# Adapting New Categories for Food Recognition with Deep Representation

Shuang Ao
Department of Computer Science
Western University
London, Ontario
Email: sao@uwo.ca

Charles X. Ling
Department of Computer Science
Western University
London, Ontario
Email: cling@csd.uwo.ca

***Abstract*—**

## I. Introduction

## II. Related Works

The motivation of transfer knowledge between different domains is to apply the previous information from the source domain to the target one, assuming that there exists certain relationship, explicit or implicit, between the feature space of these two domains [1]. Technically, previous work can be concluded into solving the following three issues: what, how and when to transfer [2].

**What to transfer.** Previous work tried to answer this question from three different aspects: selecting transferable instances, learning transferable feature representations and transferable model parameters. Instance-based transfer learning assume that part of the instances in the source domain could be re-used to benefit the learning for the target domain. Lim et al. proposed a method of augmenting the training data by borrowing data from other classes for object detection [3]. Learning transferable features means to learn common feature that can alleviate the bias of data distribution in target domain. Recently, Long et al. proposed a method that can learn transferable features with deep neural network and showed some impressive results on the benchmarks [4]. Parameter transfer approach assumes that the parameters of the model for the source task can be transfered to the target task. Yang et al. proposed Adaptive SVMs by transferring parameters by incorporating the auxiliary classifier trained from source domain [5]. On top of Yang's work, Ayatar et al proposed PMT-SVM that can determine the transfer regularizer according to the target data automatically [6]. Tommasi et al. proposed Multi-KT that can utilize the parameters from multiple source models for the target classes [2]. Kuzborskij et al. proposed a similar method to learn new categories by leveraging over the known source [7].

**When and how to transfer.** The question *when to transfer* arises when we want to know if the information acquired from previous task is relevant to the new one (i.e. in what situation, knowledge should not be transfered). In some extreme situation, where the source domain and target one are not relevant, brutal-force transferring the knowledge between them could even degrade the performance of the classifier in target domain. This is often referred to as negative transfer. *How to transfer* the prior knowledge effectively should be carefully designed to prevent inefficient and negative transfer. Some previous work consists in using generative probabilistic method [8] [9] [10]. Bayesian learning methods can predict the target domain by combining the prior source distribution to generate a posterior distribution. Alternatively, some previous max margin methods show that it is possible to learn from a few examples by minimizing the Leave-One-Out (LOO) error for the training model [7] [11]. Previous work shows that there is a closed-form implementation of LOO cross-validation that can generate unbiased model estimation for LS-SVM [12].

Our work correspond to the context above. In this paper, we propose a method based on parameter transfer approach with max margin classifier. We address our work on how to prevent negative transfer when we are not sure whether the prior knowledge is relevant to our target task. We mathematically proof that, without any data distrbution assumption, the superior bound of the training loss for our transfer method is the loss of a method learning directly (i.e. without using any prior knowledge). This indecates that when the prior knowledge hurts the transfer procedure, our method can avoid negative transfer automatically.

## III. Problem Statement

Our method works in the following scenario. There is a image dataset (source data) containing $N$ categories and a classifier trained from this dataset to distinguish these $N$ categories. This (source) classifier and the features used to learn it is publicly accessible while the dataset itself is private (unknown distribution). Now we collect our own image dataset (target data) coming from $N+1$ categories. This target dataset consists of $N$ identical categories to the source data and one new category related to the previous $N$ categories. In order to train a new classifier for our new task, we would expect our classifier to get better results with respect to

- Maximize positive transfer. Since we know that these two task share some information, our classifier should transfer useful information as much as possible.
- Minimize negative transfer. The data distribution of the source data is unknown. In the extreme situation, the data distribution of the two task could be totally different. Then the knowledge from previous task is negative transfer and should be disposed.

In this paper, we focus our work on transferring the knowledge with LS-SVM as the classifier for multi-class transfer problem. In the following we briefly introduce the mathermatical setting of our problem and show

### A. LS-SVM Setting and Definition

Here we introduce the notations used in the rest of the paper. We use any letter with apostrophe to denote the information from the source data, e.g. if $f(x)$ denotes the model for the target task, $f'(x)$ denotes the model for the source one.

TABLE I. USEFUL NOTATIONS IN THIS PAPER

| | |
|---|---|
| $f'(x)$ | binary function for source task |
| $f(x)$ | binary function for target task |
| $\phi(x)$ | function mapping the input sample into a high dimensional feature space. |
| $K(x,x)$ | kernel matrix with $\phi(x_i)\cdot\phi(x_j)$ corresponding to its element $(i,j)$ |
| $X$ | instance matrix with each row representing one instance |
| $W$ | (N+1)-column hyperplane matrix for target task. Each column represents one hyperplane of a binary model |
| $W'$ | hyperplane matrix for the source task |
| $a'$ | the Lagrangian multiplier matrix for source problem. Each column represents a set of |
| $a$ | the Lagrangian multiplier matrix for target problem |
| $b', b$ | the bias vector for source and target task |
| $a_i, w_i$ | $i_{th}$ column of matrix $a$ and $w$ |
| $d_\gamma$ | diagonal matrix with$[\gamma_1,...,\gamma_N]$ in its main diagonal |
| $\beta$ | row vector $[\beta_1,...,\beta_N]$ |
| $\varepsilon_{ny_i}$ | $\varepsilon_{ny_i}=1$ if $n=y_i$ and 0 otherwise |

Assume that, for our $(N+1)$-category target task, $x\in\mathcal{X}$ and $y\in\mathcal{Y}=\{1,2,...,N+1\}$ are the input vector and output for the learning task respectively. Meanwhile, we have a set of binary linear classifiers $f'_n(x)=\phi(x)w'_n+b'_n$, for $n=1,...,N$ trained from an unknown distribution with One-Versus-All (OVA) strategy. Now we want to learn a set of new classifier $f_n(x)=\phi(x)w_n+b_n, n=1,...,N+1$, so that example $x$ is assigned to the category $j$ if $j\equiv\arg\max_{n=1,...,N+1}\{f_n(x)\}$. In LS-SVM, the solution of the model parameters $(w_n,b_n)$ can be found by solving the following optimization problem:

$$\textbf{min}\quad R(w_n)+\frac{C}{2}\sum_{i}^{l}(Y_{i,n}-\phi(x_i)w_n-b_n)^2$$

Where $R(w_n)$ is the regularization term to guarantee good generalization performance and avoid overfitting. $\mathbf{Y}$ is a encoded label matrix so that $Y_{i,n}=1$ if $y_i=n$ and $-1$ otherwise.

In classic LS-SVM setting, the regularization term is set to $\frac{1}{2}\|w_n\|^2$ and the optimal $w_n=\phi(X)^T\alpha_n$ while the parameters $(\aleph_n,b_n)$ can be found by solving

$$\begin{bmatrix} K(X,X)+\frac{1}{C}\mathrm{I} & \mathbf{1} \\ \mathbf{1^T} & 0 \end{bmatrix}\begin{pmatrix}\alpha_n\\ b_n\end{pmatrix}=\begin{pmatrix}Y_n\\0\end{pmatrix} \tag{1}$$

Here $I$ is the identity matrix and $\mathbf{1}$ is a column vector with all its elements equal to 1.

Now our task can be divided into two separate part: learning the $N$ overlapped categories and the new category. We know that the source and target share $N$ categories. From previous work [5], the regularization term can be written as $\frac{1}{2}\|w_n-\gamma_n w'_n\|^2$. Here, $\gamma_n$ is the regularization parameter controlling the amount of transfer. For the task for new category, we can use multi-source kernel learning strategy in [2]

So the multi-class transfer problem can be solved by optimizing the following objective function:

$$\begin{aligned}\textbf{min}\quad & \frac{1}{2}\sum_{n=1}^{N}\|w_n-\gamma_n w'_n\|^2+\frac{1}{2}\left\|w_{N+1}-\sum_{k=1}^{N}w'_k\beta_k\right\|^2\\ & \frac{C}{2}\sum_{n=1}^{N+1}\sum_{i=1}^{l}e_{i,n}^2\\ \textbf{subject to}\quad & e_{i,n}=Y_{i,n}-\phi(x_i)w_n-b_n\end{aligned} \tag{2}$$

The closed-form of the optimal solution to Eq. (2) is:

$$\begin{aligned} w_n&=\gamma_n w'_n+\sum_{i}^{l}\alpha_{in}\phi(x_i) \qquad n=1,...,N\\ w_{N+1}&=\sum_{k}^{N}\beta_k w'_k+\sum_{i}^{l}\alpha_{i(N+1)}\phi(x_i)\end{aligned}$$

Here $\alpha_{ij}$ is the element $(i,j)$ in $\alpha$.

Let $\psi$ denotes the first term of left-hand side in Eq. (1). closed form in matrix format

$$\begin{bmatrix}\alpha\\ b\end{bmatrix}=\begin{bmatrix}\alpha'\\ b'\end{bmatrix}-\begin{bmatrix}\alpha''\\ b''\end{bmatrix}\begin{bmatrix}d_\gamma & \beta^T\\ 0 & 0\end{bmatrix} \tag{3}$$

From Eq. (3) we can see that, the solution of Eq. (1) is completed once $\gamma=[\gamma_1,...,\gamma_N]$ and $\beta$ are set.

### B. Optimize $\gamma$ and $\beta$

introduce LOO error estimation

introduce our objective function

Let us call $\xi_i$ the loss of our multi-class prediction for example $x_i$ and $\xi_i$ can be defined as [13]:

$$\xi_i(\gamma,\beta)=\max_{n\in\{1,...,N+1\}}\left[1-\varepsilon_{ny_i}+\hat{Y}_{in}(\gamma,\beta)-\hat{Y}_{iy_i}(\gamma,\beta)\right] \tag{4}$$

Where $\varepsilon_{ny_i}=1$ if $n=y_i$ and 0 otherwise. $\xi_i(\gamma,\beta)>0$ if example $x_i$ is misclassified. The intuition behind this loss function is to enforce the distance between the true class and other classes to be at least 1.

And we define our objective function as:

$$\begin{aligned}\textbf{min}\quad & \frac{\lambda_1}{2}\sum_{j=1}^{N}\|\gamma_j\|^2+\frac{\lambda_2}{2}\sum_{j=1}^{N}\|\beta_j\|^2+\sum_{i=1}^{l}\xi_i\\ \textbf{subject to}\quad & 1-\varepsilon_{ry_i}+\hat{Y}_{ir}(\gamma,\beta)-\hat{Y}_{iy_i}(\gamma,\beta)\le\xi_i;\\ & \lambda_1,\lambda_2\ge 0\end{aligned} \tag{5}$$

explanation of two parameters, further refer to [13]

By adding a dual set of variables, one for each constraint, we get the Lagrangian of the optimization problem:

$$\begin{aligned}\textbf{max}\quad & L(\gamma,\beta,\xi,\eta)=\frac{\lambda_1}{2}\sum_{j=1}^{N}\|\gamma_j\|^2+\frac{\lambda_2}{2}\sum_{j=1}^{N}\|\beta_j\|^2+\sum_{i=1}^{l}\xi_i\\ & +\sum_{i,r}\eta_{i,r}\left[1-\varepsilon_{ry_i}+\hat{Y}_{ir}(\gamma,\beta)-\hat{Y}_{iy_i}(\gamma,\beta)-\xi_i\right]\\ \textbf{subject to}\quad & \forall i,r\quad \eta_{i,r}\ge 0\end{aligned} \tag{6}$$

Algorithm for optimizaiton, convex-non-differentiable

**Algorithm 1** $\gamma$ optimization

**Input:** $\psi, \alpha', \alpha'', T, \psi,$
**Output:** $\gamma = \{\gamma^1, ..., \gamma^n\}, \beta$

$\beta \leftarrow 0, \gamma \leftarrow 1$
**for** $iter = 1$ to $T$ **do**
  $\hat{Y} \leftarrow Y - (\psi \circ I)^{-1} \left(\alpha' - \left[\begin{array}{cc} \alpha'' d_\gamma & \alpha'' \beta^T \end{array}\right]\right)$
  $\Delta_\gamma = 0, \Delta_\beta = 0$
  **for** $i = 1$ to $l$ **do**
    $\Delta_\gamma \leftarrow \Delta_\gamma + \lambda_1 \gamma$
    $\Delta_\beta \leftarrow \Delta_\beta + \lambda_2 \beta$
    **for** $r = 1$ to $N + 1$ **do**
      $l_{ir} = 1 - \varepsilon_{y_i r} + \hat{Y}_{ir} - \hat{Y}_{iy_i}$
      **if** $l_{ir} > 0$ **then**
        **if** $y_i, r \in \{1, ..., N\}$ **then**
          $\Delta_\gamma^{y_i} \leftarrow \Delta_\gamma^{y_i} - \frac{\alpha''_{iy_i}}{\psi_{ii}^{-1}}$
          $\Delta_\gamma^{r} \leftarrow \Delta_\gamma^{r} + \frac{\alpha''_{ir}}{\psi_{ii}^{-1}}$
        **else if** $y_i = N + 1$ **then**
          $\Delta_\beta \leftarrow \Delta_\beta - \frac{\alpha''_i}{\psi_{ii}^{-1}}$
          $\Delta_\gamma^{r} \leftarrow \Delta_\gamma^{r} + \frac{\alpha_{ir}}{\psi_{ii}^{-1}}$
        **else**
          $\Delta_\gamma^{y_i} \leftarrow \Delta_\gamma^{y_i} - \frac{\alpha''_{iy_i}}{\psi_{ii}^{-1}}$
          $\Delta_\beta \leftarrow \Delta_\beta + \frac{\alpha''_i}{\psi_{ii}^{-1}}$
        **end if**
      **end if**
    **end for**
  **end for**
  $\beta \leftarrow \beta - \frac{\Delta_\beta}{l \times iter}$
  $\gamma \leftarrow \gamma - \frac{\Delta_\gamma}{l \times iter}$
**end for**

### C. Analysis

Convergence analysis

Superior bound analysis

*Theorem 1:* Assume that $\bar{\xi}_i$ is the multi-class loss of example $x_i$ when $\gamma = \beta = \mathbf{0}$. Let $\gamma^*, \beta^*$ be the optimal solution for Eq. (6) and $\xi_i^*$ be the multi-class loss with respective to example $x_i$. Then for every example $x_i \in \mathcal{X}$, we have:

$$\sum_i \xi_i \leq \sum_i \bar{\xi}_i$$

*Proof:* When $\gamma = \beta = \mathbf{0}$, from Eq. (4) we can get:

$$\bar{\xi}_i = \max_n \left[1 - \varepsilon_{ny_i} + \frac{(\alpha'_{iy_i} - \alpha'_{in})}{\psi_{ii}^{-1}}\right]$$

To obtain the optimal value of $\gamma$ and $\beta$, we have to seek the saddle point of the Lagrangian problem in (6) by finding the minimum for the prime variables $\{\gamma, \beta, \xi\}$ and the maximum for the dual variables $\eta$. To find the minimum of the primal problem, we require:

$$\frac{\partial L}{\partial \xi_i} = 1 - \sum_n \eta_{in} = 0 \rightarrow \sum_n \eta_{in} = 1$$

Similarly, for $\gamma$ and $\beta$, we require:

$$\begin{aligned} \frac{\partial L}{\partial \gamma_n} &= \lambda_1 \gamma_n + \sum_i \eta_{in} \theta_{in} - \sum_{i, n=y_i} \left(\sum_q \eta_{iq}\right) \theta_{in} \gamma_n \\ &=_1 \lambda_1 \gamma_n + \sum_i \eta_{in} \theta_{in} - \sum_i \varepsilon_{ny_i} \theta_{in} = 0 \\ &\Rightarrow \gamma_n^* = \frac{1}{\lambda_1} \sum_i \left(\varepsilon_{ny_i} - \eta_{in}\right) \theta_{in} \end{aligned} \tag{7}$$

In $=_1$ we use the facts that $\sum_n \eta_{in} = 1$ and use $\varepsilon_{ny_i}$ to replace it.

$$\begin{aligned} \frac{\partial L}{\partial \beta_n} &= \lambda_2 \beta_n + \left[\sum_{i,n} \frac{\eta_{in} \alpha''_{in}}{\psi_{ii}^{-1}} \left(\delta_n - \delta_{y_i}\right)\right] = 0 \\ &\Rightarrow \beta_n^* = \frac{1}{\lambda_2} \sum_{i,n} \frac{\eta_{in} \alpha''_{in}}{\psi_{ii}^{-1}} \left(\delta_{y_i} - \delta_n\right) \end{aligned} \tag{8}$$

As the strong duality holds,the primal and dual objectives coincide. Plug Eq (7) and (8) into Eq. (6), we have:

$$\sum_{i,n} \eta_{in} \left[1 - \varepsilon_{ny_i} + \hat{Y}_{in}\left(\gamma^*, \beta^*\right) - \hat{Y}_{iy_i}\left(\gamma^*, \beta^*\right) - \xi_i^*\right] = 0$$

Expand the equation above, we have:

$$\begin{aligned} &\sum_{i,n} \eta_{in} \left[1 - \varepsilon_{n,y_i} + \frac{(\alpha'_{iy_i} - \alpha'_{in})}{\psi_{ii}^{-1}} - \xi_i\right] \\ &= \lambda_1 \sum_r \left\|\gamma_r^*\right\|^2 + \lambda_2 \sum_r \left\|\beta_r^*\right\|^2 \geq 0 \end{aligned}$$

Rearranging the above, we obtain:

$$\begin{aligned} &\sum_{i,n} \eta_{in} \left[1 - \varepsilon_{n,y_i} + \frac{(\alpha'_{iy_i} - \alpha'_{in})}{\psi_{ii}^{-1}}\right] \\ &\geq \sum_{i,n} \eta_{in} \xi_i = \sum_i \xi_i \end{aligned} \tag{9}$$

The left-hand side of Inequation (9) can be bounded by:

$$\begin{aligned} &\sum_{i,n} \eta_{in} \left[1 - \varepsilon_{ny_i} + \frac{(\alpha'_{iy_i} - \alpha'_{in})}{\psi_{ii}^{-1}}\right] \\ &\leq \sum_i \left(\sum_n \eta_{in} \max_r \left\{1 - \varepsilon_{ry_i} + \frac{(\alpha'_{iy_i} - \alpha'_{ir})}{\psi_{ii}^{-1}}\right\}\right) \\ &= \sum_i \left(\sum_n \eta_{in} \bar{\xi}_i\right) = \sum_i \bar{\xi}_i \end{aligned} \tag{10}$$

■

When setting $\gamma = \beta = \mathbf{0}$, we don't utilize any knowledge from previous task (see Eq. (2)). From Theorem 1 we can conclude our method can always outperform the method learning directly.

discuss $\lambda$

## IV. Experiment

### A. Dataset

### B. Baselines

We compare our algorithm with two kinds of baselines. The first one is methods without leveraging any prior knowledge (no transfer baselines). The second consists of some methods with transfer techniques. Here are the no transfer baselines **No transfer:**

**Source+1:**

### C. transfer from good oracle

### D. from bad oracle

### E. mixed

## V. Conclusion

## Acknowledgment

## References


[1] S. J. Pan and Q. Yang, "A survey on transfer learning," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 22, no. 10, pp. 1345–1359, 2010.

[2] T. Tommasi, F. Orabona, and B. Caputo, "Learning categories from few examples with multi model knowledge transfer," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 5, pp. 928–941, 2014.

[3] J. J. Lim, "Transfer learning by borrowing examples for multiclass object detection," Ph.D. dissertation, Massachusetts Institute of Technology, 2012.

[4] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France*, 2015, pp. 97–105.

[5] J. Yang, R. Yan, and A. G. Hauptmann, "Cross-domain video concept detection using adaptive svms," in *Proceedings of the 15th international conference on Multimedia*. ACM, 2007, pp. 188–197.

[6] Y. Aytar and A. Zisserman, "Tabula rasa: Model transfer for object category detection," in *Computer Vision (ICCV), 2011 IEEE International Conference on*. IEEE, 2011, pp. 2252–2259.

[7] I. Kuzborskij, F. Orabona, and B. Caputo, "From n to n+ 1: Multiclass transfer incremental learning," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 3358–3365.

[8] J. Davis and P. Domingos, "Deep transfer via second-order markov logic," in *Proceedings of the 26th annual international conference on machine learning*. ACM, 2009, pp. 217–224.

[9] X. Wang, T.-K. Huang, and J. Schneider, "Active transfer learning under model shift," in *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, 2014, pp. 1305–1313.

[10] T. Zhou and D. Tao, "Multi-task copula by sparse graph regression," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 771–780.

[11] T. Tommasi, F. Orabona, and B. Caputo, "Safety in numbers: Learning categories from few examples with multi model knowledge transfer," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*. IEEE, 2010, pp. 3081–3088.

[12] G. C. Cawley, "Leave-one-out cross-validation based model selection criteria for weighted ls-svms," in *Neural Networks, 2006. IJCNN'06. International Joint Conference on*. IEEE, 2006, pp. 1661–1668.

[13] K. Crammer and Y. Singer, "On the algorithmic implementation of multiclass kernel-based vector machines," *The Journal of Machine Learning Research*, vol. 2, pp. 265–292, 2002.

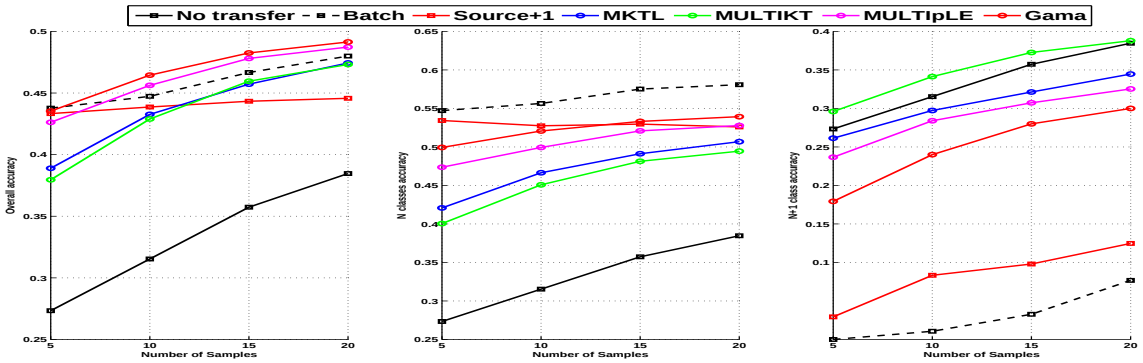


Fig. 1. Acc for good oracle

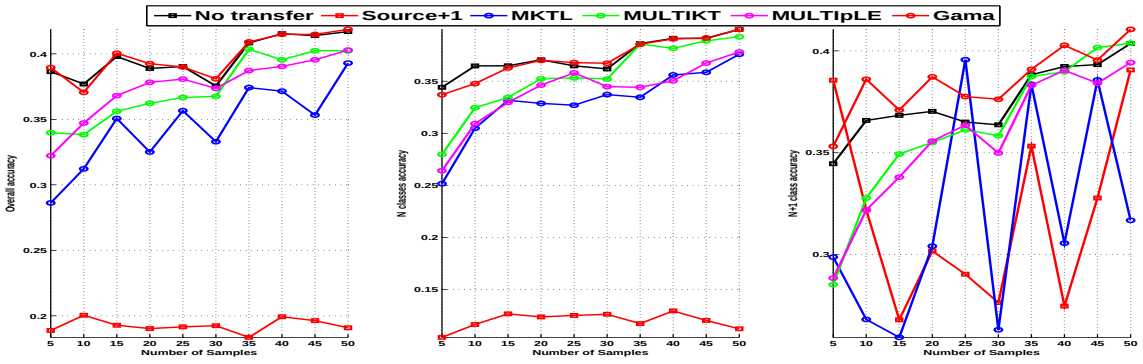


Fig. 2. Acc for bad oracle