Improving One-Shot Learning through Fusing Side Information

Yao-Hung Hubert Tsai **Ruslan Salakhutdinov**
Machine Learning Department, School of Computer Science, Carnegie Mellon University
{yaohungt, rsalakhu}@cs.cmu.edu

1 Introduction

Deep neural networks (DNNs) often struggle when training on classes with very few samples. In this paper, we focus on the extreme case: *one-shot learning* which has only one training sample per category. We treat the problem of one-shot learning to be a transfer learning problem: how to efficiently transfer the knowledge from 'lots-of-examples' to 'one-example' classes. More precisely, we propose to fuse side information for compensating the missing information across classes. In our paper, side information represents the relationship or prior knowledge between categories: for example, unsupervised feature vectors of categories derived from Wikipedia such as Word2Vec vectors (Mikolov et al., 2013), or tree hierarchy label structure such as WordNet structure (Miller, 1995).

We propose to first integrate side information using Hilbert-Schmidt Independence Criterion (HSIC) (Gretton et al., 2005) between the learned data embeddings and the learned label-affinity kernel, which is inferred from the side information. Since HSIC serves as a statistical dependency measurement, our learned feature representations can be maximally dependent on the corresponding label space. Next, to achieve better adaptation from 'lots-of-examples' to 'one-examples' classes, we introduce an attention mechanism for 'lots-of-examples' classes on the learned label-affinity kernel.

We empirically show that our proposed learning architecture (see Fig. 1) improves over traditional softmax regression networks as well as state-of-the-art attentional regression networks (Vinyals et al., 2016) on one-shot recognition tasks.

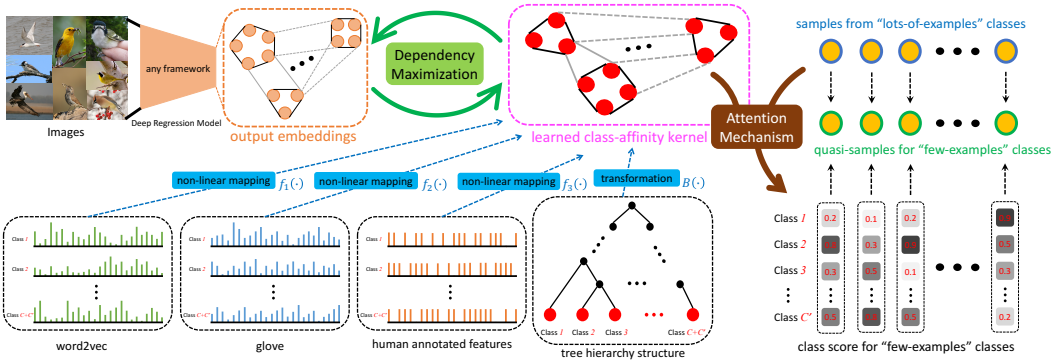

Figure 1: Fusing side information when learning data representation. We first construct a label-affinity kernel through deep kernel learning using multiple types of side information. Then, we enforce the dependency maximization criteria between the learned label-affinity kernel and the output embeddings of a regression model. Samples in 'lots-of-examples' classes are used to generate quasi-samples for 'one-example' classes. These generated quasi-samples can be viewed as additional training data.

2 Proposed Method

2.1 Notation

Let $\mathbf{S}$ denote the support set for the classes with lots of training examples. $\mathbf{S}$ consists of N data-label pairs $\mathbf{S} = \{\mathbf{X}, \mathbf{Y}\} = \{x_i, y_i\}_{i=1}^{N}$, where y_i ranges within C classes. We assume that we have M different kinds of side information $\mathbf{R} = \{R^1, R^2, \cdots, R^M\}$, where R^m can either be supervised/ unsupervised class embeddings or even hierarchical structures inferred from tree-based object structures such as ImageNet (Krizhevsky et al., 2012). Similarly, we have a different support set $\mathbf{S}'$ for 'one-examples' classes that $\mathbf{S}' = \{\mathbf{X}', \mathbf{Y}'\} = \{x'_i, y'_i\}_{i=1}^{N'}$ in which y'_i ranges within C' classes (disjoint from the classes in $\mathbf{S}$). Side information $\mathbf{R}' = \{R'^1, R'^2, \cdots, R'^M\}$ for $\mathbf{S}'$ is also provided. Last, θ_X and θ_R are the model parameters dealing with the data and side information, respectively.

2.2 Dependency Measure on Data and Side Information

The output embeddings $g_{\theta_X}(\mathbf{X})$ and side information $\mathbf{R}$ can be seen as two interdependent random variables, and we hope to maximize their dependency on each other. To achieve this goal, we adopt Hilbert-Schmidt Independence Criterion (HSIC) (Gretton et al., 2005). HSIC acts as a non-parametric independence test between two random variables, $g_{\theta_X}(\mathbf{X})$ and $\mathbf{R}$, by computing the Hilbert-Schmidt norm of the covariance operator over the corresponding domains $\mathcal{G} \times \mathcal{R}$. Furthermore, let k_g and k_r be the kernels on $\mathcal{G}, \mathcal{R}$ with associated Reproducing Kernel Hilbert Spaces (RKHSs). A slightly biased empirical estimation of HSIC (Gretton et al., 2005) could be written as follows:

$$\mathrm{HSIC}(\mathbf{S}, \mathbf{R}) = \frac{1}{(N-1)^2}\mathrm{tr}(\mathbf{H}\mathbf{K}_\mathbf{G}\mathbf{H}\mathbf{K}_\mathbf{R}), \tag{1}$$

where $\mathbf{K}_\mathbf{G} \in \mathbb{R}^{N\times N}$ with $\mathbf{K}_{\mathbf{G}ij} = k_g(x_i, x_j) = g_{\theta_X}(x_i)^\top \cdot g_{\theta_X}(x_j)$, $\mathbf{K}_\mathbf{R} \in \mathbb{R}^{N\times N}$ with $\mathbf{K}_{\mathbf{R}ij} = k_r(y_i, y_j) = \sum_{m=1}^{M} \frac{1}{M} k_{r^m}(y_i, y_j)$, and $\mathbf{H} \in \mathbb{R}^{N\times N}$ with $\mathbf{H}_{ij} = \mathbb{1}_{\{i=j\}} - \frac{1}{(N-1)^2}$. We consider two variants of $k_{r^m}(\cdot,\cdot)$ based on whether R^m is represented by class embeddings or tree-based label hierarchy. In short, $\mathbf{K}_\mathbf{G}$ and $\mathbf{K}_\mathbf{R}$ respectively stand for the relationships between data and categories, and HSIC provides a statistical dependency guarantee on the learned embeddings and labels.

a) R^m is represented by class embeddings:

Class embeddings can either be supervised features such as human annotated features or unsupervised features such as *word2vec* or *glove* features. Given $R^m = \{r_c^m\}_{c=1}^{C}$ with r_c^m representing class embeddings of class c, we define $k_{r^m}(\cdot,\cdot)$ as:

$$k_{r^m}(y_i, y_j) = f_{m,\theta_R}(r_{y_i}^m)^\top \cdot f_{m,\theta_R}(r_{y_j}^m),$$

where $f_{m,\theta_R}(\cdot)$ denotes the non-linear mapping from R^m. In this setting, we can capture the intrinsic structure by adjusting the categories' affinity through learning $f_{m,\theta_R}(\cdot)$ for different types of side information R^m.

b) R^m is represented by tree hierarchy:

If the labels form a tree hierarchy (e.g., *wordnet* (Miller, 1995) tree structure in ImageNet), then we can represent the labels as a tree covariance matrix $\mathbf{B}$ defined in Bravo et al. (2009), which is proved to be equivalent to the taxonomies in the tree (Blaschko et al., 2013). Specifically, following the definition of Theorem 2 in Bravo et al. (2009), a matrix $\mathbf{B} \in \mathbb{R}^{C\times C}$ is the tree-structured covariance matrix if and only if $\mathbf{B} = \mathbf{V}\mathbf{D}\mathbf{V}^\top$ where $\mathbf{D} \in \mathbb{R}^{2C-1\times 2C-1}$ is the diagonal matrix indicating the branch lengths of the tree and $\mathbf{V} \in \mathbb{R}^{C\times 2C-1}$ denoting the topology.

For any given tree-based label hierarchy, we define $k_{r^m}(\cdot,\cdot)$ to be

$$k_{r^m}(y_i, y_j) = (\mathbf{B}^m)_{y_i,y_j} = (\mathbf{Y}^\top \mathbf{B}^m \mathbf{Y})_{i,j} \ ,$$

where $\mathbf{Y} \in \{0,1\}^{C\times N}$ is the label matrix and $\mathbf{B}^m$ is the tree-structured covariance matrix of R^m. In other words, $k_{r^m}(y_i, y_j)$ indicates the weighted path from the root to the nearest common ancestor of nodes y_i and y_j (see Lemma 1 in (Blaschko et al., 2013)).

In eq. (1), we can try integrating different types of side information R^m with both class-embedding and tree-hierarchy-structure representation. In short, maximizing eq. (1) makes the data representation

kernel $\mathbf{K_G}$ maximally dependent on the side information $\mathbf{R}$ seen from the kernel matrix $\mathbf{K_R}$. Hence, introducing HSIC criterion provides an excellent way of transferring knowledge across different classes. Note that, if $\mathbf{K_R}$ is an identity matrix, then there are no relationships between categories, which results in a standard classification problem.

So far, we have defined a joint learning on the support set $\mathbf{S}$ and its side information $\mathbf{R}$. If we have access to different support set $\mathbf{S}'$ and the corresponding side information $\mathbf{R}'$, we can easily incorporate them into the HSIC criterion; i.e., $\text{HSIC}(\{\mathbf{S}, \mathbf{S}'\}, \{\mathbf{R}, \mathbf{R}'\})$. Hence we can effectively transfer the knowledge both intra and inter sets.

2.3 Quasi-Samples Generation

Our second aim is to use a significant amount of data in 'lots-of-examples' classes to learn the prediction model for 'one-example' classes. We present an attention mechanism over the side information $\mathbf{R}$ and $\mathbf{R}'$ to achieve this goal.

For a given data-label pair $\{x, y\}$ in $\mathbf{S}$, we define its quasi-label $\tilde{y}'$ as follows:

$$\tilde{y}' = P_{\theta_R}(y'|y; \mathbf{R}, \mathbf{R}') = \sum_{i \in \mathbf{S}'} a_r(y, y'_i) y'_i,$$

where $a_r(\cdot, \cdot)$ acts as an attentional kernel from $\mathbf{R}$ to $\mathbf{R}'$, which can be formulated as

$$a_r(y, y'_i) = \frac{e^{k_r(y, y'_i)}}{\sum_{j \in \mathbf{S}'} e^{k_r(y, y'_j)}}.$$

In other words, given the learned label affinity kernel, for each category in 'lots-of-examples' classes, we can form a label probability distribution on the label space for 'one-example' classes; i.e., $\tilde{y}' = P_{\theta_R}(y'|y; \mathbf{R}, \mathbf{R}')$. Moreover, given the other set $\mathbf{S}'$, we can also derive the label probability distribution $P_{\theta_X}(y'|x; \mathbf{S}')$ under any regression model for 'one-example' classes. Our strategy is to minimize the cross entropy between $P_\theta(y'|x; \mathbf{S}')$ and $\tilde{y}'$. In short, we can treat each data-label pair $\{x, y\}$ in 'lots-of-examples' classes to be a quasi-sample $\{x, \tilde{y}'\}$ for 'one-example' classes, as illustrated in Fig. 2.

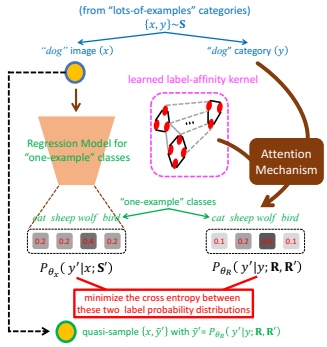

Figure 2: Quasi-samples generation: We take *dog* as an example class from "lots-of-examples" categories. "One-example" categories consist of *cat, sheep, wolf,* and *bird*. Best viewed in color.

2.4 Objectives

The overall training objective is defined as follows:

$$\max \ \alpha \, \text{HSIC}\big(\{\mathbf{S}, \mathbf{S}'\}, \{\mathbf{R}, \mathbf{R}'\}\big) + \frac{1}{|\mathbf{S}|} \sum_{i \in \mathbf{S}} y_i^\top \log P_{\theta_X}\big(y_i|x_i; \mathbf{S}\big) + \alpha \, \tilde{y}_i'^\top \log P_{\theta_X}\big(y_i'|x_i; \mathbf{S}'\big),$$

where α is the trade-off parameter.

For any given test example x'_{test}, the predicted output class is defined as

$$\hat{y}'_{test} = \text{argmax}_{y'} \, P_{\theta_X}(y'|x'_{test}; \mathbf{S}').$$

Table 1: Average performance (%) over 40 random trials for standard one-shot recognition task.

Dataset	softmax_net	HSIC$^{\dagger}_{softmax}$	HSIC$_{softmax}$	attention_net [Vinyals et al. (2016)]	HSIC$^{\dagger}_{attention}$	HSIC$_{attention}$
CUB	26.93 ± 2.41	29.26 ± 2.22	$\mathbf{31.49 \pm 2.28}$	29.12 ± 2.44	33.12 ± 2.48	$\mathbf{33.75 \pm 2.43}$
AwA	66.39 ± 5.38	69.98 ± 5.47	$\mathbf{71.29 \pm 5.64}$	72.27 ± 5.82	$\mathbf{77.86 \pm 4.76}$	76.98 ± 4.99

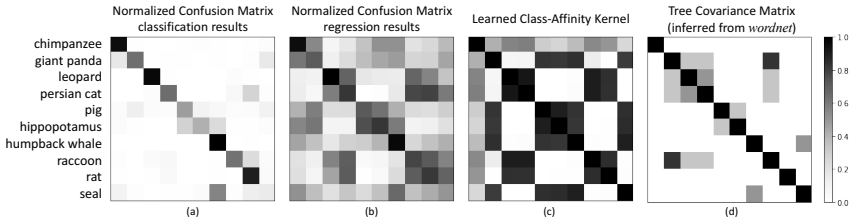

Figure 3: For AwA dataset: (a) normalized confusion matrix for classification, (b) normalized confusion matrix for regression, (c) learned class-affinity kernel in proposed$_{attention}$, and (d) tree covariance matrix.

3 EVALUATION

We evaluate our method (HSIC$_{softmax}$ and HSIC$_{attention}$) on top of two different regression networks: traditional softmax regression (softmax_net) and attentional regression (attention_net) introduced by (Vinyals et al., 2016). Two datasets are adopted for one-shot recognition task: Caltech-UCSD Birds 200-2011 (CUB) (Welinder et al., 2010) and Animals with Attributes (AwA) (Lampert et al., 2014). CUB is a fine-grained dataset in which the categories are both visually and semantically similar, while AwA is a general dataset. Four types of side information are considered: supervised human annotated attributes (att) (Lampert et al., 2014), unsupervised Word2Vec features ($w2v$) (Mikolov et al., 2013), unsupervised Glove features (glo) (Pennington et al., 2014), and the hierarchy tree structures (hie) inferred from *wordnet* (Miller, 1995). We also provide two variants (HSIC$^{\dagger}_{softmax}$ and HSIC$^{\dagger}_{attention}$) when considering no quasi-samples generation technique.

One-Shot Recognition Task: Table 1 lists the average recognition performance for our standard one-shot recognition experiments. HSIC$_{softmax}$ and HSIC$_{attention}$ are jointly learned with all four types of side information: att, $w2v$, glo, and hie. We first observe that the methods with side information achieve superior performance over the methods which do not learn with side information. For example, HSIC$_{softmax}$ improves over softmax_net by 4.56% on CUB dataset and HSIC$_{attention}$ enjoys 4.71% gain over attention_net on AwA dataset. These results indicate that fusing side information can benefit one-shot learning.

Next, we examine the variants of our proposed architecture. In most cases, the construction of the quasi-samples benefits the one-shot learning. The only exception is the 0.88% performance drop from HSIC$^{\dagger}_{attention}$ to HSIC$_{attention}$ in AwA. Nevertheless, we find that our model converges faster when introducing the technique of generating quasi-samples.

Confusion Matrix and the Learned Class-Affinity Kernel: Following the above experimental setting, for test classes in AwA, in Fig. 3, we provide the confusion matrix, the learned label-affinity kernel using HSIC$_{attention}$, and the tree covariance matrix (Bravo et al., 2009). We first take a look at the normalized confusion matrix for classification results. For example, we observe that *seal* is often misclassified as *humpback whale*; and from the tree covariance matrix, we know that *seal* is semantically most similar to *humpback whale*. Therefore, even though our model cannot predict *seal* images correctly, it still can find its semantically most similar classes.

Additionally, it is not surprising that Fig. 3(b), normalized confusion matrix, is visually similar to Fig. 3(c), the learned class-affinity kernel. The reason is that one of our objectives is to learn the output embeddings of images to be maximally dependent on the given side information. Note that, in this experiment, our side information contains supervised human annotated attributes, unsupervised word vectors (Word2Vec (Mikolov et al., 2013) and Glove (Pennington et al., 2014)), and a *WordNet* (Miller, 1995) tree hierarchy.

On the other hand, we also observe the obvious change in classes relationships from *WordNet* tree hierarchy (Fig. 3 (d)) to our learned class-affinity kernel (Fig. 3 (c)). For instance, *raccoon* and *giant panda* are species-related, but they distinctly differ in size and color. This important information is missed in *WordNet* but not missed in human annotated features or word vectors extracted from Wikipedia. Hence, our model bears the capability of arranging and properly fusing various types of side information.

Table 2: Average performance (%) for the different availability of side information.

CUB							
available side information	*none*	*att*	*w2v*	*glo*	*hie*	*att/w2v/glo*	*all*
HSIC$_{softmax}$	26.93 ± 2.41	30.93 ± 2.25	30.67 ± 2.10	30.53 ± 2.42	32.15 ± 2.28	30.58 ± 2.12	31.49 ± 2.28
HSIC$_{attention}$	29.12 ± 2.44	32.86 ± 2.34	33.37 ± 2.30	33.31 ± 2.50	**34.10 ± 2.40**	33.72 ± 2.45	33.75 ± 2.43
AwA							
available side information	*none*	*att*	*w2v*	*glo*	*hie*	*att/w2v/glo*	*all*
HSIC$_{softmax}$	66.39 ± 5.38	70.08 ± 5.27	69.30 ± 5.41	69.94 ± 5.62	73.32 ± 5.12	70.44 ± 6.74	71.29 ± 5.64
HSIC$_{attention}$	72.27 ± 5.82	76.60 ± 5.05	76.60 ± 5.15	**77.38 ± 5.15**	76.88 ± 5.27	76.84 ± 5.65	76.98 ± 4.99

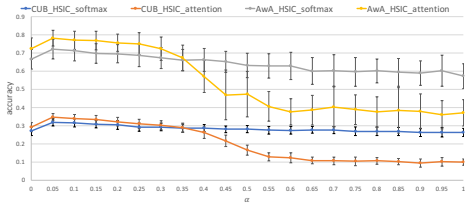

Figure 4: Parameter sensitivity analysis experiment. Our proposed methods jointly learn with all four side information: *att*, *w2v*, *glo*, and *hie*. Best viewed in color.

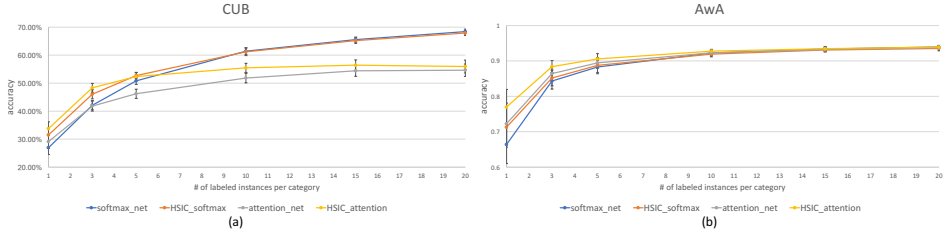

Figure 5: Experiment for increasing labeled instance per category in test classes. Our proposed methods jointly learn with all four side information: *att*, *w2v*, *glo*, and *hie*. Best viewed in color.

Availability of Various Types of Side Information: In Table 2, we evaluate our proposed methods when not all four types of side information are available during training. It is surprising to find that there is no particular rule of combining multiple side information or using a single side information to obtain the best performance. A possible reason would be the non-optima for using kernel average. That is to say, in our current setting, we equally treat contribution of every type of side information to the learning of our label-affinity kernel. Nevertheless, we still enjoy performance improvement of using side information compared to not using it.

Parameter Sensitivity on α: Since α stands for the trade-off parameter for fusing side information through HSIC and quasi-examples generation technique, we studied how it affects model performance. We alter α from 0 to 1.0 by step size of 0.05 for both HSIC$_{softmax}$ and HSIC$_{attention}$ models. Fig. 4 shows that larger values of α does not lead to better performance. When $\alpha \leq 0.3$, our proposed method outperforms softmax_net and attention_net. Note that HSIC$_{softmax}$ and HSIC$_{attention}$ relax to softmax_net and attention_net when $\alpha = 0$. When $\alpha > 0.3$, the performance of our proposed method begins to drop significantly, especially for HSIC$_{attention}$. This is primarily because too large values of α may cause the output embeddings of images to be confused by semantically similar but visually different classes in the learned label-affinity kernel (e.g., Fig. 3 (c)).

From One-Shot to Few-Shot Learning: In Fig. 5, we increase the labeled instances in test classes and evaluate the performance of softmax_net, attention_net, and our proposed architecture. We observe that HSIC$_{softmax}$ converges to softmax_net and HSIC$_{attention}$ converges to attention_net when more labeled data are available in test classes during training. In other words, as labeled instances increase, the power of fusing side information within deep learning diminishes. This result is quite intuitive as deep architecture perform well when training on lots of labeled data.

For the fine-grained dataset CUB, we also observe that *attentional regression* methods are at first outperform *softmax regression* methods, but perform worse when more labeled data are present during training. Note that *softmax regression* networks have one additional softmax layer (one-hidden-layer fully-connected neural network) compared to *attentional regression* networks. Therefore, *softmax regression* networks can deal with more complex regression functions (i.e., regression for the fine-grained CUB dataset) as long as they have enough labeled examples.

Acknowledgements

This work was supported by DARPA award D17AP00001, Google focused award, and Nvidia NVAIL award.

References

Blaschko, M. B., Zaremba, W., and Gretton, A. (2013). Taxonomic prediction with tree-structured covariances. In *ECML-PKDD*.

Bravo, H. C., Wright, S. J., Eng, K. H., Keles, S., and Wahba, G. (2009). Estimating tree-structured covariance matrices via mixed-integer programming. In *AISTATS*.

Gretton, A., Bousquet, O., Smola, A., and Schölkopf, B. (2005). Measuring statistical dependence with hilbert-schmidt norms. In *International conference on algorithmic learning theory*.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *NIPS*.

Lampert, C. H., Nickisch, H., and Harmeling, S. (2014). Attribute-based classification for zero-shot visual object categorization. *IEEE T-PAMI*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *NIPS*.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *EMNLP*.

Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al. (2016). Matching networks for one shot learning. In *NIPS*.

Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., and Perona, P. (2010). Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology.