Zero-shot Image Classification with Logic Adapter and Rule Prompt

Dongran Yu
Key Laboratory of Symbolic Computation and Knowledge Engineer, Ministry of Education, School of Artificial Intelligence, Jilin University
Changchun, China
yudran@foxmail.com

Xueyan Liu*
Key Laboratory of Symbolic Computation and Knowledge Engineer, Ministry of Education, School of Computer Science and Technology, Jilin University
Changchun, China
xueyanliu@jlu.edu.cn

Bo Yang*
Key Laboratory of Symbolic Computation and Knowledge Engineer, Ministry of Education, School of Computer Science and Technology, Jilin University
Changchun, China
ybo@jlu.edu.cn

ABSTRACT

Zero-shot image classification, which aims to predict unseen classes whose samples have never appeared during the training phase, is crucial in the Web domain because many new web images appear on various websites. Attributes, as annotations for class-level characteristics, are widely used semantic information for this task. However, most current methods often fail to capture discriminative image features between similar images from different classes, leading to unsatisfactory zero-shot image classification results. This is because they solely focus on limited visual-attribute feature alignment. Therefore, we propose a **Z**ero-**S**hot image **C**lassification with **L**ogic adapter and **R**ule prompt method called ZSCLR, which utilizes logic adapter and rule prompts to encourage the model to capture discriminative image features and achieve reasoning. Specifically, ZSCLR consists of a visual perception module and a logic adapter. The visual perception module extracts image features from training data. At the same time, the logic adapter utilizes the Markov logic network to encode the extracted image features and rule prompts for refining the discriminative image features. Due to predicates of rule prompts representing symbolic discriminative features, the proposed model can focus more on these discriminative features and achieve more precise image classification. Additionally, the logic adapter enables the model to adapt from recognizing images in seen classes to those in unseen classes through the reasoning of the Markov logic networks. We implement experiments on three standard zero-shot image classification benchmarks, and ZSCLR achieves competitive performance. Furthermore, ZSCLR can provide explanations for its predictions through rule prompts.

CCS CONCEPTS

• **Computing methodologies** → **Object identification**.

*Corresponding author

WWW '24, May 13–17, 2024, Singapore, Singapore

ACM ISBN 979-8-4007-0171-9/24/05…$15.00
https://doi.org/10.1145/3589334.3645554

KEYWORDS

Zero-shot Learning, Image Classification, Logic Adapter, Rule Prompt, Markov Logic Network

ACM Reference Format:
Dongran Yu, Xueyan Liu, and Bo Yang. 2024. Zero-shot Image Classification with Logic Adapter and Rule Prompt. In *Proceedings of the ACM Web Conference 2024 (WWW '24), May 13–17, 2024, Singapore, Singapore.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3589334.3645554

1 INTRODUCTION

Relying on massive labeled training data to classify images has led to significant progress in the computer vision domain in recent years [23]. However, annotating all objects' classes is unrealistic, especially those that are expensive to label or not efficiently collectible in real-world applications. This limitation poses challenges for supervised learning methods when classifying objects that were unseen during training. To address this issue, zero-shot learning (ZSL) has emerged as a promising approach [15, 36]. ZSL requires labeled objects from seen classes but can effectively recognize objects from unseen classes. Applying ZSL to the computer vision domain is crucial because a vast number of new web images are created daily on social media and other websites.

In this paper, we focus on zero-shot image classification (ZSC), which involves training the model on images of seen categories and recognizing images of unseen categories. Most methods map images from the visual space into the semantic space and classify them [12, 14, 20, 31]. However, these methods may lead to misclassification due to the similarity between image features. For instance, as shown in Figure 1 (a) left, images of instance "*clay_colored_sparrow*" and "*brewer_sparrow*" are mapped into the semantic space, where the image feature of "*brewer_sparrow*" falls into the "*clay_colored_sparrow*" category space since they are visually similar. Based on this situation, we observe that objects with similar appearances exhibit similar image features in the semantic space, producing a wrong classification. To avoid such misclassifications, the model needs to capture discriminative image features, such as vital attributes that distinguish differences. For example, the discriminative features of "*clay_colored_sparrow*" and "*brewer_sparrow*" are the shape of the bill, e.g., "*cone_bill*" and "*dagger_bill*", and the color of the wing, e.g., "*grey_wing*" and "*brown_wing*". If informing the model about these discriminative features via rule prompts such as $\texttt{cone\_bill}(x) \wedge \texttt{grey\_wing}(x) \Rightarrow \texttt{clay\_colored\_sparrow}(x)$ and $\texttt{dagger\_bill}(x) \wedge \texttt{brown\_wing}(x)$

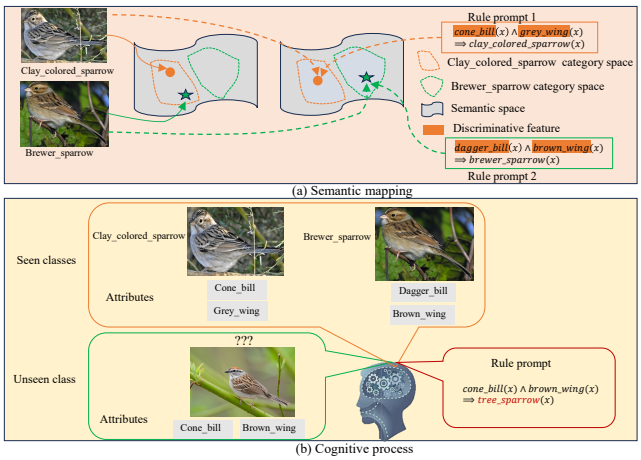

Figure 1: The illustrative diagrams. (a) Due to the object's visual features being similar, the *Brewer_sparrow* is misclassified as a *Clay_colored_sparrow* in semantic space. It is correctly recognized using discriminative features in the rule prompts by our proposed ZSCLR. (b) The cognitive process of the human for recognizing objects unseen previously.

$\Rightarrow$ brewer_sparrow(x), images can be better classified in the semantic space, as depicted in Figure 1 (a) right. Therefore, the proper discovery of discriminative image features in semantic spaces for ZSC is of great importance.

Recently, several methods have emerged for learning discriminative image features in ZSC. Researchers have focused more on attention networks, leading to the development of attention-based ZSC methods [3, 14, 37]. These methods leverage attribute descriptions, e.g., word vectors, as auxiliary information to discover discriminative image features, facilitating accurate alignment with semantic representations. Although these efforts improve the classification accuracy in ZSC, the results are still unsatisfactory, particularly when handling datasets containing very visually similar images belonging to different classes. Because these approaches primarily rely on attention networks, which focus on limited semantic alignments between visual and attribute features. Logic rules, which condense human intelligence and knowledge, can be utilized to guide extracting discriminative image features. However, due to the challenge of integrating two distinct representations: logic rules in symbolic form and image representations in vector/matrix form, there is currently no existing method to apply them to ZSC tasks in terms of image classification from unseen classes. Therefore, in this paper, we seek to address the following question: *How do we properly use logic rules to capture discriminative image features for improving zero-shot image classification accuracy?*

To address the above issue, inspired by the human cognitive process, we will introduce a logic adapter for integrating logic rules to ZSC. Humans can comprehend unseen objects through reasoning based on prior experiences, even without prior exposure to them. For example, in Figure 1 (b), seen classes are "*clay_colored_sparrow*" with attribute features "*cone_bill*", "*grey_wing*" and "*brewer_sparrow*" with "*dagger_bill*", "*brown_wing*", while unseen class is "*tree_sparrow*". When humans encounter a new image of a "*tree_sparrow*" for the first time, they rely on prior knowledge from having seen images of "*clay_colored_sparrow*" and "*brewer_sparrow*". By comparing the new image's visual attributes, such as "*cone_bill*" and "*brown_wing*", to what they have learned from seen classes, humans can select the proper rule prompt, i.e., cone_bill(x)$\wedge$brown_wing(x) $\Rightarrow$ tree_sparrow(x), from the knowledge base containing various rule prompts. Thus, they can logically deduce that the new image from the unseen class is indeed "*tree_sparrow*".

Based on the above analysis, we have proposed a method called ZSCLR (**Z**ero-**S**hot image **C**lassification with **L**ogic adapter and **R**ule prompt), which integrates rule prompts into ZSC models to capture discriminative image features. ZSCLR comprises two key components: a visual perception module and a logic adapter. The visual perception module, designed using CNN and attention network, primarily focuses on image feature extraction. Meanwhile, the logic adapter takes image features from the visual perception module and encodes them and rule prompts that predicates represent symbolic discriminative features via Markov logic network (MLN)[17]. More concretely, in the logic adapter, we introduce a statistical relation learning model, i.e.,MLN. MLN can combine statistical models (e.g., ZSC models) and relational models (e.g., rule prompts) to attain a unified representation, such as a joint probability distribution, and achieve reasoning via computing posterior. Moreover, rule prompts with symbolic discriminative features can tell the model which features are discriminative in an image, such as the shape of the bill, and then the model can focus more on what during training. In this paper, rule prompts are formalized through first-order logics (FOLs), such as dagger_bill(x)$\wedge$brown_wing(x) $\Rightarrow$ brewer_sparrow(x), encoding them within the MLN. FOLs serve a dual purpose: they represent logical relationships between attribute features and classes and provide a powerful means of expressing symbolic knowledge. We use a variational Expectation-Maximization (EM) algorithm to train the model in an end-to-end way and utilize the logic adapter to reasoning results during the test. Additionally, the logic adapter offers a reasoning process through FOLs, which enables the model to adapt from seen classes to recognize unseen classes and explain why a particular decision was made. Finally, we present the results of our experiments conducted on the AwA2 [26], CUB [25] and SUN [16] datasets to evaluate the performance and interpretability of ZSCLR. These results vividly illustrate the remarkable superiority and promising potential of ZSCLR in zero-shot image classification.

In summary, our contributions can be summarized as follows:

- To the best of our knowledge, ZSCLR is the first to integrate logic rules into zero-shot image classification and is a novel paradigm for zero-shot image classification. It includes a visual perception module and a logic adapter.
- The visual perception module extracts image features via attributes guided, while the logic adapter enhances this process by utilizing rule prompts to refine discriminative image features. Furthermore, the logic adapter employs a Markov logic network to integrate rule prompts and the visual perception module and achieve reasoning. Importantly, our approach enables end-to-end training within a flexible variational EM framework. It not only enhances model performance for unseen classes but also provides interpretability for predictions, offering insights into the underlying reasoning process.

- Based on our extensive experimental results, we attain the superior performance of ZSCLR compared to the state-of-the-art methods. Furthermore, we illustrate the interpretability of ZSCLR by offering visualizations that significantly enhance the comprehension of the underlying reasoning process.

Table 1: Important notations and their descriptions.

Notations	Descriptions
$\mathcal{X}^s(\mathcal{X}^u), x^s(x^u)$	seen (unseen) image
$\mathcal{Y}^s(\mathcal{Y}^u), y^s(y^u)$	seen (unseen) class label
$\boldsymbol{a}_y^s(\boldsymbol{a}_y^u), \mathbf{A}^s(\boldsymbol{A}^u)$	seen (unseen) class attribute
$\boldsymbol{V}, \boldsymbol{V}^{'}$	image feature
$\boldsymbol{A_{tt}}$	attribute feature
S	the number of seen classes
$\boldsymbol{\alpha}$	attribute weight matrix of image features
$\boldsymbol{\phi}$	prediction score
R^s, R^u, R, r	logic rule (FOL)
a_r	ground atom in a logic rule
$\mathcal{A}_r, \mathcal{A}$	ground atom set(s)
$\mathcal{A}_O, \mathcal{A}_H$	observed and unobserved ground atom set(s)
φ	potential function
w	weight sets of the logic rules
w_r	weight of a logic rule

2 RELATED WORK

Zero-shot learning aims to train a model to recognize classes not included in its original training. There are many works to study zero-shot learning in image classification. These works can be classified into three categories: embedding-based approaches, generative-based approaches and knowledge-based approaches. Embedding-based approaches aim to learn a mapping function for visual-semantic interaction. They determine the label of a sample by matching their vectors in the same space using similarity metrics. Some approaches are implemented by mapping the visual features to semantic space by [18, 27, 29]. In contrast, other methods propose mapping the semantic features into visual space and point out that using semantic space as shared latent space may reduce the variance of features [35]. As the embedding is learned only on seen classes, these embedding-based methods inevitably overfit to seen classes. To address this problem, generative-based methods have been introduced utilizing generative models such as VAEs, GANs to learn semantic-visual mapping to generate visual features of unseen classes, which can offset the shortage of unseen classes and convert ZSL into a supervised classification task. This category focuses on learning a class condition generator to generate features of unseen classes [1, 33] or using semantic information (e.g., attributions) of class to generate features of unseen classes directly [5, 18, 22]. However, these methods still usually yield relatively undesirable results since they cannot capture the subtle differences between seen and unseen classes. Knowledge-based methods have been explored to capture more correlations between seen and unseen classes. Most approaches utilize knowledge graphs (e.g., class hierarchies and commonsense knowledge) as side information. Specifically, they are used to build relationship graphs between seen and unseen classes as a semantics-level graph to learn recognizing unseen classes [2, 10, 11, 24, 30].

Besides the three mentioned categories, the zero-shot image classification methods can also be divided into discriminative and non-discriminative methods based on whether they consider the importance of different features. Since the above-mentioned approaches fail to account for discriminative features, they are attributed to non-discriminative methods. Recently, some discriminative methods have begun to explore the discriminative image features using attention networks [3, 4, 6, 28, 37]. Xie et al. [28] constructs a region graph using parts of the object via the attention technique and achieves transferring knowledge between different classes. Chen et al. [3, 4] utilize mutually visual-attribute attention sub-net for semantic distillation, encouraging the model to explore the discriminative features for image. To solve the attribute imbalance and co-occurrence, Chen et al. [6] introduces an attribute-level contrastive learning mechanism.

To some extent, our model is a discriminative approach. In contrast to existing approaches, we incorporate logic rules as auxiliary information to capture discriminative image features and logical relationships between discriminative features and classes, and employ the Markov logic network for prediction. Moreover, our model introduces rule prompts with symbolic discriminative features, providing interpretability compared to those who use attention networks to capture discriminative image features.

3 METHODOLOGY

Problem Setting. Zero-shot classification (ZSC) aims to recognize unseen classes by transferring knowledge from the seen domain $\mathcal{D}^s$ to the unseen domain $\mathcal{D}^u$. The training data for seen classes is denoted as $\mathcal{D}^s = \{(x^s, y^s, \boldsymbol{A}^s) | x^s \in \mathcal{X}^s, y^s \in \mathcal{Y}^s\}$, where $\mathcal{X}^s$ represents the image sets with class labels from $\mathcal{Y}^s$, and $\boldsymbol{A}^s \in \mathbb{R}^{S \times m}$ represents the class attributes of the seen classes. Similarly, $\mathcal{D}^u = \{(x^u, y^u, \boldsymbol{A}^u) | x^u \in \mathcal{X}^u, y^u \in \mathcal{Y}^u\}$ is data of the unseen classes. Additionally, $\mathcal{X} = \mathcal{X}^s \cup \mathcal{X}^u$. In ZSC, the class space is disjoint between the seen and unseen domains, i.e., $\mathcal{Y}^s \cap \mathcal{Y}^u = \varnothing$. The model is trained on the seen classes $\mathcal{Y}^s$ but is tested on the unseen classes $\mathcal{Y}^u$. To bridge the gap between seen and unseen categories, auxiliary information such as attribute descriptions $\boldsymbol{A}^s$ and $\boldsymbol{A}^u$ is required. To aid in understanding the paper, important notations and their descriptions have been provided in Table 1.

Overview. In Figure 2, ZSCLR consists of a visual perception module and a logic adapter. The visual perception module can learn a mapping function from visual space to semantic space to extract image features $\boldsymbol{V}^{'}$. In this paper, to train the visual perception module, we compute the inner product of both extracted image features and class attributes to attain a score of classification. To make extracted image features more discriminative, we design a logic adapter. The logic adapter initially receives image features from the visual perception module. It then models rule prompts through Markov logic networks to learn a joint probability distribution. In logic adapter,

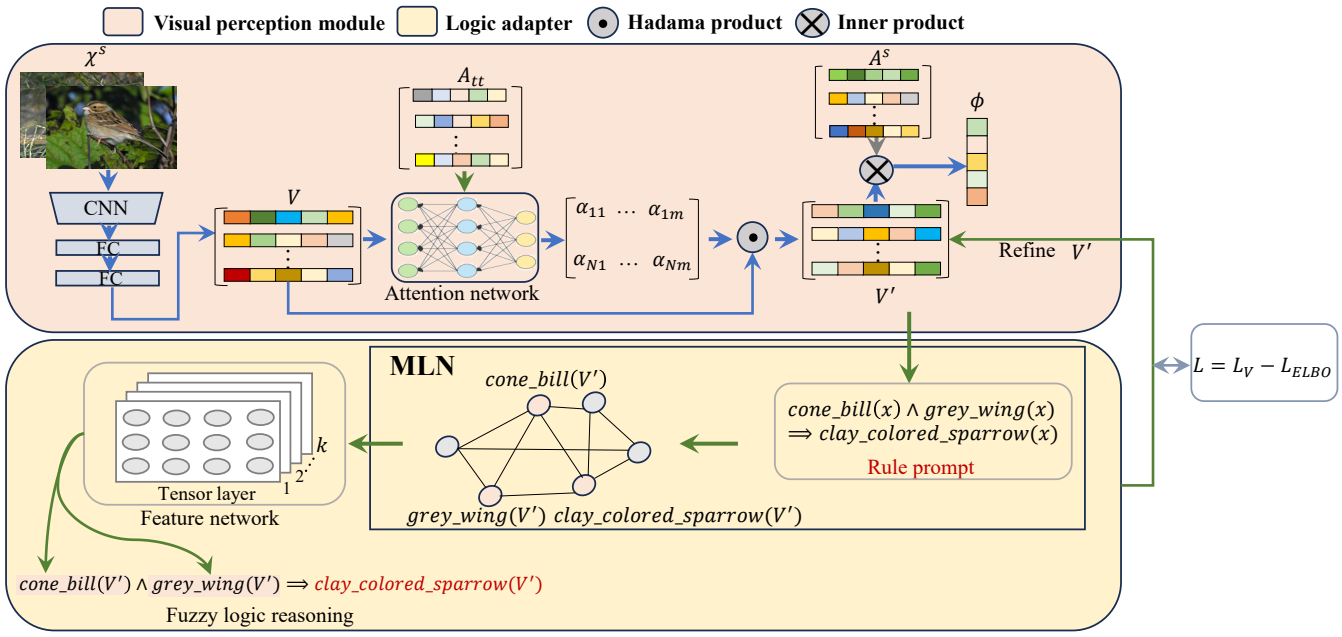

Figure 2: Overview of ZSCLR: ZSCLR comprises a visual perception module and a logic adapter. The visual perception module extracts image features using a CNN, fully connected layers (FCs), and an attention network guided by attribute features. The learned image features are then input into the logic adapter, which employs a Markov logic network to learn a joint probability distribution capturing shared variables between seen and unseen classes. This facilitates effective knowledge transfer in zero-shot image classification, allowing inference of attribute feature labels and class labels via feature network and fuzzy logic.

we compute posterior via feature network to predict attribute labels (e.g., $clon\_bill(V^{'})$), and combine these attribute labels and rule prompts to infer class labels (e.g., $clay\_colored\_sparrow(V^{'})$) through fuzzy logic reasoning. During training, this process can refine image features $V^{'}$ via backpropagation to attain discriminative image features. In this process, the logic adapter serves a dual purpose: aiding the model in acquiring discriminative image features and facilitating adaptation to new environments, i.e., recognizing unseen classes. In our ZSCLR, the input includes seen class images $\mathcal{X}^s$, corresponding attribute features$\boldsymbol{a}_y^s$, class attribute labels $\boldsymbol{A}^s$ and rule prompts R^s of seen classes during training. In the testing phase, we input unseen class images $\mathcal{X}^u$, their corresponding attribute feature vectors $\boldsymbol{a}_y^u$, and the knowledge base R^u containing the rule prompts for unseen classes.

3.1 Visual Perception Module

In the Visual Perception Module (VPM), we aim to map the visual features in visual space to the corresponding attribute features in the semantic space. Specifically, the VPM requires two inputs for obtaining image features $V^{'}$: a set of images $\mathcal{X} = \{x_1, ..., x_N\}$, where N is the number of images, and matrix of the attribute features $\boldsymbol{A}_{tt} \in \mathbb{R}^{N\times m}$, where m is dimension of attribute feature. The primitive features, represented by $\boldsymbol{V} \in \mathbb{R}^{N\times m}$, are obtained by feeding these into a CNN, and then the fully connected layers with ReLU activation. As previous work [13] claims that learning useful visual features is crucial in ZSC, we coarsely screen out good image features, denoted as $\boldsymbol{V}^{'} \in \mathbb{R}^{N\times m}$, using the attention network guided by attribute features from a given image. These image features are prepared for the logic adapter to refine the discriminative image features. Specifically, we need to calculate the attention for each feature by

$$\boldsymbol{\alpha} = \text{softmax}(\boldsymbol{A}_{tt}\boldsymbol{W}_a\boldsymbol{V}), \tag{1}$$

where $\boldsymbol{\alpha} \in \mathbb{R}^{N\times m}$ represents attribute weight matrix of image features, and $\boldsymbol{W}_a \in \mathbb{R}^{m\times N}$ is a learnable matrix.

Next, primitive image features $\boldsymbol{V}$ and $\boldsymbol{\alpha}$ calculate Hadama product to attain $\boldsymbol{V}^{'}$ by

$$\boldsymbol{V}^{'} = \boldsymbol{\alpha} \odot \boldsymbol{V}, \tag{2}$$

where $\odot$ is the Hadama product.

Then, $\boldsymbol{V}^{'}$ are refined and updated by the logic adapter in Section 3.2. Finally, we calculate similar scores for both image features and class attribute labels to attain image class labels. Note, this score calculation is performed solely during training to generate the loss Eq. (6) in VPM, thereby facilitating model training. More concretely, image feature $\boldsymbol{V}^{'}$ match its corresponding seen class attribute labels $\boldsymbol{A}^s$, which is formulated as compute inner product:

$$\boldsymbol{\phi} = \boldsymbol{V}^{'}\boldsymbol{A}^{s\top}, \tag{3}$$

where $\boldsymbol{\phi} \in \mathbb{R}^{N\times S}$ is the score that represents the distribution of classes for the images.

3.2 Logic Adapter

As the core of ZSCLR, the logic adapter (LA) integrates the VPM with rule prompts to extract discriminative image features. Meanwhile, LA can adapt the trained model from seen to unseen classes, and enables reasoning through Markov logic networks. To achieve this integration, it's crucial to establish a unified representation between the VPM and rule prompts. Thus, we introduce Markov logic networks (MLN), a framework that seamlessly combines statistical models (VPM) with relational models (rule prompt) into a unified representation [7]. For this purpose, we use FOLs like $\texttt{dagger\_bill}(x) \wedge \texttt{brown\_wing}(x) \Rightarrow \texttt{brewer\_sparrow}(x)$ as rule prompts. These FOLs establish logical relationships between symbolic discriminative features (attributes) and classes. Specifically, LA uses MLN to learn a joint probability distribution for symbolic discriminative features, enabling the prediction of discriminative feature labels by calculating posterior probabilities. The class labels are then inferred using fuzzy logic based on the predicted discriminative feature labels.

In fact, MLN can be thought of as a knowledge base utilizing FOLs. In MLN, we assume there is a FOL set R for a given dataset, and an FOL $r \in R$ comprises a set of predicate functions, like $\{p_r^1(x), p_r^2(x), ...\}$, where the domain of x is the set of constants $C = \{c_1, c_2, ...\}$, c_i denotes object i. When the constants are assigned to the arguments of a predicate, these assigned predicates are called ground atoms, and each FOL corresponds to a ground atom set $\mathcal{A}_r = \{a_r^1, a_r^2, ...\}$. Using the examples from Figure 1 (a), the constants c_1 and c_2 are the image features $V_1^{'}$ and $V_2^{'}$ obtained by the VPM, the ground atom sets are $\mathcal{A}_1 = \{\texttt{cone\_bill}(V_1^{'}), \texttt{grey\_wing}(V_1^{'}), \texttt{clay\_colored\_sparrow}(V_1^{'}), \texttt{cone\_bill}(V_2^{'}), \texttt{grey\_wing}(V_2^{'}), \texttt{clay\_colored\_sparrow}(V_2^{'})\}$, and $\mathcal{A}_2 = \{\texttt{dagger\_bill}(V_1^{'}), \texttt{brown\_wing}(V_1^{'}), \texttt{brewer\_sparrow}(V_1^{'}), \texttt{dagger\_bill}(V_2^{'}), \texttt{brown\_wing}(V_2^{'}), \texttt{brewer\_sparrow}(V_2^{'})\}$ for the two FOLs, i.e., the rule prompts.

MLN models FOLs as a probabilistic graph, where nodes represent ground atoms, and edges between nodes correspond to logical relationships between these ground atoms. For the FOL set R and its ground atoms, the joint distribution defined by MLN can be represented as follows:

$$P_w = \frac{1}{Z(w)} \exp\{\sum_{r \in R} w_r \sum_{a_r \in \mathcal{A}_r} \varphi(a_r)\}, \tag{4}$$

where $Z(w)$ is the partition function summing overall ground atoms. w represents the weight sets of all FOLs, and w_r is the corresponding weight of each FOL. φ denotes a potential function in terms of the number of times the FOL is true.

3.3 Model Optimization

We consider both the VPM and LA in our model and optimize them end-to-end. For the VPM, we follow the same way as in LFGAA [13] that uses triplet loss [19] to learn image features by simultaneously enlarging interclass distance and reducing intra-class distance:

$$L_T = \frac{1}{N} \sum_{i=1}^{N} [||V_i^{'} - V_j^{'}||^2 - ||V_i^{'} - V_m^{'}||^2 + \beta]_+, \tag{5}$$

where $V_i^{'}$, $V_j^{'}$ and $V_m^{'}$ serve as anchor, positive and negative sample within a triplet respectively. $[\circ]_+$ is equivalent to $max(\circ, 0)$. β is a hyperparameter controlling the desired margin between the positive and negative pairs.

Furthermore, to learn a mapping function from visual features to semantic space, we design the following loss:

$$L_S = -\frac{1}{N} \sum_{i=1}^{N} \log \frac{\exp(\boldsymbol{\phi}_i)}{\sum_S \exp(\boldsymbol{\phi}_i)}, \tag{6}$$

where $\boldsymbol{\phi}_i$ represents class label score of each image.

Therefore, the VPM's final loss can be written as follows:

$$L_V = L_T + L_S. \tag{7}$$

Logic adapter is a MLN, we need to maximize the log-likelihood $\log P_w(\mathcal{A}_O)$ of all the observed predicates (variables) $\mathcal{A}_O$. However, it is intractable to maximize the overall objective directly since it requires computing the whole partition function $Z(w)$ and integrating over all predicates. Therefore, as suggested in [32], the approach is to optimize the variational evidence lower bound (ELBO) of the data log-likelihood, which is as follows:

$$L_{ELBO} = E_Q[\log P_w] - E_Q[\log Q(\mathcal{A}_\mathcal{H}|\mathcal{A}_O)], \tag{8}$$

where $\mathcal{A}_O$ is observed ground atom sets, while $\mathcal{A}_\mathcal{H}$ is unobserved ground atom sets. $Q(\mathcal{A}_\mathcal{H}|\mathcal{A}_O)$ is the variational posterior.

We use the variational EM algorithm to optimize L_{ELBO}, that is to minimize KL divergence between the $Q(\mathcal{A}_\mathcal{H}|\mathcal{A}_O)$ and $P_w(\mathcal{A}_\mathcal{H}|\mathcal{A}_O)$ to implement inference during E-step. In the M-step, the algorithm uses gradient descent to learn the weight of the FOL. Due to the inference of the MLN is #P-complete [17], we need to use mean-field theory to factorize variational distribution $Q(\mathcal{A}_\mathcal{H}|\mathcal{A}_O)$:

$$Q(\mathcal{A}_\mathcal{H}|\mathcal{A}_O) = \prod_{\mathcal{A}_{\mathcal{H}i} \in \mathcal{A}_\mathcal{H}} Q(\mathcal{A}_{\mathcal{H}i}). \tag{9}$$

To improve inference effectiveness, we use neural networks (feature networks in this paper) to parameterize variational calculation in Eq. (9). Consequently, this variational operation becomes a process of learning parameters. In this paper, we use a tensor network as our feature network to model $Q(\mathcal{A}_\mathcal{H}|\mathcal{A}_O)$ as $Q_\theta(\mathcal{A}_\mathcal{H}|\mathcal{A}_O)$ in Figure 3, and θ is the parameter of the feature network.

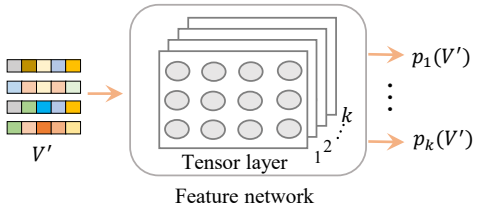

Figure 3: Feature network. The inputs are image features of the object, e.g., $V^{'}$, and the outputs are probabilities of the attribute labels of the object, e.g., $p_k(V^{'})$. k represents the tensor layer, and each layer is a predicate.

Thus, according to Eq. (4), Eq. (9) and feature network, the Eq. (8) can be rewritten as follows:

$$L_{ELBO} = \sum_{r \in R} w_r \sum_{a_r \in \mathcal{A}_r} E_{Q_\theta}[\varphi(a_r)] - \log Z(w) - \sum_{\mathcal{A}_{\mathcal{H}_i} \in \mathcal{A}_{\mathcal{H}}} E_{Q_\theta}[\log Q_\theta(\mathcal{A}_{\mathcal{H}} | \mathcal{A}_O)]. \tag{10}$$

Finally, we combine optimization targets of the visual perception module and logic adapter to minimize final loss L:

$$L = L_V - L_{ELBO}. \tag{11}$$

In the M-step, the model needs to learn the weights of the FOLs. The partition function $Z(w)$ has an exponential number of terms, which makes it intractable to optimize ELBO directly. To address this issue, we use pseudo-likelihood [17] to optimize L_{ELBO} in M-step, as the following:

$$P^*_w := E_{Q_\theta}[\sum_{r, \mathcal{A}_i \in \mathcal{A}_r} \log P_w(\mathcal{A}_i | MB_{\mathcal{A}_i})], \tag{12}$$

where $MB_{\mathcal{A}_i}$ is Markov blanket of the ground atom $\mathcal{A}_i$. For each rule r that connects $\mathcal{A}_i$ to its Markov blanket, we optimize the weights w_r by gradient descent, the derivative is following:

$$\nabla_{w_r} E_{Q_\theta}[\log P_w(\mathcal{A}_i | MB_{\mathcal{A}_i})] \simeq Y_{\mathcal{A}_i} - P_w(\mathcal{A}_i | MB_{\mathcal{A}_i}), \tag{13}$$

where $Y_{\mathcal{A}_i} = 0$ or 1 if $\mathcal{A}_i$ is an observed variable, and $Y_{\mathcal{A}_i} = Q_\theta(\mathcal{A}_i)$ otherwise.

3.4 Inference

Prediction. After training ZSCLR, the model recognizes unseen images using the logic adapter. Specifically, the model uses the visual perception module to extract the discriminative image features, denoted as $\boldsymbol{V}'$. These discriminative image features are then input into the logic adapter. In the logic adapter, the feature network is used to calculate the posterior probability $Q_\theta(\boldsymbol{V}')$, attaining the attribute feature labels of the instances. Then, the model leverages rule prompts from unseen classes to combine recognized attribute feature labels to infer class labels via fuzzy logic reasoning.

Interpretability. Our ZSCLR offers interpretability for predictions through corresponding rule prompts. That is to say, the model not only can predict class labels of objects in the images but also tell reasons for classifying the images to the class labels through symbolic language. In this study, we employ key attribute characteristics as our discriminative image features. To enhance understanding, we represent these attribute features using symbolic logic rules, replacing them in vector form. For example, when the model recognizes an unseen class image, it can identify the key attribute characteristics present in the sample and subsequently classify the unseen class object based on both these key attribute characteristics and the associated rule prompts. In this process, the key attribute characteristics and their corresponding rule prompts serve as explanations for the predictions.

4 EXPERIMENT

4.1 Experimental Setup

Datasets. We employed three challenging benchmark datasets, AwA2 (Animals with Attributes 2) [26], CUB (Caltech UCSD Birds 200-2011) [25] and SUN (SUN Attribute) [16] to validate our method. These datasets offer varying degrees of granularity, with AwA2 and SUN being coarse-grained datasets and CUB being fine-grained dataset. AwA2 consists of 37,322 images distributed across 50 animal categories, each associated with 85 attributes. Within this dataset, 40 categories are considered seen during training, while the remaining 10 are unseen during training and are used for evaluation. CUB comprises 11,788 images spanning 200 bird classes, each associated with 312 attributes. Among these classes, 150 classes are designated as seen during training, while the remaining 50 are unseen and used for evaluation. SUN dataset consists of 717 categories, comprising 14,340 images, with each category associated with 102 attributes. Among these, 645 categories are designated for the training set, while the remaining 72 categories form the test set. Moreover, to enable logic-based reasoning, we constructed FOLs based on attributes, taking the form of $\texttt{attribute}_1(x) \wedge \texttt{attribute}_2(x) \wedge \cdots \wedge \texttt{attribute}_n(x) \Rightarrow \texttt{class}(x)$. In these FOLs, the rule body consists of attributes, while the rule head represents class labels. In this paper, to divide the classes into seen and unseen classes, we adopted the Proposed Split (PS) method [26]. The statistics of the datasets are shown in Table 2.

Table 2: Dataset statistics.

Datasets	Attributes	$\lvert\mathcal{Y}^s\rvert$	$\lvert\mathcal{Y}^u\rvert$	$\lvert\mathcal{X}^s\rvert$	$\lvert\mathcal{X}^u\rvert$	$\lvert R\rvert$
AwA2	85	40	10	30,414	6,908	50
CUB	312	150	50	7,057	4,731	200
SUN	102	645	72	10,320	4,020	717

Evaluation Protocols. We followed the evaluation protocol outlined in [26] and assessed the top-1 accuracy in two distinct settings. First, we conducted experiments within the conventional zero-shot learning (CZSL) framework, where exclusively unseen categories were involved in testing. Consequently, all prediction results were restricted to be drawn from unseen classes, and this accuracy is denoted as Acc. Secondly, we ventured into the realm of generalized zero-shot learning (GZSL), where test images encompassed both seen and unseen categories. In this setting, we computed the top-1 accuracy for both seen (S) and unseen (U) categories independently. Furthermore, to measure the trade-off between performance on seen and unseen categories, we calculated the harmonic mean (H) by using the formula $\mathsf{H} = 2 \times \frac{\mathsf{S} \times \mathsf{U}}{\mathsf{S} + \mathsf{U}}$.

Implementation Details. We employed CNN as our basic image features extractor in the visual perception module. Indeed, our CNN can be initialized by a pre-trained backbone such as ResNet101 or GoogleNet [21]. Before being fed into the model, the images need to be randomly cropped for data augmentation. For optimization, we used the Adam optimizer with specific configurations. On the AwA2 dataset, we set the number of epochs to 15, the batch size to 64, and the learning rate to 1e-4. On the CUB dataset, we conducted training for 20 epochs, with a batch size of 32 and a learning rate of 1e-5. For SUN dataset, we set the number of epochs to 25, the batch size to 64, and the learning rate to 1e-4.

Baselines. We compare our ZSCLR with representative approaches proposed in recent years. These approaches are divided into two classes: Non-Discriminative methods include E-PGN [33], Composer [8], GCM-CF [34], FREE [5], LFGAA [13], DAZLE [9], APN [29], CF-ZSL[30]; Discriminative methods include MSDN [4], TransZero [3], DUET[6].

Table 3: Results (%) of our method and compared baselines. The best results in baselines are marked in bold. "-" is not reported in their paper.

Classes	Methods	AWA2				CUB				SUN			
		CZSL	GZSL			CZSL	GZSL			CZSL	GZSL		
		Acc	U	S	H	Acc	U	S	H	Acc	U	S	H
Non-Discriminative	E-PGN [33]	73.4	52.6	83.5	64.6	72.4	52.0	61.1	56.2	-	-	-	-
	Composer [8]	71.5	62.1	77.3	68.8	69.4	56.4	63.8	59.9	62.6	55.1	22.0	31.4
	GCM-CF [34]	-	60.4	75.1	67.0	-	61.0	59.7	60.3	-	47.9	37.8	42.2
	FREE [5]	-	60.4	75.4	67.1	-	55.7	59.9	57.7	-	47.4	37.2	41.7
	LFGAA [13]	68.1	27.0	**93.4**	41.9	67.6	36.2	**80.9**	50.0	61.5	18.5	40.0	25.3
	DAZLE [9]	67.9	60.3	75.7	67.1	66.0	56.7	59.6	58.1	59.4	52.3	24.3	33.2
	APN [29]	68.4	57.1	72.4	63.9	72.0	65.3	69.3	67.2	61.6	41.9	34.0	37.6
	CF-ZSL[30]	69.2	33.3	82.0	47.4	66.2	36.3	72.9	48.5	-	29.4	**45.8**	35.8
Discriminative	MSDN [4]	70.1	62.0	74.5	67.7	76.1	68.7	67.5	68.1	65.8	52.2	34.2	41.3
	TransZero [3]	70.1	61.3	82.3	70.2	76.8	69.3	68.3	68.8	65.6	52.6	33.4	40.8
	DUET[6]	69.9	63.7	84.7	72.7	72.3	62.9	72.8	67.5	64.4	45.7	**45.8**	45.8
	ZSCLR (our)	**76.0**	**70.4**	76.7	**73.4**	**77.8**	**70.5**	74.3	**72.4**	**66.3**	**52.8**	45.3	**48.7**

4.2 Comparision with State-of-the-Arts

Conventional Zero-Shot Learning. In Table 3, we present the results of Non-Discriminative methods and Discriminative methods in both CZSL and GZSL. In the CZSL scenario, on the one hand, we find that Discriminative methods exhibit certain advantages compared to most Non-Discriminative methods. This advantage stems from the discriminative features carrying shared information between seen and unseen classes, such as "having wings", which enhances the model's ability to transfer from recognizing seen to unseen classes. On the other hand, we observe that ZSCLR performs optimally across different datasets, validating the effectiveness of the model. We attribute this to ZSCLR's capability not only to capture discriminative features but also to leverage rule prompts for capturing relationships between discriminative features, seen classes, and unseen classes. This includes co-occurrence relationships, thereby assisting model in recognizing unseen classes.

Generalized Zero-Shot Learning. In the GZSL scenario, our ZSCLR shows relatively poor performance in S compared to several other methods. This is because our model focuses on recognizing unseen classes in CZSL scenario. Therefore, the logic adapter only includes rule prompts of unseen classes in the inference stage (Section 3.4). Due to the lack of rule prompts of seen classes during inference, seen classes are predicted only through the visual perception module, which may lose some useful information. However, our ZSCLR performs the best in both H and U. In the GZSL scenario, H is an important metric for evaluating the model's performance. If H performs well, it indicates that the model has the ability to balance seen and unseen classes.

Table 4: Ablation studies for different compositions of ZSCLR. The *Att Net* is the attention network.

Methods	AwA2	CUB
VPM w/o *Att Net*	62.2	51.0
VPM w/ *Att Net*	68.4	67.6
VPM w/o *Att Net*, w/ LA	70.3	70.5
ZSCLR	76.0	77.8

4.3 Ablation Studies

To evaluate the impact of different modules of the model on the results, we conducted an ablation experiment on the AwA2 and CUB datasets. In Table 4, we first observe a sharp drop in performance if using only a visual perception module (VRM) with CNN (VPM w/o *Att Net*) to predict unseen class images via computing the inner product of both extracted image features and class attribute labels. Second, the VRM with both CNN and attention network as the image feature extractor to predict results through the inner product of extracted image features and class attribute labels (VPM w/ *Att Net*). We observe that employing an attribute-guided strategy for extracting visual features benefits our model. Next, we also assess the influence of the logic adapter, where the VRM with CNN and predict results by logic adapter (VPM w/o *Att Net*, w/ LA), which results in better performance compared to the VPM w/ *Att Net*. This demonstrates that the logic adapter not only benefits the capture of discriminative image features but also adapts the model to unseen classes. Finally, we find that by combining all these components, the model can achieve the best results. Furthermore, in our ZSCLR, the feature network is primarily responsible for recognizing discriminative features, so choosing an appropriate feature network is crucial. Therefore, we preliminarily validate the different results produced by tensor networks and MLP. The results is shown in Appendix A.1.

4.4 Qualitative Results

Effectiveness and Interpretability Analyze. To provide a clear illustration of ZSCLR's effectiveness and interpretability, we have employed heatmaps to visualize the discriminative image features learned by the proposed model ZSCLR, Non-Didcriminative method LFGAA [13] and Didcriminative method DUET[6] on the CUB's testing sample. As depicted in Figure 4, the highlighted regions represent the captured discriminative features. Compared to the Non-Didcriminative and Didcriminative method, our ZSCLR obtains good results, e.g., accurately positioning by paying attention to discriminative feature regions. This demonstrates ZSCLR's effectiveness in capturing discriminative image attributes utilizing rule prompt knowledge. Furthermore, by combining the predicted discriminative feature labels with the rule prompts, ZSCLR can infer class labels, e.g., *groove_billed_ani*, and provide an interpretation of the results. This reasoning process is transparent, allowing for easy comprehension of the model's recognition process when presented with an image. For instance, when the model identifies an

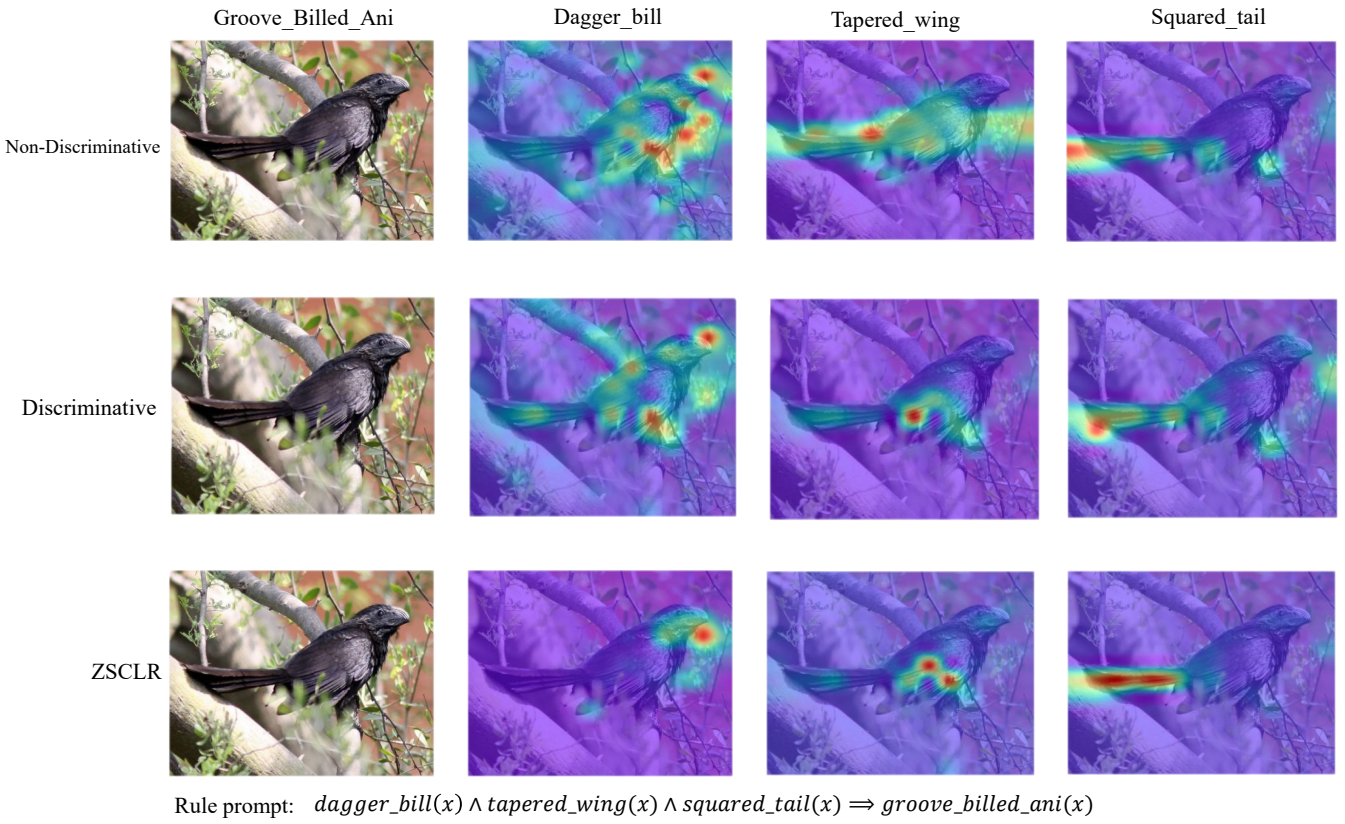

Figure 4: Interpretabilty. Visualization of learned discriminative image features for Non-Didcriminative method LFGAA [13], Didcriminative method DUET[6] and our ZSCLR on the CUB dataset. We highlight the position of three discriminative features, such as the shape of the bill, the shape of the wing, and the shape of the tail in an image.

image as *groove_billed_ani*, it can provide an explanation by highlighting the presence of features like *digger_bill*, *tapered_wing* and *squared_tail* in the image, and then logically deducing that the object possessing these features corresponds to the *groove_billed_ani* class, according to the provided rule prompts. Furthermore, to demonstrate ZSCLR's ability to capture discriminative image features, we visualize features for unseen classes through t-SNE and use Mutual Information (NMI) to quantify results of the cluster on the AwA2 and CUB datasets as shown Figure 5 in Appendix A.2.

5 CONCLUSION AND DISCUSSION

In this paper, we propose the zero-shot image classification model with logic adapter and rule prompt (ZSCLR) to more accurately classify images in zero-shot scenes and provide the results' interpretability. ZSCLR consists of two modules: a visual perception module and a logic adapter. The visual perception module aims to extract image features. At the same time, the logic adapter takes image features from the visual perception module and encodes them and rule prompts via the Markov logic network. During training, the logic adapter refines these image features using backpropagation to derive discriminative image features and adapt the model from seen classes to unseen classes. Furthermore, ZSCLR offers explanations for its prediction results through rule prompts with symbolic discriminative features. Comprehensive experiments conducted on three well-known benchmarks underscore the superior performance of ZSCLR. We believe that ZSCLR presents an innovative direction for the research community, particularly in the context of logic reasoning.

Our ZSCLR not only predicts classes of images but also explains reasons for predictions by providing an interpretation in natural language form. Meantime, the model can locate the position of the discriminative features of objects. Therefore, in the future, ZSCLR has the potential to integrate into visual question models with natural language interactive Q&A. For instance, given an image and its prompt like "there are *bill*, *wing* and *tail* in the image" in the natural language style, the model can answer the following questions like "*What is an object in this image?*", "*Why is the object?*", "*Where are the discriminative features for determining the object?*", and so on.

6 ACKNOWLEDGMENTS

This work was supported by the National Key R&D Program of China under Grant No. 2021ZD0112500; the National Natural Science Foundation of China under Grant Nos. U22A2098, 62172185, 62206105 and 62202200.

REFERENCES

[1] Wieland Brendel and Matthias Bethge. 2019. Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. *International Conference on Learning Representations* (2019).

[2] Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. 2016. Synthesized classifiers for zero-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5327–5336.

[3] Shiming Chen, Ziming Hong, Yang Liu, Guo-Sen Xie, Baigui Sun, Hao Li, Qinmu Peng, Ke Lu, and Xinge You. 2022. Transzero: Attribute-guided transformer for zero-shot learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 330–338.

[4] Shiming Chen, Ziming Hong, Guo-Sen Xie, Wenhan Yang, Qinmu Peng, Kai Wang, Jian Zhao, and Xinge You. 2022. Msdn: Mutually semantic distillation network for zero-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 7612–7621.

[5] Shiming Chen, Wenjie Wang, Beihao Xia, Qinmu Peng, Xinge You, Feng Zheng, and Ling Shao. 2021. Free: Feature refinement for generalized zero-shot learning. In *Proceedings of the IEEE/CVF international conference on computer vision*. 122–131.

[6] Zhuo Chen, Yufeng Huang, Jiaoyan Chen, Yuxia Geng, Wen Zhang, Yin Fang, Jeff Z Pan, and Huajun Chen. 2023. Duet: Cross-modal semantic grounding for contrastive zero-shot learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37. 405–413.

[7] Pedro Domingos and Daniel Lowd. 2019. Unifying logical and statistical AI with Markov logic. *Commun. ACM* 62, 7 (2019), 74–83.

[8] Dat Huynh and Ehsan Elhamifar. 2020. Compositional zero-shot learning via fine-grained dense feature composition. *Advances in Neural Information Processing Systems* 33 (2020), 19849–19860.

[9] Dat Huynh and Ehsan Elhamifar. 2020. Fine-grained generalized zero-shot learning via dense attribute-based attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 4483–4493.

[10] Michael Kampffmeyer, Yinbo Chen, Xiaodan Liang, Hao Wang, Yujia Zhang, and Eric P Xing. 2019. Rethinking knowledge graph propagation for zero-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 11487–11496.

[11] Dehui Kong, Xiliang Li, Shaofan Wang, Jinghua Li, and Baocai Yin. 2023. Learning visual-and-semantic knowledge embedding for zero-shot image classification. *Applied Intelligence* 53, 2 (2023), 2250–2264.

[12] Lu Liu, Tianyi Zhou, Guodong Long, Jing Jiang, Xuanyi Dong, and Chengqi Zhang. 2021. Isometric propagation network for generalized zero-shot learning. *International Conference on Learning Representations* (2021).

[13] Yang Liu, Jishun Guo, Deng Cai, and Xiaofei He. 2019. Attribute attention for semantic disambiguation in zero-shot learning. In *Proceedings of the IEEE/CVF international conference on computer vision*. 6698–6707.

[14] Yang Liu, Lei Zhou, Xiao Bai, Yifei Huang, Lin Gu, Jun Zhou, and Tatsuya Harada. 2021. Goal-oriented gaze estimation for zero-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 3794–3803.

[15] Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg S Corrado, and Jeffrey Dean. 2014. Zero-shot learning by convex combination of semantic embeddings. *International Conference on Learning Representations* (2014).

[16] Genevieve Patterson and James Hays. 2012. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *2012 IEEE conference on computer vision and pattern recognition*. 2751–2758.

[17] Matthew Richardson and Pedro Domingos. 2006. Markov logic networks. *Machine learning* 62, 1 (2006), 107–136.

[18] Edgar Schonfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. 2019. Generalized zero-and few-shot learning via aligned variational autoencoders. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 8247–8255.

[19] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 815–823.

[20] Tasfia Shermin, Shyh Wei Teng, Ferdous Sohel, Manzur Murshed, and Guojun Lu. 2022. Integrated generalized zero-shot learning for fine-grained classification. *Pattern Recognition* 122 (2022), 108246.

[21] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 1–9.

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

[23] Ziyu Wan, Dongdong Chen, Yan Li, Xingguang Yan, Junge Zhang, Yizhou Yu, and Jing Liao. 2019. Transductive zero-shot learning with visual structure constraint. *Advances in Neural Information Processing Systems* 32 (2019).

[24] Jiwei Wei, Yang Yang, Zeyu Ma, Jingjing Li, Xing Xu, and Heng Tao Shen. 2022. Semantic Enhanced Knowledge Graph for Large-Scale Zero-Shot Learning. *arXiv preprint arXiv:2212.13151* (2022).

[25] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. 2010. Caltech-UCSD birds 200. (2010).

[26] Yongqin Xian, Bernt Schiele, and Zeynep Akata. 2017. Zero-shot learning-the good, the bad and the ugly. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4582–4591.

[27] Guo-Sen Xie, Li Liu, Xiaobo Jin, Fan Zhu, Zheng Zhang, Jie Qin, Yazhou Yao, and Ling Shao. 2019. Attentive region embedding network for zero-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 9384–9393.

[28] Guo-Sen Xie, Li Liu, Fan Zhu, Fang Zhao, Zheng Zhang, Yazhou Yao, Jie Qin, and Ling Shao. 2020. Region graph embedding network for zero-shot learning. In *European Conference on Computer Vision*. 562–580.

[29] Wenjia Xu, Yongqin Xian, Jiuniu Wang, Bernt Schiele, and Zeynep Akata. 2020. Attribute prototype network for zero-shot learning. *Advances in Neural Information Processing Systems* 33 (2020), 21969–21980.

[30] Bo Yang, Yuxueqing Zhang, Yida Peng, chunxu Zhang, and Jing Hang. 2021. Collaborative Filtering Based Zero-Shot Learning. *Journal of Software* 32, 9 (2021), 2801–2815.

[31] Guanyu Yang, Kaizhu Huang, Rui Zhang, John Y Goulermas, and Amir Hussain. 2020. Self-focus deep embedding model for coarse-grained zero-shot classification. In *International Conference on Brain Inspired Cognitive Systems*. 12–22.

[32] Dongran Yu, Bo Yang, Qianhao Wei, Anchen Li, and Shirui Pan. 2022. A Probabilistic Graphical Model Based on Neural-Symbolic Reasoning for Visual Relationship Detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 10609–10618.

[33] Yunlong Yu, Zhong Ji, Jungong Han, and Zhongfei Zhang. 2020. Episode-based prototype generating network for zero-shot learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 14035–14044.

[34] Zhongqi Yue, Tan Wang, Qianru Sun, Xian-Sheng Hua, and Hanwang Zhang. 2021. Counterfactual zero-shot and open-set visual recognition. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 15404–15414.

[35] Li Zhang, Tao Xiang, and Shaogang Gong. 2017. Learning a deep embedding model for zero-shot learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2021–2030.

[36] Ziming Zhang and Venkatesh Saligrama. 2016. Zero-shot recognition via structured prediction. In *European Conference on Computer Vision*. 533–548.

[37] Yizhe Zhu, Jianwen Xie, Zhiqiang Tang, Xi Peng, and Ahmed Elgammal. 2019. Semantic-guided multi-attention localization for zero-shot learning. *Advances in Neural Information Processing Systems* 32 (2019).

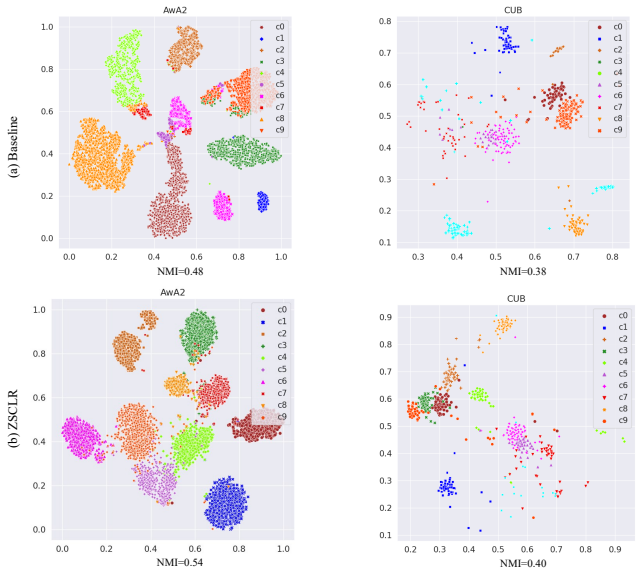

Figure 5: t-SNE visualizations of visual features for unseen classes on AwA2 and CUB dataset in CZSL, respectively. (a) represents the baseline method, while (b) is our ZSCLR. C_i represents different unseen classes.

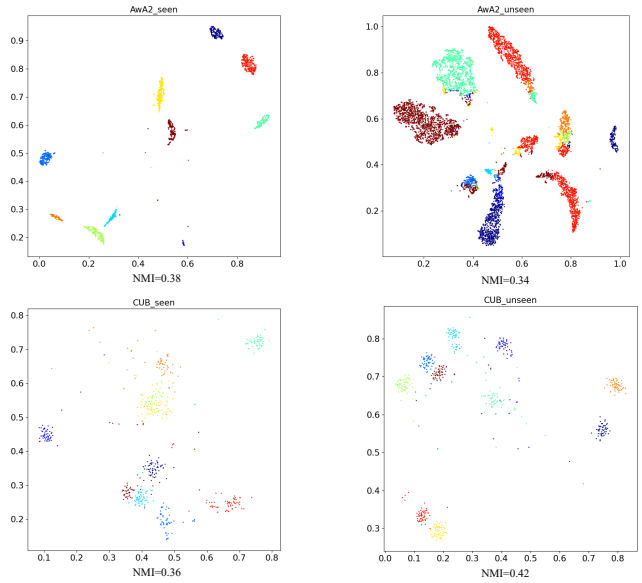

Figure 6: t-SNE visualizations of visual features for both seen and unseen classes on AwA2 and CUB datasets in GZSL, respectively. We show 10 classes in this experiment.

A EXPERIMENT RESULTS

A.1 Ablation of the feature network

To illustrate the experimental results of the model in CZSL(Acc) and GZSL(H) scenarios, we separately present the results for Acc and H. The preliminary results are shown in the table 5. The results indicate that the tensor network is effectively employed, yielding favorable results. This is because tensor networks can integrate information about the relationships between input vectors to enhance prediction capabilities, while MLP only relies on the inputs themselves. Therefore, we choose the tensor network as a feature network in our paper.

Table 5: Ablation studies for different feature networks.

Methods	AwA2(Acc/H)	CUB(Acc/H)
ZSCLR w/tensor network	76.0/73.4	77.8/72.4
ZSCLR w/MLP	70.7/69.6	72.6/70.4

A.2 Discriminative Image Features Analyze

To demonstrate ZSCLR's ability to capture discriminative image features, we visualize features for unseen classes through t-SNE and use Mutual Information (NMI) to quantify results of the cluster on the AwA2 and CUB datasets as shown in Figure 5 and Figure 6. A higher NMI value indicates good cluster quality.

In CZSL scenes, as shown in Figure 5. Compared to the method [13], ZSCLR displays distinct clusters. This affirms that our model effectively learns discriminative and transferable features. It also underscores the logic adapter's role in prompting the visual perception module to capture fine-grained attribute semantics shared between seen and unseen classes, resulting in the learning of discriminative visual representations that facilitate knowledge transfer.

In GZSL scenes, as shown in Figure 6. We visualize visual features of both seen and unseen classes in the AwA2 and CUB datasets. To facilitate visualization, we randomly selected 10 test classes for each dataset. We observed that our ZSCLR shows good clustering results on both seen and unseen classes. This demonstrates the superiority and potential of ZSCLR for knowledge transfer.