues, hierarchical categorization, and applications.

We review published articles, conference papers, book chapters, and high-quality preprints (i.e., arXiv) related to GZSL commencing from its popularity in 2016 till early 2021. However, we may miss some of the recently published studies, which is not avoidable. In summary, the main contributions of this review paper include:

- comprehensive review of the GZSL methods, to the best of our knowledge, this is the first paper that attempts to provide an in-depth analysis of the GZSL methods;
- hierarchical categorization of the GZSL methods along with their corresponding representative models and real-world applications;
- elucidation on the main research gaps and suggestions for future research directions.

1.2 Organization

This review paper contains six sections. Section 2 gives an overview of GZSL, which includes the problem formulation, semantic information, embedding spaces and challenging issues. Section 3 reviews inductive and semantic transductive GZSL methods, in which a hierarchical categorization of the GZSL methods is provided. Each category is further divided into several constituents. Section 4 focuses on the transductive GZSL methods. Section 5 presents the applications of GZSL to various domains, including computer vision and natural language processing (NLP).
on the research gaps and trends for future research, along with concluding remarks, is presented in Section 6.

2 OVERVIEW OF GENERALIZED ZERO-SHOT LEARNING

2.1 Problem Formulation

Let \( S = \{(x^i_1, a^i_1, y^i_1)\}_{i=1}^N | x^i_1 \in \mathcal{X}, a^i_1 \in A^s, y^i_1 \in Y^s \} \) and \( U = \{(x^i_2, a^i_2, y^i_2)\}_{i=1}^N | x^i_2 \in \mathcal{X}, a^i_2 \in A^u, y^i_2 \in Y^u \} \) represent the seen and unseen class data sets, respectively, where \( x^i_1, x^i_2 \in \mathbb{R}^D \) indicate the \( D \)-dimensional images (visual features) in the feature space \( \mathcal{X} \) that can be obtained using a pre-trained deep learning model such as ResNet [28], VGG-19 [29], GoogLeNet [30]; \( a^i_1, a^i_2 \in \mathbb{R}^K \) indicate the \( K \)-dimensional semantic representations (i.e., attributes or word vectors) in the semantic space \( \mathcal{A} \); \( Y^s = \{y^s_1, ..., y^s_C^s\} \) and \( Y^u = \{y^u_1, ..., y^u_{C^u}\} \) indicate the label sets of both seen and unseen classes in the label space \( \mathcal{Y} \), where \( C^s \) and \( C^u \) are the number of seen and unseen classes, \( \mathcal{Y} = Y^s \cup Y^u \) denotes the union of both seen and unseen classes and \( Y^s \cap Y^u = \emptyset \). In GZSL, the objective is to learn a model \( f_{\text{GZSL}}: \mathcal{X} \rightarrow \mathcal{Y} \) for classifying \( N \) test samples, i.e., \( D_{ts} = \{x_m, y_m\}_{m=1}^N \) where \( x_m \in \mathbb{R}^D \), and \( y_m \in \mathcal{Y} \).

The training phase of GZSL methods can be divided into two broad settings: inductive learning and transductive learning [24]. Inductive learning utilises only the visual features and semantic information of the seen classes to build a model. In transductive setting, in addition to the seen class information, the semantic representations and unlabelled visual features of the unseen classes are exploited for learning [24], [31]. Fig. 2 illustrates the main difference of transductive GZSL and inductive GZSL settings [32], [33]. As can be seen, the prior of seen classes is available for both settings, but availability of the prior of unseen classes depends on the setting, as follows. In inductive learning setting, there is no prior knowledge of the unseen classes. In transductive learning setting where the models use the unlabelled samples of the unseen classes, the prior of unseen classes is available. Although several frameworks have been developed under transductive learning [34], [35], [36], [37], [38], [39], [40], [41], this learning paradigm is impractical. On one hand, it violates the unseen assumption and reduces challenge. On the other hand, it is not practical to assume that unlabelled data for all unseen classes are available. In addition, some studies [35], [36], [42], [43] under transductive learning employ all image samples of the unseen classes during training, while others [44], [45] split the samples into two equal portions, one for training and another for inference.

Recently, several researchers [33], [46], [47] argued that most of the generative-based methods, which are reviewed in Section 3.2, are not pure inductive learning as the semantic information of the unseen classes is used to generate the visual features for the unseen classes. They categorized such generative-based methods as semantic transductive learning. In addition, they proposed inductive generative-based methods [33], [46], [47] without accessing the semantic information of unseen classes before testing.

2.2 Semantic Information

Semantic information is the key to GZSL. Since there are no labelled samples from the unseen classes, semantic information is used to build a relationship between both seen and unseen classes, thus making it possible to perform generalized zero-shot recognition. The semantic information must contain recognition properties of all unseen classes to guarantee that enough semantic information is provided for each unseen class. It also should be related to the samples in the feature space to guarantee usability of semantic information.

The idea of using semantic information is inspired by human recognition capability. Human can recognize samples from the unseen classes with the help of semantic information. As an example, a child can easily recognize zebra, if he/she has seen horses previously, and have the knowledge that a zebra looks like a horse with black and white strips. Semantic information builds a space that includes both seen and unseen classes, which can be used to perform ZSL and GZSL. The most widely used semantic information for GZSL can be grouped into manually defined attributes [13], word vectors [48], or their combinations.

2.2.1 Manually Defined Attributes

These attributes describe the high-level characteristics of a class (category), such as shape (i.e., circle) and color (i.e., blue), which enable the GZSL model to recognize classes in the world. The attributes are accurate, but require human efforts in annotation, which are not suitable for large-scale problems [49]. Wu et al. [50] proposed a global semantic consistency network (GSC-Net) to exploit the semantic attributes for both seen and unseen classes. Lou et al. [51] developed a data-specific feature extractor according to the attribute label tree.

2.2.2 Word Vectors

These vectors are automatically extracted from large text corpus (such as Wikipedia) to represent the similarities and differences between various words and describe the properties of each object. Word vectors require less human labor, therefore they are suitable for large-scale data sets. However, they contain noise which compromises the model performance. As an example, Wang et al. [52] applied Node2Vec to produce the conceptualized word vectors. The studies in [53], [54], [55], [56] attempted to extract semantic representations from noisy text descriptions, and Akata et al. [57] proposed to extract semantic representations from multiple textual sources.
2.3 Embedding Spaces

Most GZSL methods learn an embedding/mapping function to associate the low-level visual features of the seen classes with their corresponding semantic vectors. This function can be optimized either via a ridge regression loss [59], [60] or ranking loss with respect to the compatibility scores of two spaces [61]. Then, the learned function is used to recognize novel classes by measuring the similarity level between the prototype representations and predicted representations of the data samples in the embedding space. As every entry of the attribute vector represents a description of the class, it is expected that the classes with similar descriptions contain a similar attribute vector in the semantic space. However, in the visual space, the classes with similar attributes may have large variations. Therefore, finding such an embedding space is a challenging task, causing visual semantic ambiguity problems.

On the one hand, the embedding space can be divided into either euclidean or non-euclidean spaces. While the euclidean space is simpler, it is subject to information loss. The non-euclidean space, which is commonly based on graph networks, manifold learning, or clusters, usually uses the geometrical relation between spaces to preserve the relationships among the data samples [27]. On the other hand, the embedding space can be categorized into semantic embedding, visual embedding and latent space embedding [58]. Each of these categories are discussed in the following subsections.

2.3.1 Semantic Embedding

Semantic embedding (Fig. 3a) learns a (forward) projection function from the visual space to the semantic space using different constraints or loss functions, and perform classification in the semantic space. The aim is to force semantic embedding of all images belonging to a class to be mapped to some ground-truth label embedding [61], [62]. Once the best projection function is obtained, the nearest neighbor search can be performed for recognition of a given test image.

2.3.2 Visual Embedding

Visual embedding (Fig. 3b) learns a (reverse) projection function to map the semantic representations (back) into the visual space, and perform classification in the visual space. The goal is to make the semantic representations close to their corresponding visual features [59]. After obtaining the best projection function, the nearest neighbor search can be used to recognize a given test image.

2.3.3 Latent Embedding

Both semantic and visual embedding models learn a projection/embedding function from the space of one modality, i.e., visual or semantic, to the space of other modality. However, it is a challenging issue to learn an explicit projection function between two spaces due to the distinctive properties of different modalities. In this respect, latent space embedding (Fig. 3c) projects both visual features and semantic representations into a common space $L$, i.e., a latent space, to explore some common semantic properties across different modalities [40], [63], [64]. The aim is to project visual and semantic features of each class nearby into the latent space. An ideal latent space should fulfill two conditions: (i) intra-class compactness, and (ii) inter-class separability [40]. Introduced by Zhang et al. [65], this mapping aims to overcome the hubness problem of ZSL models, which is discussed in Section 2.4.

2.4 Challenging Issues

In GZSL, several challenging issues must be addressed. The hubness problem [65], [67] is one of the challenging issues of early ZSL and GZSL methods that learn semantic embedding space and utilize the nearest neighbor search to perform recognition. Hubness is an aspect of the curse of dimensionality that affects the nearest neighbors method, i.e., the number of times that a sample appears within the $k$-nearest neighbors of other samples [68]. Dinu et al. [69] observed that a large number of different map vectors are surrounded by many common items, in which the presence of such items causes problem in high-dimensional spaces.

The projection domain shift problem is another challenging issue of ZSL and GZSL methods. Both ZSL and GZSL models first leverage data samples of the seen classes to learn a mapping function between the visual and semantic spaces. Then, the learned mapping function is exploited to project the unseen class images from the visual space to semantic space. On the one hand, both visual and semantic spaces are two different entities. On the other hand, data samples of the seen and unseen classes are disjoint, unrelated for some classes, and their distributions can be different, resulting in a large domain gap. As such, learning an embedding space using data samples from the seen classes without any adaptation to the unseen classes causes the projection domain shift problem [42], [63], [66]. This problem is more challenging in GZSL, due to the existence of the seen classes during prediction. Since GZSL methods are required to recognize both seen and unseen classes during inference, they are usually biased towards the seen classes (which is discussed as the third problem of GZSL task in the next paragraph), because of the availability of visual features of seen classes during learning. Therefore, it is critical to learn a precise mapping function to avoid bias and ensure the effectiveness of the resulting GZSL models. The projection domain shift problem differs from the vanilla domain shift problem.
The main idea of GZSL is to classify objects of both seen and unseen classes by transferring knowledge from the seen classes to the unseen ones through semantic representations. To achieve this, two key issues must be addressed: (i) how to transfer knowledge from the seen classes to unseen ones; (ii) how to learn a model to recognize images from both seen and unseen classes without having access to the labeled samples of unseen classes [2]. In this regard, many methods have been proposed, which can be broadly categorized into:

Embedding-based methods: learn an embedding space to associate the low-level visual features of seen classes with their corresponding semantic vectors. The learned projection

interpreted as the prior likelihood of a sample from the unseen classes. When $\gamma \to -\infty$, the classifier will classify all data samples into one of the seen classes, and vice versa. Le et al. [77] proposed to find an optimal $\gamma$ that balances the trade-off between accuracy of the seen and unseen classes. Later, several studies used the calibrated stacking technique to solve the GZSL problem [73], [78], [79], [80], [81]. Similar to the calibrated stacking, scaled calibration [82] and probabilistic representation [83], [84] have been proposed to balance the trade-off between both seen and unseen classes. Studies [2], [85] made the unseen classes more confident and the seen classes less confident using temperature scaling [86].

Detectors aim to identify whether a test sample belongs to the seen or unseen classes. This strategy limits the set of possible classes by providing information to which set (seen or unseen) a test sample belongs to. Socher et al. [15] considered that the unseen classes are projected to out-of-distribution (OOD) with respect to the seen ones. Then, data samples from unseen classes are treated as outliers with respect to the distribution of the seen classes. Bhattacharjee [74] developed an auto-encoder-based framework to identify the set of possible classes. To achieve this, additional information, i.e., the correct class information, is imposed into the decoder to reconstruct the input samples.

Later, entropy-based [76], probabilistic-based [75], [87], distance-based [88], cluster-based [89] and parametric novelty detection [50] approaches have been developed to detect OOD, i.e., the unseen classes. Felix et al. [90] learned a discriminative model using the latent space to identify whether a test sample belongs to a seen or unseen class. Geng et al. [91] decomposed GZSL into open set recognition (OSR) [9] and ZSL tasks.

3 REVIEW OF GZSL METHODS

Unlike the vanilla (conventional) domain shift problem where two domains share the same categories [70] and only the joint distribution of input and output differs between the training and test stages [71], the projection domain shift can be directly observed in terms of the projection shift, rather than the feature distribution shift. In addition, the seen (source) domain classes are completely different from the unseen (target) domain classes, and both can even be unrelated [42].

Fig. 4a shows an ideal unbiased mapping function that forces the projected visual samples of both seen and unseen classes to surround their own semantic features in the latent space. In practice, the training and test samples are disjoint in GZSL tasks. This results in learning an unbiased mapping function for the seen classes, which can project the visual features of the unseen classes far away from their semantic features (see Fig. 4b). This problem is more common in inductive-based methods, as they have no access to the unseen class data during training. To overcome this problem, inductive-based methods incorporate additional constraints or information from the seen classes. Besides that, several transductive-based methods have been developed to alleviate the projection domain shift problem [35], [36], [37], [38]. These methods use the manifold information of the unseen classes for learning.

Since GZSL methods use the data samples from the seen classes to learn a model to perform recognition for both seen and unseen classes, they are usually biased towards the seen classes, leading to misclassification of data from the unseen classes into the seen classes (see Fig. 5), in which most of the ZSL methods cannot effectively solve this problem [72]. To mitigate this issue, several strategies have been proposed, such as calibrated stacking [20], [73] and novelty detector [15], [74], [75], [76]. The calibrated stacking [20] method balances the trade-off between recognizing data samples from both seen and unseen classes using the following formulation.

\[
\hat{y} = \arg \max_{c \in \mathcal{Y}} f_c(x) - \gamma \Pi_{c \in \mathcal{Y}^s},
\]

where $\gamma$ is a calibration factor and $\Pi[\cdot] \in \{0,1\}$ indicates whether $c$ is from seen classes or otherwise. In fact, $\gamma$ can be
function is used to recognize novel classes by measuring the similarity level between the prototype representations and predicted representations of the data samples in the embedding space (see Fig. 6a).

**Generative-based methods:** learn a model to generate images or visual features for the unseen classes based on the samples of seen classes and semantic representations of both classes. By generating samples for unseen classes, a GZSL problem can be converted into a conventional supervised learning problem (see Fig. 6b). Based on single homogenous process, a model can be trained to classify the test samples belonging to both seen and unseen classes and solve the bias problem. Although these methods perform recognition in the visual space and they can be categorized as visual embedding models, we separate them from the embedding based methods.

A hierarchical categorization of both methods, together with their sub-categories, is provided in Fig. 7.

### 3.1 Embedding-Based Methods

In recent years, various embedding-based methods have been used to formulate a framework to tackle GZSL problems. These methods can be divided into graph-based, attention-based, autoencoder-based, meta learning, compositional learning and bidirectional learning methods, as shown in Fig. 7. In the following subsections, we review each of these categories (note that compositional learning and other methods are reviewed in the supplementary), and provide a summary of these methods in Table 4 in the supplementary.

#### 3.1.1 Out-Of-Distribution Detection-Based Methods

Out-of-distribution or outlier detection aims to identify data samples that are abnormal or significantly different from other available samples. Several studies applied outlier detection techniques to solve GZSL tasks [15], [74], [75], [92]. First, outlier detection techniques are employed to separate the seen class instances from those of the unseen classes. Then, domain expert classifiers (seen/unseen), e.g., standard classifiers for seen classes and ZSL methods for unseen classes, are adopted to separately classify seen and unseen class data samples. Lee et al. [92] developed a novelty detector based on a hierarchical taxonomy based on language information. Their method builds a taxonomy with the hypernym-hyponym relationships between known classes, in a way that objects of unseen classes are expected to be categorized into one of the most relevant seen classes. Atzmon and Chechik [75] devised a probability based gating mechanism and introduced a Laplace-like prior into the gating mechanism to distinguish seen class samples from the unseen ones. However, training the gate is a challenging issue since visual features from the unseen classes are unavailable during training. To solve this issue, boundary-based OOD classifier [93], semantic encoding classifier [94] and domain detector based on entropy gate [74] have been proposed to separate seen class samples from the unseen ones. In addition, the studies in [95], [96] integrated OOD techniques into generative methods to tackle GZSL tasks.

#### 3.1.2 Graph-Based Methods

Graphs are useful for modelling a set of objects with a data structure consisting of nodes and their relationships (edges) [97]. Graph learning leverages machine learning techniques...
to extract relevant features, mapping properties of a graph into a feature vector with the same dimensions in the embedding space. Machine learning techniques convert graph-based properties into a set of features without projecting the extracted information into a lower dimensional space [98]. Generally, each class is represented as a node in a graph-based method. Each node is connected to other nodes (i.e., classes) through edges that encode their relationships. The geometric structure of features in the latent space is preserved in a graph, leading to a compact representation of richer information, as compared with other techniques [63]. Nonetheless, learning a classifier using structured information and complex relationships without visual examples for the unseen classes is a challenging issue, and the use of graph-based information increases the model complexity.

Recently, graph learning techniques have shown effective paradigm in GZSL [63], [99], [100], [101], [102], [103]. For example, the shared reconstructed graph (SRG) [63] (Fig. 8) uses the cluster center of each class to represent the class prototype in the image feature space, and reconstructs each semantic prototype as follows:

$$e_k = \sum_{i=1}^{K} a_i^k b_i^k = A b_k, \quad s.t. \quad b_k^2 = 0,$$

where $b_k \in \mathbb{R}^{K \times 1}$ includes the reconstruction coefficients, and $K = 1, \ldots, C_s + C_u$. After learning the relationships among classes, the shared reconstruction coefficients between two spaces is learned to synthesize image prototypes for the unseen classes. SRG contains a sparsity constraint, which enables the model to divide the classes into many clusters of different subspaces. A regularization term is adopted to select fewer and relevant classes during the reconstruction process. The reconstruction coefficients are shared, in order to transfer knowledge from the semantic prototypes to image prototypes. In addition, the unseen semantic embedding method is used to mitigate the projection domain shift problem, in which the graph of the seen image prototypes is adopted to alleviate the space shift problem.

AGZSL [99], i.e., asymmetric graph-based ZSL, combines the class-level semantic manifold with the instance-level visual manifold by constructing an asymmetric graph. In addition, a constraint is made to project the visual and attribute features orthogonally when they belong to different classes. The studies in [52], [104], [105], [106] exploit the graph convolutional network (GCN) [107] to transfer knowledge among different categories. In [52], [104], GCN is applied to generate super-classes in the semantic space. The cosine distance is used to minimize the distance between the visual features of the seen classes and corresponding semantic representations, and a triplet margin loss function is optimized to avoid the hubness problem. Discriminative anchor generation and distribution alignment (DAGDA) [105] uses a diffusion-based GCN to generate anchors for each category. Specifically, a semantic relation regularization is derived to refine the distribution in the anchor space. To mitigate the hubness problem, two auto-encoders are employed to keep the original information of both features in the latent space. Besides that, Xie et al. [106] devised an attention technique to find the most important regions of the image and then used these regions as a node to represent the graph.

3.1.3 Meta Learning-Based Methods

Meta learning, which is also known as learning to learn, is a subset of learning paradigms that learns from other learning algorithms. It aims to extract transferable knowledge from a set of auxiliary tasks, in order to devise a model while avoiding the overfitting problem. The underlying principle of a meta learning method helps identify the best learning algorithm for a specific data set. Meta learning improves the performance of learning algorithms by changing some aspects according to the experimental results and by optimizing the number of experiments. Several studies [108], [109], [110], [111], [112], [113] utilized meta learning strategy to solve GZSL problems. Meta learning based GZSL methods divide the training classes into two sets, i.e., support and query, which correspond to the seen and unseen classes. Different tasks are trained by randomly selecting the classes from both the support and query sets. This mechanism helps meta learning methods to transfer knowledge from the seen to unseen classes, therefore alleviating the bias problem [112].
Sung et al. [108] trained an auxiliary parameterized network using a meta learning method. The aim is to parameterize a feedforward neural network for GZSL. Specifically, a relation module is devised to compute the similarity metric between the output of a cooperation module and the feature vector of a data sample. Then, the learned function is used for recognition. Introduced in [109], the meta-learning method consists of a task module to provide an initial prediction and a correction module to update the initial prediction. Various task modules are designed to learn different subsets of training samples. Then, the prediction from a task module is updated through training of the correction module. Other examples of generative-based meta-learning modules for GZSL are also available, e.g., [110], [111], [112], [113].

3.1.4 Attention-Based Methods

Unlike other methods that learn an embedding space between global visual features and semantic vectors, attention-based methods focus on learning the most important image regions. In other words, the attention mechanism seeks to add weights into deep learning models as trainable parameters to augment the most important parts of the input, e.g., sentences and images. In general, attention-based methods are effective for identifying fine-grained classes because these classes contain discriminative information only in a few regions. Following the general principle of an attention-based method, an image is divided into many regions, and the most important regions are identified by applying an attention technique [83], [114]. One of the major advantages of the attention mechanism is its capability to recognize important information pertinent to performing a task in an input, leading to improved results. On the other hand, the attention mechanism generally increases computational load, affecting real-time implementation of attention-based methods.

The dense attention zero-shot learning (DAZLE) [83] (Fig. 9) obtains visual features by focusing on the most relevant regions pertaining to each attribute. Then, an attribute embedding technique is devised to adjust each obtained attribute feature with its corresponding attribute semantic vector. A similar approach is applied to solve multi-label GZSL problems in [115]. The attentive region embedding network (AREN) [114] incorporates an attentive compressed second-order embedding mechanism to automatically discover the discriminative regions and capture the second-ordered appearance differences. In [116], semantic representations are used to guide the visual features to generate an attention map. In [33], an embedding space was formulated by measuring the focus ratio vector for each dimension using a self-focus mechanism. Zhu et al. [31] used a multi-attention model to map the visual features into a finite set of feature vectors. Each feature vector is modeled as a $C$-dimensional Gaussian mixture model with the isotropic components. Using these low-dimensional embedding mechanisms allows the model to focus on the most important regions of the image as well as remove irrelevant features, therefore reducing the semantic-visual gap. In addition, a visual oracle is proposed for GZSL to reduce noise and provide information on the presence/absence of classes.

The gaze estimation module (GEM) [117], which is inspired by the gaze behavior of human, i.e., paying attention to the parts of an object with discriminative attributes, aims to improve the localization of discriminative attributes. In [118], the localized attributes are used for projecting the local features into the semantic space. It exploits a global average pooling scheme as an aggregation mechanism to further reduce the bias (since the obtained patterns for the unseen classes are similar to those of seen ones) and improve localization. In contrast, the semantic-guided multi-attention (SGMA) localization model [119] jointly learns from both global and local features to provide a richer visual expression.

Liu et al. [120] used a graph attention network [121] to generate optimized attribute vector for each class. Ji et al. [122] proposed a semantic attention network to select the same number of visual samples from each training class to ensure that each class contributes equally during each training iteration. In addition, the model proposed in [123] searches for a combination of semantic representations to separate one class from others. On the other hand, Paz et al. [56] designed visually relevant language [124] to extract relevant sentences to the objects from noisy texts.

3.1.5 Bidirectional Learning Methods

This category leverages the bidirectional projections to fully utilize information in data samples and learn more generalizable projections, in order to differentiate between the seen and unseen classes [58], [80], [125], [126]. In [125], the visual and semantic spaces are jointly projected into a shared subspace, and then each space is reconstructed through a bidirectional projection learning. The dual-triplet network (DTNet) [126] uses two triplet modules to construct a more discriminative metric space, i.e., one considers the attribute relationship
between categories by learning a mapping from the attribute space to visual space, while another considers the visual features.

Guo et al. [80] considered a dual-view ranking by introducing a loss function that jointly minimizes the image view labels and label-view image rankings. This dual-view ranking scheme enables the model to train a better image-label matching model. Specifically, the scheme ranks the correct label before any other labels for a training sample, while the label-view image ranking aims to rank the respective images to their correct classes before considering the images from other classes. In addition, a density adaptive margin is used to set a margin based on the data samples. This is because the density of images varies in different feature spaces, and different images can have different similarity scores. The model proposed in [58] consists of two parts: (i) visual-label activating that learns an embedding space using regression; (ii) semantic-label activating that learns a projection function from the semantic space to the label space. In addition, a bidirectional reconstruction constraint between the semantic and labels is added to alleviate the projection shift problem.

Zhang et al. [32] explained a class level overfitting problem. It is related to parameter fitting during training without prior knowledge about the unseen classes. To solve this problem, a triple verification network (TVN) is used for addressing GZSL as a verification task. The verification procedure aims to predict whether a pair of given samples belongs to the same class or otherwise. The TVN model projects the seen classes into an orthogonal space, in order to obtain a better performance and a faster convergence speed. Then, a dual regression (DR) method is proposed to regress both visual features and semantic representations to be compatible with the unseen classes.

3.1.6 Autoencoder-Based Methods

Autoencoders (AEs) are unsupervised learning techniques that leverage NNs for representation learning. They learn how to compress/encode the data firstly. Then, they learn how to reconstruct the data back into a representation as close to the original data as possible. In other words, AEs exploit an encoder to learn an embedding space and then employ a decoder to reconstruct the inputs. The main advantage of AEs is that they can be trained in an unsupervised manner.

AEs have been widely used to solve GZSL problem. To achieve this, a decoder can be imposed by an additional constraint for learning different mappings [127]. The framework proposed by Biswas and Annadani [128] integrates the similarity level, e.g., cosine distance, into an objective function to preserve the relations. *Latent space encoding* (LSE) [129] explores some common semantic characteristics between different modalities and connects them together. For each modality, an encoder-decoder framework is exploited, in which an encoder is used to decompose the inputs as a latent representation while a decoder is employed to reconstruct the inputs. *Product quantization* ZSL (PQZSL) [130] defines an orthogonal common space, which learns a codebook, to project the visual features and semantic representations into a common space using a center loss function [131] and an auto-encoder, respectively. The orthogonal common space enables the classes to be more discriminative, thus the model can achieve better performances. In addition, PQZSL compresses the visual features into compact codes using quantizers to approximate the nearest neighbors. In addition, study [132] adopts two variational auto-encoder (VAE) models to learn the cross-modal latent features from the visual and semantic spaces, respectively. Cross alignment and distribution alignment strategies are devised to match the features from different spaces.

3.2 Generative-Based Methods

Generative-based methods are originally designed to generate examples from the existing ones to train DL models [133] and compensate the imbalanced classification problems [134]. Recently, these methods have been adopted to generate samples (i.e., images or visual features) for unseen classes by leveraging their semantic representations. The generated data samples are required to satisfy two conflicting conditions: (i) semantically related to the real samples, and (ii) discriminative so that the classification algorithm can classify the test samples easily. To satisfy the former condition, some underlying parametric representations can be used, while the classification loss function can be used to satisfy the second condition [135]. Generative adversarial networks (GANs) [136] and variational autoencoders (VAEs) [137] are two prominent members of generative models that have achieved good results in GZSL. In the following sub-sections, a review on generative-based methods under the semantic transductive learning setting is presented. Note that due to the page limit, the review of other generative based methods is provided in the supplementary. In addition, a summary of this category is provided in Table 5 in the supplementary.

3.2.1 Generative Adversarial Networks

GANs generate new data samples by computing the joint distribution \( p(y, x) \) of samples utilizing the class conditional density \( p(x|y) \) and class prior probability \( p(y) \). GANs consist of a generator \( G_{SV} : Z \times A \rightarrow \mathcal{X} \) that uses semantic attributes \( A \) and a random uniform or Gaussian noise \( z \in Z \), to generate visual feature \( \tilde{x} \in \mathcal{X} \), and a discriminator \( D_V : \mathcal{X} \times A \rightarrow [0, 1] \) that distinguishes real visual features from the generated ones. When a generator learns to synthesize data samples for the seen classes conditioned on their semantic representations \( A_s \), it can be used to generate data samples for the unseen classes through their semantic representations \( A_u \). However, the original GAN models are difficult to train, and there is a lack of variety in the generated samples. In addition, the mode collapse is a common issue in GANs, as there are no explicit constraints in the learning objective. To overcome this issue and stabilize the training procedure, many GAN models with alternative objective functions have been developed. Wasserstein GAN (WGAN) [139] mitigates the mode collapse issue using the Wasserstein distance as the objective function. It applies a weight clipping method on the discriminator to allow incorporation of the Lipschitz constraint. The improved WGAN model [140] reduces the negative effects of WGAN by utilizing gradient penalty instead of weight clipping.
Fig. 10. A general view of the f-CLSWGAN model [139]. It learns a generative model based on the visual features of seen classes conditioned on their corresponding semantic representations. It also uses classification loss to generate more discriminative visual features.

Xian et al. [138] devised a conditional WGAN model with a classification loss to synthesize visual features for the unseen classes, which is known as f-CLSWGAN (see Fig. 10). To synthesize the related features, the semantic feature is integrated into both generator and discriminator by minimizing the following loss function:

\[
L_{WGAN} = E[D(\hat{x}; a^s)] - E[D(\hat{x}; a^a)] - \lambda E[(\|\nabla_x D(\hat{x}; a^s)\|_2 - 1)^2],
\]

where \(\hat{x} = G(z, a^s)\), \(\hat{x} = \alpha x^s + (1 - \alpha) \tilde{x}\) with \(\alpha \sim U(0, 1)\), \(\lambda\) is penalty factor and discriminator \(D: \mathcal{X} \times \mathcal{C} \rightarrow \mathbb{R}\) is a multi-layer perceptron. In (3), the first and second terms approximate the Wasserstein distance, while the last term is the gradient penalty. In addition, to generate discriminative features, the negative log-likelihood is used to minimize the classification loss, i.e.,:

\[
L_{CLS} = -E_{\tilde{x} \sim p_{\tilde{x}|c}} \log P(y^s|\tilde{x}; \theta),
\]

where \(P(y^s|\tilde{x}; \theta)\) is the probability that \(\tilde{x}\) is predicted as class \(y^s\), which is estimated by a softmax classifier by \(\theta\) and optimized by visual features of the seen classes. The final objective function can be written as:

\[
\min_G \max_D L_{WGAN} + \beta L_{CLS},
\]

where \(\beta\) is the weighting factor.

Since then, the conditional GAN (CGAN) has been combined with different strategies to generative discriminative visual features for the unseen classes, including optimal transport-based approach [141], TFGNSCS [142] which is an extended version of f-CLSWGAN that considers transfer information, meta-learning approach [110] which is based on model-agnostic meta-learning [143], LisGAN [144] which focuses on extracting information from soul samples, SPGAN [145] which devises a similarity preserving loss with classification loss, Semantic rectifying GAN (SR-GAN) [146] which employs the semantic rectifying network (SRN) [147] to rectify features, CIZSL [148] which is inspired by the human creativity process [149], MGA-GAN [150] which uses multi-graph similarity, and MKFNet-NFG [151] which is adaptive fusion module based on the attention mechanism. Xie et al. [152] proposed cross knowledge learning and taxonomy regularization to train more relevant semantic features and generalized visual features, respectively. In addition, Bucher et al. [135] developed four different conditional GANs, i.e., generative moment matching network (GMMN) [153], AC-GAN [154], denoising auto-encoder [155], and adversarial auto-encoder [156], to generate data samples for the unseen classes. The empirical results indicate that GMMN outperforms other methods.

However, the generative-based models synthesize highly the unconstrained visual features for the unseen classes that can produce synthetic samples far from the actual distribution of real visual features. To alleviate this issue and address the unpaired training issue during visual feature generation, Verma et al. [67] proposed a cycle consistency loss (which is discussed further in the next sub-section), to ensure the generated visual features map back into their respective semantic space. Using this feedback mechanism allows the model to generate more discriminative visual features. In addition, the reconstructing generated features, which are unlabeled, are useful for the model to operate as a semi-supervised setting.

Later, the studies in [157], [158], [159], [160], [161], [162], [163] devise a cycle consistency loss term in their objective functions. Specifically, DASCN [157], which is a dual learning model, combines a classification loss function with a semantics-consistency adversarial loss function. In the cycle-consistent adversarial network for ZSL (CANZSL) [159], visual features are first synthesized from noisy text. Then an inverse adversarial network is adopted to convert the generated features into text, in order to ensure that the synthesized visual features accurately reflect the semantic representations. In addition, in studies [160], [161], [162], a multi-modal cycle consistency loss function is developed to preserve the semantic consistency of the generated visual features. AFC-GAN [161] introduces a boundary loss function to maximize the decision boundary between the seen and unseen features. Boomerang-GAN [162] uses a bidirectional auto-encoder to judge the reconstruction loss of the generated features and semantic embedding. Moreover, two-level adversarial visual-semantic coupling (TACO) [164] maximizes the joint likelihood of visual and semantic features by augmenting a generative network with inference. This joint learning allows the model to better capture the underlying modes of the data distribution.

3.2.2 Variational Autoencoders (VAEs)

VAE models identify the relationship between data sample \(x\) and the distribution of the latent representation \(z\). A parameterized distribution \(q_\theta(z|x)\) is derived to approximate the posterior probability \(p(z|x)\). Similar to GAN, VAE models consist of two components: (i) an encoder \(q_\theta(z|x)\) with parameter \(\theta\), and (ii) a decoder \(p_\theta(x|z)\) with parameter \(\theta\). The encoder maps sample space \(x\) to latent space \(z\) with respect to its class \(c\), while a decoder maps the latent space back to the sample. Conditional VAE (CVAE) [166] synthesizes sample \(\tilde{x}\) with certain properties by maximizing the lower bound of conditional likelihood \(p(x|a)\), i.e.,

\[
\mathcal{L}(\Phi, \theta; x, a) = -KL(q_\theta(z|x, a) \| p_\theta(z|a)) + E_{q_\theta(z|x, a)}[\log p_\theta(x|z, a)].
\]

Studies [165], [167] use CVAE-based frameworks to generate visual features for GZSL. CVAE-SZL [165] (Fig. 11) adopts a neural network to model both encoder and decoder.
4.1 Embedding-Based Methods

As explained earlier, GZSL methods under the transductive learning setting can alleviate the projection domain shift problem by taking unlabeled data samples from the unseen classes into consideration. Accessing the unlabeled data allows the model to know the distribution of the unseen classes and consequently learn a discriminative projection function. It also permits synthesis of the related visual features for the unseen classes by using generative based methods. Since the number of publications is limited, we categorize transductive-based GZSL methods into two: embedding-based and generative-based methods. Table 6 in the supplementary summarizes the transductive-based GZSL methods.

4.1 Embedding-Based Methods

This category of transductive-based GZSL methods leverages the unlabeled data samples from the unseen classes to learn a projection function between the visual and semantic spaces. Using these unlabeled data samples, the geometric structure of the unseen classes can be estimated. Then, a discriminative projection function in the common space, i.e., semantic, visual, or latent, can be formulated, in order to alleviate the projection domain shift problem. To map the visual features of the seen classes to their corresponding semantic representations, a classification loss function is derived, and an additional constraint is required to extract useful information from the data samples from the unseen classes.

4.2 Generative-Based Methods

Another CVAE-based bidirectional learning model is GDAN [168]. It contains a discriminator to estimate the similarity level with respect to each visual-textual feature pair. The discriminator communicates with two other networks using a dual adversarial loss function. CADA-VAE [169] learns the shared cross-modal latent features of image and class attributes through a set of distribution alignment (DA) and cross alignment (CA) loss functions. A VAE model [138] is exploited to learn the latent features of each data modality, i.e., image and semantic representations. Then, the CA loss function, which can be estimated through decoding the latent feature of a data sample from another modality of the same class, is used for reconstruction. In addition, the DA loss function is used to match generated image and class representations by minimizing their distance. Chen et al. [170] attempted to minimize the uncertainty in the overlapped areas of both seen and unseen classes using an entropy-based calibration. Li et al. [171] employed a multi-variate regressor to map back the output the VAE decoder to the class attributes.

3.2.3 Combined GANs and VAEs

On one hand, VAEs generate blurry images due to the use of the element-wise distance in their structures [172]. On the other hand, the training process of GANs is not stable [173]. To address these limitations, Larsen et al. [172] proposed VAEGAN, in which the discriminator in GAN is used to measure the similarity metric. VAEGAN with similarity measure is able to generate better samples than models with element-wise metrics. Later, Gao et al. [174], [175] proposed a joint generative model (known as Zero-VAE-GAN) to generate high-quality visual features for the unseen classes. More specifically, CVAE conditioned on semantic attributes is combined with CGAN conditioned on both categories and attributes. Besides that, a categorization network and a perceptual reconstruction loss [176] are incorporated to generate high-quality features. Verma et al. [113] leveraged the meta-learning strategy to train a generative model that integrates CVAE and CGAN to generate visual features from data sets that contain few samples for each class. Moreover, Xu et al. [177] proposed a dual learning framework based on CVAE and CGAN with an additional classifier to generate more discriminative visual features for unseen classes.

POURNAPAH ET AL.: REVIEW OF GENERALIZED ZERO-SHOT LEARNING METHODS
Fig. 12. A general view of QFSL [44]. It first projects the visual features of the seen classes into a number of fixed points in the semantic space. Then, a bias loss function (12) is used to map the visual features of the unseen classes into other points.

An example is Quasi-fully supervised learning (QFSL) [44] (Fig. 12), which projects the visual features of the seen classes into a number of fixed points in a semantic space using a fully connected network with the ReLU activation function. The unlabeled visual features from the unseen classes are mapped onto other points with the following loss function.

$$L = \frac{1}{N_s} \sum_{u=1}^{N_s} L_b(x_u) + \frac{1}{N_u} \sum_{u=1}^{N_u} \gamma L_b(x_u) + \lambda \Omega(W), \quad (11)$$

where $L_b$ and $\Omega$ are the classification loss function and regularization factor, $L_b$ is the bias loss, i.e.,

$$L_b = -\ln \sum_{u \in Y^u} p_u, \quad (12)$$

where $p_u$ indicates the predicted probability of the $u$-th class.

Other semantic embedding models have been proposed, e.g., [16], [81] and [178]. Fu et al. [16] developed a semantic manifold structure of the class prototypes distributed in the embedding space. A dual visual semantic mapping paths (DMaP) [178] is formulated to learn the connection between the semantic manifold structure and visual-semantic mapping of the seen classes. It extracts the inter-class relation between the seen and unseen classes in both spaces. For a given test sample, if the inter-class relationship consistency is satisfied, a prediction is produced. On the other hand, adaptive embedding ZSL (AEZSL) [81] learns a visual-semantic mapping for each unseen class by assigning higher weights to the recognition tasks pertaining to more relevant seen classes.

Several studies [37], [38], [45] focus on the development of visual embedding-based models under the transductive setting. Hu et al. [38] leveraged super-class prototypes, instead of the seen/unseen class prototypes, to align with the seen and unseen class domains. The $K$-means clustering algorithm is used to group the semantic representations of all seen and unseen classes into $r$ superclass. The DTN [45], which is a probabilistic-based method, decomposes the training phase into two independent parts. One phase adopts the cross-entropy loss function to learn the labeled seen classes, while the other part applies a combination of cross-entropy loss function and Kullback–Leibler divergence to learn from the unlabeled samples in the unseen classes. In the second phase, cross-entropy is adopted to avoid the transferring of samples from the unseen classes into seen classes.

MFMR [179] employs a matrix tri-factorization [180] framework to construct a projection function in the latent space. Two manifold regularization terms are formulated to preserve the geometric structure in both spaces, while a test time manifold structure is developed to alleviate the shift problem. In [39], [40], the pseudo-labeling technique is used to solve the bias problem by incorporating the testing samples into the training process.

4.2 Generative-Based Methods

This category of GZSL methods uses the unlabeled unseen data samples to generate the related visual features for the unseen classes. A generative model is usually formulated using the data samples of the seen classes, while the unlabeled samples from the unseen classes are used to fine-tune the model based on unsupervised strategies [174], [181]. As an example, Gao et al. [174] developed two self-training strategies based on pseudo-labeling procedure. The first strategy uses $K$-nearest neighbour (K-NN) algorithm to provide pseudo-labels for the unseen visual features. As such, the semantic representations of the unseen classes are used to generate $N$ fake unseen visual features from the Gaussian distribution. Then, for each unseen class, the average of $N$ fake features is used as an anchor. At the same time, $K$-NN is used to update the anchor. Finally, the top $M$ percent of unseen features are selected to fine-tune the generative model. The second strategy obtains the pseudo-labels directly through the classification probability. SABR-T [34] uses WGAN to generate the latent space for the unseen classes via minimizing the marginal difference between the true latent space representation of the unlabeled samples of unseen classes and the generated space. On the other hand, VAEGAN-D2 [182] learns the marginal feature distribution of the unlabeled samples using an additional unconditional discriminator.

5 APPLICATIONS

Due to the ubiquitous demand of machine learning techniques for large scale data sets and the rapid digital advances in recent years, the GZSL methods are now being applied to a variety of applications, particularly in computer vision and natural language processing (NLP). We discuss these applications in the following subsections, which are divided into image classification (the most popular application of GZSL), object detection, video processing, and NLP.

5.1 Computer Vision

In computer vision, GZSL is applied to solve problems related to both images and videos.

Image processing: Image classification is among the most popular applications of GZSL methods. The aim is to classify images from both seen and unseen classes. In this regard, a number of standard benchmark data sets have been introduced. Due to the page limit, a detailed review of these data sets is provided in the supplementary.

Object detection is another popular application in computer vision that has increasingly gained importance for large-scale tasks. It aims to locate objects, in addition to their
recognition. Over the years, many CNN-based models have been developed to recognize the seen classes. However, collecting sufficient annotated samples with ground-truth bounding-boxes is not scalable, and new methods have emerged. Currently, advances in zero-shot detection allow the conventional object detection models to detect classes that do not match previously learned ones. The reported methods in the literature include detecting general categories from single label [92], [183], [184], [185], [186], [187] to multi-label [188], [189], multi-view (CT and x-ray) [190] and tongue constitution recognition [191]. In addition, GZSL has been applied to segment both seen and unseen categories [192], [193], retrieve images from large scale data sets [194], [195] as well as perform image annotation [52].

Video processing: Recognizing human actions and gestures from videos is one of the most challenging tasks in computer vision, due to a variety of actions that are not available among the seen action categories. In this regard, GZSL-based frameworks are employed to recognize single label [25], [95], [166], [196] and multi-label [197] human actions. As an example, CLASTER [196] is a clustering-based method using reinforcement learning. In [197], a multi-label ZSL (MZSL) framework using JLR (Joint Latent Ranking Embedding) is proposed. The relation score of various action labels is measured for the test video clips in the semantic embedding and joint latent visual spaces. In addition, a multi-modal framework using audio, video, and text was introduced in [198], [199], [200].

5.2 Natural Language Processing

NLP [201], [202] and text analysis [203] are two important application areas of machine learning and deep learning methods. In this regards, GZSL-based frameworks have been developed for different NLP applications such as text classification with single label [104], [158], multi-label [114], [204] as well as noisy text description [53], [54]. Wang et al. [104] proposed a method based on the semantic embedding and categorical relationships, in which benefits of the knowledge graph is exploited to provide supervision in learning meaningful classifiers on top of the semantic embedding mechanism.

The application of GZSL on multi-label text classification plays a key role in generating latent features in text data. Song et al. [204] proposed a new GZSL model in a study on International Classification of Diseases (ICD) to identify classification codes for diagnosis of various diseases. The model improves the prediction on the unseen ICD codes without compromising its performance on seen ICD codes. On the other hand, for multi-label ZSL, Huynh and Elhamifar [115] trained all applied models on the seen labels and then tested on the unseen labels, whereas for multi-label GZSL, they tested all models on the both seen and unseen labels. In this regard, a new shared multi-attention mechanism with a novel loss function is devised to predict all labels of an image, which contain multiple unseen categories.

6 DISCUSSIONS AND CONCLUSIONS

In this section, we discuss the main findings of our review. Several research gaps that lead to future research directions are highlighted.

We categorize the GZSL methods into two groups: (i) embedding-based, and (ii) generative-based methods. Embedding-based methods learn an embedding space, whether visual-semantic, semantic-visual, common/latent space, or a combination of them (bidirectional), to link the visual space of the seen classes to their corresponding semantic space. They use the learned embedding space to recognize data samples from both seen and unseen classes. Various strategies have been developed to learn the embedding space. We categorize these strategies into graph-based, autoencoder-based, meta-learning-based, attention-based, compositional learning-based, bidirectional learning-based and other methods. In contrast, the generative-based methods convert GZSL into a conventional supervised learning problem by generating visual features for the unseen classes. Since the visual features of the unseen classes are not available during training, learning a projection function or a generative model using data samples from the seen classes cannot guarantee generalization pertaining to the unseen classes. Although the low-level attributes are useful for classification of the seen classes, it is challenging to extract useful information from structured data samples due to the difficulty in separating different classes. Thus, the main challenge of both methods is the lack of unseen visual samples for training, leading to the bias and projection domain shift problems. Table 2 in the supplementary summarizes the main properties of the embedding- and generative-based methods. In addition, Table 3 in the supplementary describes different embedding- and generative-based methods.

Embedding-based versus. Generative-based methods: Embedding-based models have been proposed to address the ZSL and GZSL problems. These methods are less complex, and they are easy to implement. However, their capabilities in transferring knowledge are restricted by semantic loss, while the lack of visual samples for the unseen classes causes the bias problem. This results in poor performances of the models under the GZSL setting. This can be seen in Table 4 in the supplementary, where the accuracy of seen classes (Acc_c) are higher than accuracy of unseen classes (Acc_u). In semantic embedding models, the projection of visual features to a low dimensional semantic space shrinks their variance and restricts their discriminability [65]. The compatibility scores are unbounded, and ranking may not be able to learn certain semantic structures because of the fixed margin. Moreover, they usually perform search in the shared space, which causes the hubness problem [50], [58], [62].

Although visual embedding models are able to alleviate the hubness problem [58], they suffer from several issues. First, the visual features and semantic representations are obtained independently, and they are heterogeneous, e.g., they are from different spaces. As such, the data distributions of both spaces can be different, in which two close categories in one space can be located far away in other space [42]. In addition, regression-based methods can not explicitly discover the intrinsic topological structure between two spaces. Therefore, learning a projection function directly from the space of one modality to the space of another modality may cause information loss, increased complexity, and overfitting toward the seen classes (see Table 4 in the supplementary). To mitigate this issue, several studies have attempted to project visual features and
semantic representations into a latent space. Such a projection can reconcile the structural differences between the two spaces. Besides that, bidirectional learning models aim to learn a better projection and adjust the seen-unseen class domains. However, these models still suffer from the projection domain shift and bias problems.

Generative-based methods are more effective in solving the bias problem, owing to synthesizing visual features for the unseen classes. This leads to a slightly balanced $Acc_s$ and $Acc_u$ performance, and consequently higher harmonic mean ($H$), as compared with those from embedding based methods (see Tables 4 and 5). However, the generative based methods are susceptible to the bias problem. While, the availability of visual samples for the unseen classes allows the models to perform recognition of both seen and unseen classes in a single process, they generate visual features through learning a model class conditioned on the seen classes. They do not learn a generic model for generalization toward both seen and unseen class generations [110]. They are complex in structure and difficult to train (owing to instability). Their performance is restricted either by obtaining the distribution of visual features using semantic representations or using the euclidean distance as the constraint to retain the information between the generated visual features and real semantic representations. In addition, the unconstrained generation of data samples for the unseen classes may produce samples far from their actual distribution. These methods have access to the semantic representations (under semantic transductive setting) and unlabeled unseen data (under transductive setting), which violate the GZSL setting.

**Inductive setting versus. Transductive setting:** As stated earlier, GZSL methods under the transductive setting know the distribution of the unseen classes, as they have access to the unlabeled samples of the unseen classes. Therefore, they can solve the bias and shift problems. This results in a balanced $Acc_s$ and $Acc_u$ performance with a high $H$ score as compared with those from GZSL methods under the inductive and semantic transductive settings (Table 6 *versus*. Tables 4 and 5).

### 6.1 Research Gaps

Despite considerable progresses in GZSL, methodologies, there are challenges pertaining to the unavailability of visual samples for the unseen classes. Based on our findings, the main research gaps that require further investigations are as follows:

**Methodology:** Although many studies to solve the projection domain shift problem are available in the literature, it remains a key challenging issue with respect to the existing models. Most of the existing GZSL methods are based on ideal data sets, which is unrealistic in real-world scenarios. In practice, the ideal setting is affected by uncertain disturbances, e.g. a few samples in each category are different from other samples. Therefore, developing robust GZSL models is crucial. To achieve this, new frameworks that incorporate domain classification without relying on latent space learning are required.

On the other hand, techniques that are capable of solving supervised classification problems can be adopted to solve the GZSL problem, e.g. ensemble models [205] and meta-learning strategy [110]. Ensemble models employ a number of individual classifiers to produce multiple predictions, and the final decision is reached by combining the predictions [206], [207], [208]. Recently, Felix et al. [73] introduced an ensemble of visual and semantic classifiers to explore the multi-modality aspect of GZSL. Meta-learning aims to improve the learning ability of the model based on the experience of several learning episodes [110], [120]. GZSL also can be combined with reinforcement learning [196], [209], [210] to better tackle new tasks. In addition, GZSL can be extended in several fronts, which include multi-modal learning [85], [198], [199], multi-label learning [189], multi-view learning [190], weakly supervised learning to progressively incorporate training instances from easy to hard [100], [129], [211], continual learning [109], long-tail learning [212], or online learning for few-shot learning where a small portion of labeled samples from some classes are available [67].

Separating the seen class domain from unseen class domain using outlier detection (or OoD) techniques is interesting, and these techniques have shown promising results in solving GZSL tasks. However, there are several potential issues. First, this approach requires training several models, i.e., an outlier detection method to separate seen classes from unseen ones, a supervised learning model to categorize seen class samples and a ZSL model to recognize unseen class samples. These requirements lead to computationally expensive and time-consuming issues. Second, separating the seen classes from unseen ones using a novelty detector is a difficult task. As the prior information of the unseen classes is usually unknown (e.g., inductive and semantic transductive settings) and there may be overlapping regions between the seen and unseen class samples, as well as difficulty in tuning model parameters, it is possible for the novelty detector to misclassify part of the data set. This issue is more challenging for fine-grained classes, where they contain discriminative information only in a few regions. Third, visual or semantic features may not be discriminative enough to separate seen classes from unseen ones. Thus, combining information of both visual and semantic spaces is imperative. These issues and the bias problem can result in incorrect final predictions. To alleviate these challenges, Dong et al. [213] categorized the test images into the compositional spaces, including source, target and uncertain spaces. The uncertain space contains test samples that cannot be confidently classified as one of the source and target spaces. Then, statistical methods are used to analyse the instance distribution and classify the ambiguous instances. However, this approach requires further investigation to study its effectiveness under various settings.

Recently, transformer-based language models have shown good results on various NLP tasks. As an example, generative pre-training (GPT) [214] is a semi-supervised method that combines unsupervised pre-training with supervised fine-tuning techniques. It uses a large corpus of unlabelled text along with a limited number of manually annotated samples in its operation. GPT-2 [215] and GPT-3 [216] are improved models of GPT, which are trained on large-scale data sets for solving tasks under ZSL and few-shot settings, respectively. Moreover, CLIP [217] and DALL-E [218] are useful transformer-based techniques for zero-shot text-to-image generation. Their ability in learning directly from raw texts (millions
of webpages) pertaining to images without any explicit supervision can be adopted for solving GZSL tasks. Unlike attributes defined by humans that contain limited information about the classes, transformers can be leveraged to identify more precise mapping functions between seen and unseen classes in the latent space. Their strong ability in learning from large scale training data sets constitutes a good direction for further research.

Data: Semantic representations play an important role in bridging the gap between the seen and unseen classes. The existing data sets use human-defined attributes or word vectors. The former shares the same attributes for different classes, and human labor is required for annotation, which is not suitable for large-scale data sets. The latter automatically extracts information from a large test corpus, which is usually noisy. Therefore, extracting useful knowledge from such data sets is difficult, especially fine-grained data sets [198], [199]. In this regard, using other modalities such as audio can improve the quality of the data samples. As such, it is crucial to focus on developing new techniques to automatically generate discriminative semantic attribute vectors, exploring other semantic spaces or combinations of various semantic embeddings that can accurately formulate the relationship between seen and unseen classes, and consequently solve the projection domain shift and bias problems. In addition, there are many unforeseen scenarios in real-world applications, such as driver-less cars or action recognition that may not be included among the seen classes. While GZSL can be applied to tackle such problems, no suitable data sets for such applications are available at the moment. This constitutes a key focus area for future research.

6.2 Concluding Remarks

Building models that can simultaneously perform recognition for both seen (source) and unseen (target) classes are vital for advancing intelligent data-based learning models. This paper, to the best of our knowledge, for the first time presents a comprehensive review of GZSL methods. Specifically, a hierarchical categorization of GZSL methods along with their representative models has been provided. In addition, evaluation protocols including benchmark problems and performance indicators, together with future research directions have been presented. This review aims to promote the perception related to GZSL.

REFERENCES


Xinlei Zhou received the BSc degree in computer science from the College of Computer Information Engineering, Jiangxi Normal University, China, in 2013. She is currently working toward the PhD degree with the College of Computer Science and Software Engineering, Shenzhen University, China. Her current research interests include the uncertainty information processing in machine learning, zero-shot learning, meta learning, and crowdsourcing learning.

Ran Wang (Member, IEEE) received the PhD degree from the Department of Computer Science, City University of Hong Kong, Hong Kong China, in 2014. From 2014 to 2016, she was a postdoctoral researcher with the Department of Computer Science, City University of Hong Kong. She is currently an Associate Professor with the College of Mathematics and Statistics, Shenzhen University, China. Her current research interests include pattern recognition, machine learning, fuzzy sets, and their applications.

Chee Peng Lim received the PhD degree in intelligent systems from the University of Sheffield, U.K., in 1997. His research interests include computational intelligence based systems for data analytics, condition monitoring, optimization, and decision support. He has published more than 350 technical papers in journals, conference proceedings, and books, received seven best paper awards, edited three books and 12 special issues in journals, and served in the editorial board of five international journals. He is currently a professor with Institute for Intelligent Systems Research and Innovation, Deakin University.

Xi-Zhao Wang (Fellow, IEEE) has been working as a professor with Big Data Institute, Shenzhen University, ShenZhen, China, since 2014. His main research interests include uncertainty modeling and machine learning for big data. He has edited more than 10 special issues and authored and co-authored three monographs, two text books, and more than 200 peer-reviewed research papers. He is the previous Board of Governors member of the IEEE SMC Society, the chair of the IEEE SMC Technical Committee on Computational Intelligence, the chief editor of Machine Learning and Cybernetics Journal, and an associate editors for a couple of journals in the related areas. He was the recipient of the IEEE SMCS Outstanding Contribution Award in 2004 and IEEE SMCS Best associate editor Award in 2006. He is the general co-chair of the 2002–2017 International Conferences on Machine Learning and Cybernetics, cosponsored by the IEEE SMCS.

Q. M. Jonathan Wu (Senior Member, IEEE) received the PhD degree in electrical engineering from the University of Wales, Swansea, U.K., in 1990. He was affiliated with the National Research Council of Canada for ten years beginning in 1995, where he became a senior research officer and a group leader. He is currently a professor with the Department of Electrical and Computer Engineering, University of Windsor, Windsor, ON, Canada. He has published more than 300 peer-reviewed papers in computer vision, image processing, intelligent systems, robotics, and integrated microsystems. His current research interests include 3-D computer vision, active video object tracking and extraction, interactive multimedia, sensor analysis and fusion, and visual sensor networks. He held the Tier 1 Canada research chair in Automotive Sensors and Information Systems from 2005 to 2019. He is an associate editor of the IEEE Transactions on Cybernetics, the IEEE Transactions on Circuits and Systems for Video Technology, the Journal of Cognitive Computation, and Neurocomputing. He has served on technical program committees and international advisory committees for many prestigious conferences. He is a fellow of the Canadian Academy of Engineering.

For more information on this or any other computing topic, please visit our Digital Library at www.computer.org/csdl.