LEARNING SEMANTICS-GUIDED VISUAL ATTENTION FOR FEW-SHOT IMAGE CLASSIFICATION

Wen-Hsuan Chu, Yu-Chiang Frank Wang

Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan

ABSTRACT

We propose a deep learning framework for few-shot image classification, which exploits information across label semantics and image domains, so that regions of interest can be properly attended for improved classification. The proposed semantics-guided attention module is able to focus on most relevant regions in an image, while the attended image samples allow data augmentation and alleviate possible overfitting during FSL training. Promising performances are presented in our experiments, in which we consider both closed and open-world settings. The former considers the test input belong to the categories of few shots only, while the latter requires recognition of all categories of interest.

Index Terms— Few-shot learning, image classification

1. INTRODUCTION

Recent development of deep neural networks have shown promising performances in a variety of vision and learning tasks. However, training such networks typically require a large amount of labeled training data beforehand, so that classification, etc. tasks can be performed accordingly. Humans, on the other hand, are generally able to learn new concepts very quickly, even with a limited amount of observation in advance. This motivates the task of few-shot learning (FSL), in which only a limited amount of data for some categories would be available during training.

While we focus on FSL tasks in this paper, we first make a distinction between two FSL settings depending on how classification is performed during testing. The first one is commonly referred to as *N-way few-shot* learning, in which an additional labeled support set from images from N different classes is available during testing (note that N need not be equal to the total amount of classes $|c|$), and the task is to select the most similar image(s) from the support set, with the output value indicating the similarity between the supporting set and the test image [1, 2, 3]. The second setting is the *N-class few-shot* learning problem, where only the test image is presented, with the goal of predicting the correct label [4].

To solve the FSL problem, many different deep learning based approaches have been recently proposed. Existing methods can be generally categorized into two groups.

One class of the approaches is based on metric-learning, with an assumption that images belonging to the same class are grouped closer to each other than those in different classes [1, 2, 3]. The other popular class of methods takes the meta-learning based approach, which aims at training a second network to extract data from a base network, so that classification at a meta level can be performed [5, 6, 7]. More recently, methods based on the incorporation of side information like semantic embedding or attribute vectors have been proposed [8]. For example, Multi-Attention Networks [9] utilizes attention models, and directly consider semantic embedding to construct attention maps for FSL.

To overcome the above limitation, we propose to advance an unsupervised dual attention module to solve for N-class few-shot learning tasks, with applications to image classification. The main novelty of our work lies in learning semantic-specific attention models to guide the neural networks during the training and inference processes. Thus, the most relevant image regions can be properly attended, even without observing a large number of training image data for that class. In our work, we use the positive attention samples along with its complement negative attention samples as a form of data augmentation, which alleviates overfitting problems for FSL. As detailed later, we present a unique unsupervised learning approach to incorporate side semantics information into the training stage. As confirmed later by experiments, our proposed model would produce attention maps with high quality and result in improved FSL performances.

2. PROPOSED MODEL

2.1. Problem Definition and Notations

We now introduce all the notations in this work. Suppose that we are given a (base) dataset S_b of C_b classes, each with a sufficient number of samples. We also observe another (new) set of data (S_n) of C_n classes, each with only a small number of data (note that $C_b \cap C_n = \Phi$). All data are in the form of data-label pairs $S_b, S_n : \{x_i, y_i\}$, where y_i are one-hot vectors with the associated dimensions. For FSL, a unique vector R_i is assigned for each label $c_i \in C_b \cup C_n$. Typically, R_i can either be an unsupervised semantic embedding of the class labels like *word2vec* [10] or *GloVe* [11], or an attribute vector

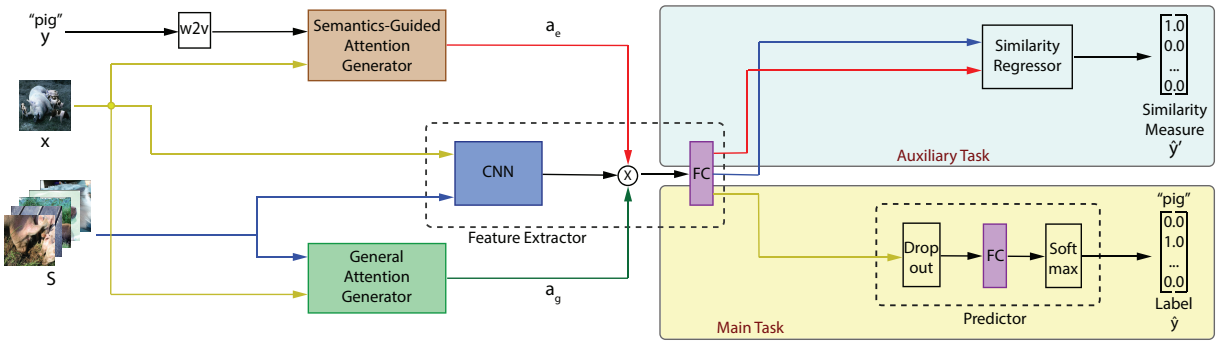

Fig. 1. Our proposed network architecture, consisting of general (green) and semantics-guided (brown) attention generators, CNN-based feature extractor (blue), and main/auxiliary attention-oriented classification. Note that x, y, and S are the input image, label, and support set, respectively. Attention maps a_e and a_g incorporates label embedding and image features for predicting label $\hat{y}$ with a similarity measure $\hat{y}^{'}$ between x and S.

as those defined in the dataset of *Animals with Attributes*.

The goal of FSL is to predict the label for input x, which either belongs to C_n or $C_b \cup C_n$; the former indicates "closed-world" classification, while the latter is a more challenging "open-world" classification. In addition, we consider an addition task of N-way classification, and randomly sample a support set S_i of M images from S_b and S_n, in which only one image shares the same class as x_i. We compute the similarity measure $\hat{y}_i^{'}$ between x_i and each image in S_i. As depicted in Fig. 1, our network architecture consists of three components: general and semantics-guided generators for visual attention θ_e, θ_g, a CNN feature extractor θ_c, and a classification module θ_p. As noted in Sect. 1, our network learns auxiliary information across embedding vectors and image data, aiming at better attending and recognizing images with few shots.

2.2. Semantics-Guided Visual Attention

We propose to learn semantics-guided visual attention for FSL, which jointly considers label semantics together with their image co-occurrence information. While techniques based on object detection [12] is a possible solution for learning particular objects or attributes, it requires ground truth salient regions or bounding boxes for training.

To produce visual attention maps in an unsupervised setting, we consider two different techniques when learning our proposed model. First, we introduce an auxiliary prediction task that allows us to incorporate the class attributes R_i for attention map generation during training. Second, we utilize negative attention samples to further improve the representation capability of our model.

2.2.1. General Attention Generator

Before detailing the two introduced components in our model, we first present the standard attention module, as depicted in Fig. 1. To generate a spatial attention map from an image, we feed the image into a small convolutional neural network, which we call the "general attention module", and multiply the normalized output a_g with the final layers of the deeper CNN, which serves as a feature extractor. The goal of this "general attention module" is to separate the foreground region from the backgrounds, attending over the most relevant regions in the input image. We use a normalized L1-loss to encourage the resulting map to be sparse (i.e., to focus on the foreground regions pixels only):

$$L_{sparse} = \frac{1}{N}\sum |a_i|, \tag{1}$$

where a_i is the pixelwise value of the attention map a_g, and N is the total number of pixels.

2.2.2. Semantics-Guided Attention Generator

To incorporate label information for learning visual attention models, we design an auxiliary classification task with an attention module in Fig. 1, which uses both image x_i and the corresponding embedding vector R_i of label y_i to produce an attention map. We refer to this module as the "semantics-guided attention module", which first uses convolutional layers to extract feature maps from the input image, followed by mapping the embedding vector to a pooling vector:

$$f_i = CNN(x_i) \tag{2}$$

$$v_i = (W_s R_i + b_s). \tag{3}$$

The extracted feature maps f_i are pooled using v_i with a sigmoid function $\sigma(\cdot)$. Its normalized output is calculated as:

$$a_e = \frac{\sigma(f_i v_i)}{max(\sigma(f_i v_i))}. \tag{4}$$

The auxiliary task measures the similarity between the input image x_i and the images in the sampled S_i. We multiply the features f_i extracted from x_i using a CNN by a_e and the

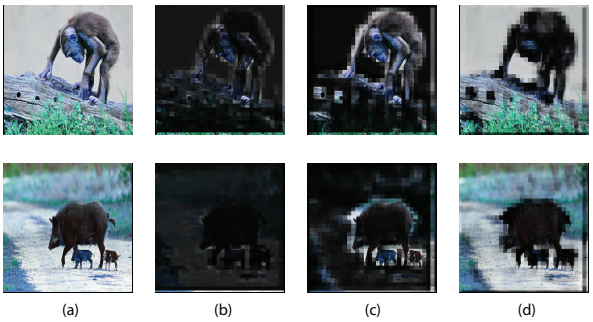

Fig. 2. Example visualization of visual attention of two images of unseen classes from AwA. (a) Original images, (b) semantics-guided attention map a_e, (c) positive attention map a_g, and (d) negative attention map $a_g^{'}$.

features extracted from the image in S_i by a_g. We then compare their similarity as follows:

$$fv_i = \sigma(W \times a_e f_i + b) \tag{5}$$

$$fv_s = \sigma(W \times a_g f_s + b), x_s \in S_i \tag{6}$$

$$\hat{y_i'} = softmax(-(fv_i - fv_s)^2). \tag{7}$$

The loss of this auxiliary prediction task L_{aux_pred} is calculated as the negative cross entropy of the predicted similarity measure $\hat{y_i'}$ and the ground truth y_i'.

Combining this with the loss in (1), we obtain the total loss of the auxiliary task as:

$$L_{aux} = L_{aux\_pred} + \alpha L_{sparse}, \tag{8}$$

where α is the weight for regularization. Finally, the semantics-guided attention map is applied to guide the general attention map by introducing a loss term over their pixelwise values:

$$L_{guide} = \frac{1}{N} \sum |a_e - a_g|. \tag{9}$$

2.2.3. *Negative Attention Samples*

In addition to learning semantic-guided visual attention, we further define and utilize negative attention maps $a_g^{'}$ as the complement of the positive ones a_g, which allow us to better recognize the image categories with few samples. We note that, the negative attention sample is the product of the extracted features and the derived negative attention map.

Ideally, the positive attention samples are the attended image regions associated with the class of interest (i.e., foreground objects), while the negative ones only have irrelevant regions covered (e.g., background or those depicted in Fig. 2). This motivates us to introduce an additional class of "no-object" into C_b and C_n, and we label the negative attention sample accordingly. It is worth noting that, while introducing negative attention samples improves visual attention, this can also be viewed as a *data augmentation* technique, which further alleviates possible overfitting problems in FSL.

Algorithm 1 Learning of Our Network

Input: Data, label, support set, and similarity measure tuples $\{(x_i, y_i, S_i, y_i^{'})\}$; parameters α_1, α_2
Output: Network weights $\theta e, \theta g, \theta c, \theta p$
  for number of training iterations **do**
    Sample k from uniform distribution between [0, 1]
    if $k < \alpha_1$ **then**
      Compute $\hat{y_i}$ with a_g and x_i
      Update $\theta g, \theta c, \theta p$ via (10) with SGD
    else if $\alpha_1 < k < \alpha_2$ **then**
      Compute $\hat{y_i}$ with x_i
      Update $\theta c, \theta p$ via $L_{pred} + \beta L_{center}$ with SGD
    else
      Compute $\hat{y_i}$ with $a_g^{'}$ and x_i
      Update $\theta g, \theta c, \theta p$ via L_{pred} with SGD
    end if
    Update $\theta e, \theta c$ according to (8) with SGD
  end for

2.3. Learning of Our Network

With the introduced modules and losses presented in Sect. 2.2, the total loss for FSL classification is now summarized as:

$$L = L_{pred} + L_{guide} + \alpha L_{sparse} + \beta L_{center}, \tag{10}$$

where $L_{center} = (fv_i - t_k)^2$ for $x_i \in c_k$ is the center loss [13] regularizing the derived features for each class (fv_i is calculated using (5) but with a_g instead of a_e, and t_k is the centroid of class c_k). L_{pred} is the negative cross-entropy loss computed on the predicted labels $\hat{y_i}$. Regularization weights α and β are fixed as 0.05 and 0.1, respectively in our work.

To train our network, the main and auxiliary classification tasks are optimized alternatively according to (10) and (8). For the main classification task, we freeze the weights of θ_e and update θ_c, θ_g, and θ_p. For the auxiliary task, we freeze the weights of θ_g and only allow the updates for θ_c and θ_e.

For each sampled minibatch, we randomly select the attention map that is multiplied before producing f_i for the image. We select the negative attention map $a_g^{'}$ 20% of the time during training data sampling, updating the "no-class" label using just the prediction loss L_{pred}. We select the positive attention map a_g 40% of the time, using the joint loss of (10). For the remaining 40% of the time, we simply take the original images and update via $L_{pred}+\beta L_{center}$. The pseudo-code for network training is summarized in Algorithm 1.

3. EXPERIMENTS

To evaluate the performance of our proposed model, we consider the datasets of CIFAR-100 [14] and Animals with Attributes (AwA) [15] for experiments. For the baseline, we use a regular CNN with dropout applied after the final convolutional layer (denoted as "Baseline"). We also compare this baseline with the center loss [13] (denoted as "Center"). For

Table 1. Results on CIFAR-100 for the closed-world setting.

	k=1	k=2	k=5	k=10
Baseline	22.25%	31.20%	42.00%	49.40%
Center [13]	24.05%	36.35%	44.25%	51.60%
Ours	25.20%	38.40%	43.80%	54.30%

Table 2. Results on CIFAR-100 for the open-world setting. For each method, the top and bottom rows show the accuracy on $C_b \cup C_n$ and C_n only, respectively.

	k=1	k=2	k=5
Baseline	10.67%	17.35%	29.61%
	39.35%	32.60%	17.45%
Center [13]	9.82%	18.10%	31.65%
	41.75%	36.90%	17.85%
Ours	11.36%	21.68%	34.88%
	40.40%	35.65%	20.80%

our method, we use the positive attention samples to calculate the centroid of class c_k. While related work using AwA exist, they focus on image retrieval and thus cannot be easily applied for comparisons. We set the number of classes M in the support set S_i to 5 for all our experiments, and we observe that increasing M does not have significant impacts on the performance. For the hyperparameters, we fix α_1 to 0.4 and α_2 to 0.8.

To learn our proposed network, we first train $\theta e, \theta g, \theta c, \theta p$ using S_b. Once converged, we reset the predictor weights θp, followed by finetuning θc and retraining θp on k training samples from C_n or $C_n \cup C_b$ (with θe and θg fixed). We note that, the above training strategy would alleviate possible overfitting problems during FSL training. Once the training stage is complete, we apply either the "closed-world" or the more challenging "open-world" settings for evaluation (i.e., recognize the input as classes of simply C_n or from $C_b \cup C_n$).

3.1. CIFAR-100

For CIFAR-100, we randomly sample 20 classes from the 100 classes as the "novel" classes, in which only few shots per class would be available. We consider the closed-world and open-world evaluation settings.

Table 1 lists the performance comparisons for the closed-world setting with different k numbers (recall that k is the number of data samples available during training for classes selected from C_n or $C_b \cup C_n$). It can be seen that, our method exploiting label information and guiding the resulting attention achieved the best performance, while the baseline model or the one with additional center loss were not able to produce comparable results.

Table 2 compares the performances in the open-world setting (with 8000 testing data for classes in C_b and 2000 for classes in C_n). From this table, we observe a performance drop in overall accuracy, which is expected. It is worth noting that, applying the center loss actually produced poorer results than the baseline did. This implies possible overfitting problems with center loss for FSL. Nevertheless, our proposed model achieved favorable results over all k numbers.

Table 3. Results on AwA for the closed-world setting.

	k=1	k=2	k=5	k=10
Baseline	31.78%	33.81%	42.53%	50.89%
Center [13]	37.34%	37.92%	48.80%	55.12%
Ours	40.97%	42.80%	55.06%	62.85%

3.2. Animals with Attributes

For the AwA dataset, we use the original 40/10 class split provided and pre-process the image size into 256x256 pixels. For simplicity, we take the classes of the first 10 images as the novel classes, and consider the first k images as training data. The remaining $|c_i| - 10$ samples are then used for evaluation, where $|c_i|$ is the number of samples provided for each of the 10 "novel" classes. For evaluation, we only consider the closed-world setting, as no testing data for the other 40 classes are available. Note that we do not further finetune the parameters for AwA and report the results directly. This also allows us to assess the generalization of our proposed model.

Table 3 lists the performances on AwA using different methods. From this table, we see that our proposed model performed favorably against the other two. Compared to the results on CIFAR-100, we observe a remarkably better improvement of our method over baseline. This is possible due to the fact that the object of interest in AwA generally occupies a smaller portion in an image when comparing to those in CIFAR-100. Thus, our learning of semantics-guided attention modules with positive/negative samples is more preferable for solving FSL tasks.

4. CONCLUSION

In this paper, we presented a deep neural network architecture for few-shot image classification. Our proposed model incorporates semantic embedding of labels during training, which allow us to attend on image regions of interest even for the object categories with few shots. Moreover, the attended positive and negative image samples can be viewed as augmented data, which further alleviate possible overfitting for FSL. In our experiments, our method performed favorably against baseline and recent deep learning approaches in both closed-world and (more challenging) open-world settings, which supports the use of our model for FSL.

Acknowledgments This work was supported by the Ministry of Science and Technology of Taiwan under grant MOST 107-2634-F-002-010.

5. REFERENCES

[1] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML Deep Learning Workshop*, 2015, vol. 2.

[2] Oriol Vinyals, Charles Blundell, Tim Lillicrap, Koray Kavukcuoglu, and Daan Wierstra, "Matching networks for one shot learning," in *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, 2016, pp. 3630–3638.

[3] Jake Snell, Kevin Swersky, and Richard Zemel, "Prototypical networks for few-shot learning," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., pp. 4080–4090. Curran Associates, Inc., 2017.

[4] Bharath Hariharan and Ross B. Girshick, "Low-shot visual recognition by shrinking and hallucinating features," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 2017, pp. 3037–3046.

[5] Adam Santoro, Sergey Bartunov, Matthew Botvinick, Daan Wierstra, and Timothy P. Lillicrap, "One-shot learning with memory-augmented neural networks," *CoRR*, vol. abs/1605.06065, 2016.

[6] Sachin Ravi and Hugo Larochelle, "Optimization as a model for few-shot learning," 2016.

[7] Chelsea Finn, Pieter Abbeel, and Sergey Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," *arXiv preprint arXiv:1703.03400*, 2017.

[8] Yao-Hung Hubert Tsai and Ruslan Salakhutdinov, "Improving one-shot learning through fusing side information," *arXiv preprint arXiv:1710.08347*, 2017.

[9] Peng Wang, Lingqiao Liu, Chunhua Shen, Zi Huang, Anton van den Hengel, and Heng Tao Shen, "Multi-attention network for one shot learning," in *2017 IEEE conference on computer vision and pattern recognition, CVPR*, 2017, pp. 22–25.

[10] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds., pp. 3111–3119. Curran Associates, Inc., 2013.

[11] Jeffrey Pennington, Richard Socher, and Christopher Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[12] Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, 2014, pp. 580–587.

[13] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao, "A discriminative feature learning approach for deep face recognition," in *European Conference on Computer Vision*. Springer, 2016, pp. 499–515.

[14] Alex Krizhevsky and Geoffrey Hinton, "Learning multiple layers of features from tiny images," 2009.

[15] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata, "Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly," *arXiv preprint arXiv:1707.00600*, 2017.