Domain-Invariant Transfer Kernel Learning

Md Enayat Ullah[1] and Abheet Aggarwal[2]
Guide: Dr. Harish Karnick

Indian Institute of Technology Kanpur,
,

Abstract. Domain invariant learning generalizes a learning model built on training data having different distribution than testing data and still can be accurate. A general principle to tackle this problem is reducing the distribution difference between training data and testing data such that the generalization error can be bounded. Transfer Kernel Learning(TKL) learns the domain invariant kernel by matching target and source distributions in the reproducing kernel Hilbert space (RKHS). A family of spectral kernels is then designed by extrapolating target eigensystem on source samples with Mercers theorem and the hyperparameters are evaluated minimizing the approximation error. Experiments are done on various datasets ,discussed below, and the results clearly shows a marked increase in performance when compared to traditional SVMs.

1 Motivation

Statistical learning theory guarantees the generalization error bound for standard supervised learning, where training data and testing data are sampled from identical probability distribution.[1] However in this new era of Big data , we have huge amount of data which is heterogeneous in nature coming from different domains such as texts, videos and images, has created a compelling requirement for statistical learning models to be adaptive across different distributions. It has been shown that when standard supervised classifiers are evaluated outside of their training datasets, the performance drops significantly.[2]

1.1 Related Work

The foremost intuition to tackle this type of problem is to first quantize the distribution discrepancy between the datasets. There have been many works focusing explicitly on minimizing the distribution discrepancy via parametric or non-parametric divergence. The parametric distribution discrepancy can be formalized by the Kullback-Leibler (KL) divergence, Bregman divergence among others. Below is the expression of KL divergence:

$$D_{KL}(P,Q) = \int_{-\infty}^{\infty} p(x) \log \frac{p(x)}{q(x)} dx$$

However, the problem one encounters in the above as well as in Bergman divergence is the requirement of density estimation which makes these divergences difficult to evaluate. The other non-paramtetric way of computing the distribution discrepancy is Maximum-Mean Discrepancy(MMD). However, methods involving joint minimization of empirical risk and MMD involve an intermediate Semi-definite programming(SDP) step, a step and thus computationally intractable.

To alleviate the need of computing the distribution discrepancy, another line of recent works have been concentrating on directly learning a domain-invariant kernel matrix. Multiple Kernel Learning(MKL) is an ensemble technique which calculates the hyperparamters taken on a linear combination of pre-computed kernels.[3] The problem with such a method is that the pre-computed kernels are incapable of capturing all kinds of variations in data and thus may be inadequate for correcting the distribution mismatch. To this end, Zhang et al. proposed a surrogate kernel matching (SKM) approach to directly match training data and testing data in a reproducing kernel Hilbert space (RKHS).[4] But, a major limitation of SKM is that it linearly transforms the complete source kernel to the eigenspace of target kernel, which not only leads to large approximation error, but also in incapable to be used in case of no-linear maps.

2 Introduction

Transfer Learning is an umbrella term which encompasses methods which are adaptive across domains. The two domains: source($\mathcal{Z}$) and target($\mathcal{X}$) are taken from different distributions, and building a model solely on the source gives poor results. In the work, the method of Zhang et al. is improved upon to a transfer kernel learning (TKL) approach, which learns a domain-invariant kernel by directly matching the source and target distributions in the reproducing kernel Hilbert space.[5] Instead of using the source kernel matrix, eigenvectors of the target kernel matrix are used to extrapolate the source kernel matrix using Nystrom Approximation method. So, essentially, what is done is we model the kernel matrix of source using the information of how the target points are distributed(eigenvectors).

The approximation is then relaxed using spectral kernel design to include hyperparameters which are obtained after minimizing the error between the extrapolation and ground truth source kernel matrix. The domain invariant kernel thus computed is plugged in standard kernel machines. We used it modify Support Vector Machines(SVM) and tested it for classification in benchmark cross-domain text and image datasets.

3 Mathematical Preliminaries

3.1 Maximum Mean Discrepancy(MMD)

A non-parametric divergence, maximum mean discrepancy compares two distributions p and q based on the distance between the expectations of the two generated datasets $\mathcal{X} = x_1, x_2, ..., x_n$ and Z =z1,z2,.....,zm in a reproducing kernel Hilbert space H.

$$MMD(p,q) \triangleq \sup_{\|f\|_{\mathcal{H}} \leq 1} (\mathbb{E}_{x\sim p}[f(x)] - \mathbb{E}_{z\sim q}[f(z)])$$

$$MMD(\mathcal{X},\mathcal{Z}) \triangleq \left\| \frac{1}{n}\sum_{i=1}^{n}\phi(x_i) - \frac{1}{m}\sum_{j=1}^{m}\phi(z_j) \right\|_{\mathcal{H}}$$

Theorem: Let p and q be probability measures and $\mathcal{H}$ be a universal RKHS, then $MMD(p,q) = 0$ iff $p = q$.

By nonlinear mapping ϕ, MMD can intrinsically capture both first- and high-order statistics. Hence , MMD can capture the proper distribution of the domain and its convergence will not make only means to be equal rather it will make all the moments to be equal in both the domains and hence it will make both the distributions in different domains to converge.

3.2 Nystrom Approximation

Mercers Theorem:$\mathcal{X}$ and $\mathcal{Z}$ are identically distributed. Let $k(z,x)$ be a continuous symmetric non-negative function which is positive semi-definite and square integrable w.r.t. distribution $p(x)$, then

$$k(z,x) = \sum_{i=1}^{\infty}\lambda_i\phi_i(z)\phi_i(x)$$

The eigenvalues $\lambda_i' s$ and orthonormal eigenfunctions $\phi_i' s$ are the solutions of:

$$\int k(z,x)\phi_i(x)p(x)dx = \lambda_i\phi_i(z)$$

Nystrom Method is essentially a quadrature formula which approximates the above integral of Mercers theorem by a representative weighted sum, which is expressed as follows[6]:

$$\sum_{j=1}^{n}\frac{k(z,x_j)\phi_i(x_j)}{n} \simeq \lambda_i\phi_i(z)$$

The fundamental theorem of the theory of reproducing kernel Hilbert space states that if the target kernel is known, then it's eigensystem can be used to reconstruct any positive semi-definite (PSD) kernel and provides a mechanism to generate kernel matrices on any new datasets.

3.3 Spectral Kernel Design

Theorem:If a positive semi-definite kernel matrix $\mathbb{K} \in \mathbb{R}_{n\times n}$ has eigensystem $\{\gamma_i, \phi_i\}_{i=1}^n, \gamma_1 \geq ... \geq \gamma_n \geq 0$, then the family of matrices

$$K_\lambda = \sum_{i=1}^{n} \lambda_i \phi_i \phi_i^T, \lambda_1 \geq ... \geq \lambda_n \geq 0$$

will produce PSD kernels with K_λ as kernel matrices.

Spectral kernel design constructs new kernels from the eigenvectors of the target kernel. The reconstructed new kernel matrix is a linear combination of multiple simple kernel.[7] This new kernel matrix is not same as that of multiple kernel learning(MKL) matrix approach as in MKL the kernels are precomputed, but here the kernels of which the new kernel is a linear combination are all computed from the eigenvectors of the target kernel.

4 Approach

4.1 Problem Formulation

Domain:
A domain $\mathcal{D}$ is composed of an d-dimensional feature space $\mathcal{F}$ and a marginal probability distribution $P(x)$ i.e. $\mathcal{D} = \{\mathcal{F}, P(x)\}, x \in \mathcal{F}$.

Transfer Kernel Learning:
Given a labeled source domain $\mathcal{Z} = \{(z_1, y_1), ...(z_m, y_m)\}$ and an unlabeled target domain $\mathcal{X} = \{x_1, ..., x_n\}$ with $\mathcal{F}_\mathcal{Z} = \mathcal{F}_\mathcal{X}, \mathcal{Y}_\mathcal{Z} = \mathcal{Y}_\mathcal{X}$, we learn a domain-invariant kernel $k(z, x) = \langle \phi(z), \phi(x) \rangle$ such that $P(\phi(z)) \simeq P(\phi(x))$. Assuming $P(y|\phi(z)) \simeq P(y|\phi(x))$, so that kernel machines trained on $\mathcal{Z}$ generalize well on $\mathcal{X}$.

However, handling $\phi(.)$ in Hilbert space is not trivial since in most cases they cannot be explicitly represented. As discussed above, we conjecture $P(\phi(x)) \simeq P(\phi(z))$ which equivalently leads to $K_\mathcal{X} \simeq K_\mathcal{Z}$[4]

Equating the kernel matrices is also a problem because a). the kernel matrices are data dependent and empirical evidence doesn't really tell us everything about the distribitiom and b)$\mathcal{X}$ and $\mathcal{Z}$ may not have the same dimensions, and we cannot evaluate the closeness between two different dimensional matrices.same dimensions. To solve this problem , we use Nystrom Kernel Approximation which gives an extrapolated source kernel matrix calculated from the eigensystem of the target kernel matrix.

4.2 Eigensystem extrapolation (Nystrom Kernel Approximation)

The preliminary covers some literature of Nystrom Approximation, wherein the integral in Mercer's Theorem is approximated using a quadrature formula:

$$\sum_{j=1}^{n} \frac{k(z, x_j)\phi_i(x_j)}{n} \simeq \lambda_i \phi_i(z)$$

Taking $z \in \mathcal{X}$ in the matricized form of the above equation, we get:

$$K_{\mathcal{X}} \Phi'_{\mathcal{X}} = \Phi'_{\mathcal{X}} \Lambda_{\mathcal{X}}$$

This gives us the standard eigendecomposition of $\mathcal{X}$:

$$K_{\mathcal{X}} \Phi_{\mathcal{X}} = \Phi_{\mathcal{X}} \Lambda_{\mathcal{X}}$$

Comparing the two equations, we come up with a way to extrapolate any point $z \in \mathcal{Z}$ using the eigensystem of $\mathcal{X}$

$$\overline{\Phi}_{\mathcal{Z}} \simeq K_{\mathcal{Z}\mathcal{X}} \Phi_{\mathcal{X}} \Lambda^{-1}$$

where $K_{\mathcal{Z}\mathcal{X}} \in \mathbb{R}^{m \times n}$ is the cross-domain similarity matrix. The extrapolated source kernel matrix is then evaluated using the extrapolated eigenvectors as:

$$\overline{K}_{\mathcal{Z}} \simeq \overline{\Phi}_{\mathcal{Z}} \Lambda_{\mathcal{X}} \overline{\Phi}^T_{\mathcal{Z}} \simeq \overline{\Phi}_{\mathcal{Z}} \Lambda_{\mathcal{X}} \overline{\Phi}^T_{\mathcal{Z}}$$

However, since Nystrom Approximation is a consequence of Mercer's Theorem which is based on the assumption that $\mathcal{X}$ and $\mathcal{Z}$ are identically distributed, the Nystrom Approximation error is inevitably high. This implies that Nystrom Approximation Error(NAE) essentially embodies MMD, and minimizing NAE is same as minimizing MMD.

4.3 Eigenspectrum Relaxation

The above problem is tackled using Spectral Kernel Design to construct new kernel matrix from the extrapolated eigensystem from above. Specifically, we relax the eigenvalues Λ_X to learnable parameters Λ so as to obtain a family of spectral kernels extrapolated from target kernel but evaluated on source data.This new eigensystem preserves the structure of the target kernel and yet can minimize the error by calculating Λ.

The extrapolated relaxed kernel matrix is: $\overline{K}_{\mathcal{Z}} = \overline{\Phi}_{\mathcal{Z}} \Lambda \overline{\Phi}^T_{\mathcal{Z}}$

4.4 Approximate Error Minimization

Now, we minimize the approximation error between the ground truth kernel matrix $K_{\mathcal{Z}}$ and the extrapolated kernel matrix $\overline{K}_{\mathcal{Z}}$ using squared loss of Frobenius norm :

$$\min_{\Lambda} \left\| \overline{K}_{\mathcal{Z}} - K_{\mathcal{Z}} \right\|_{\mathcal{F}}^2 = \left\| \overline{\Phi}_{\mathcal{Z}} \Lambda \overline{\Phi}_{\mathcal{Z}}^T - K_{\mathcal{Z}} \right\|_{\mathcal{F}}^2$$

$$\lambda_i \geq \zeta \lambda_{i+1}, i = 1, ...n-1$$

$$\lambda_i \geq 0, i = 1, ...n$$

The damping factor $\zeta(DampingFactor) \geq 1$ is constrained so as to capture the property of positive-semi definite matrices that the eigenvalues follow a power law distribution, and also to to ensure that the larger eigenvalues contributes more to knowledge transfer.

5 Implementation

With a bit of Algebra, we matricize the formulation to get the following Quadratic Programming(QP) problem:

$$\min_{\lambda} (\lambda^T \mathbf{Q} \lambda - 2\mathbf{r}^T \lambda)$$

$$\mathbf{C}\lambda \geq \mathbf{0}$$

$$\lambda \geq \mathbf{0}$$

Where:

$$\mathbf{Q} = (\overline{\Phi}_{\mathcal{Z}}^T \overline{\Phi}_{\mathcal{Z}}) \circ (\overline{\Phi}_{\mathcal{Z}}^T \overline{\Phi}_{\mathcal{Z}}^T)$$

$$\mathbf{r} = diag(\overline{\Phi}_Z^T \overline{\Phi}_Z^T)$$

$$\mathbf{C} = \mathbf{I} - \zeta \bar{\mathbf{I}}$$

Real world data exhibit the eigen-gap property, wherein largest r eigenvectors are much larger than the remaining ones and hence it is unnecessary to compute the full eigensystem. Heuristically, we take $r = min(500, n)$. Take $\lambda \in \mathbb{R}^{r \times 1}$ The above QP effectively solves for eigenspectrum hyperparameters Λ. Moreover, with regards to tuning the parameters, apart from kernel hyperparamaters and C, the eigenspectrum damping factor (ζ) is the other tunable parameter in Transfer Kernel Learning(TKL) .

5.1 Application in Support Vector Machines

The support vector machines, being a kernalizable algorithm encompasses inner products in the model building(training) as well as in the testing phase. The kernel matrix in the training procedure is the source kernel matrix $K_{\mathcal{Z}} = \Phi_{\mathcal{Z}} \Lambda_{\mathcal{Z}} \Phi_{\mathcal{Z}}^T$, which is now replaced with the extrapolated source kernel matrix $\overline{K}_{\mathcal{Z}} = \overline{\Phi}_{\mathcal{Z}} \Lambda \overline{\Phi}_{\mathcal{Z}}^T$

The testing expression in SVM include the kernel matrix $K_{\mathcal{Z}\mathcal{X}}$,which replaced with domain invariant kernel $\overline{K}_{\mathcal{X}\mathcal{Z}} = \Phi_{\mathcal{X}} \Lambda \overline{\Phi}_{\mathcal{Z}}^T$ gives the following final expression:

$$y_{\mathcal{X}} = \overline{K}_{\mathcal{X}\mathcal{Z}} (\alpha \circ y_{\mathcal{Z}}) + b$$

5.2 Scalable Implementation

The usual implementation is computationally prohibitive and thus not viable to be applied on large datasets. Since source or target is taken from a single domain/distribution, Nystrom method can further be used to approximate large matrices from a small sample of data. That small sample($n << N$ and $m << M$) implies that $K_{\mathcal{X}\hat{\mathcal{X}}} \in \mathbb{R}^{n \times \hat{n}}$ and $K_{\mathcal{Z}\hat{\mathcal{Z}}} \in \mathbb{R}^{m \times \hat{m}}$. Eigendecomposition of $K_{\hat{\mathbb{X}}}$ gives the following(extrapolation within domain):

$$\Phi_{\mathcal{X}} \simeq K_{\mathcal{X}\hat{\mathcal{X}}} \Phi_{\hat{\mathcal{X}}} \lambda_{\hat{\mathcal{X}}}^{-1}$$

Similar operations on the source dataset followed by cross domain extrapolation of source eigensystem using the target eigenvectors(self-extrapolated) gives:

$$\overline{\Phi}_{\hat{\mathcal{Z}}} \simeq K_{\hat{\mathcal{Z}}\mathcal{X}} \Phi_{\mathcal{X}} \lambda_{\hat{\mathcal{X}}}^{-1}$$

$$\overline{\Phi}_{\mathcal{Z}} \simeq K_{\mathcal{Z}\hat{\mathcal{Z}}} \overline{\Phi}_{\hat{\mathcal{Z}}} \Lambda_{\hat{\mathcal{X}}}^{-1}$$

$$K_{\mathcal{Z}} \simeq K_{\mathcal{Z}\hat{\mathcal{Z}}} K_{\hat{\mathcal{Z}}}^{-1} K_{\hat{\mathcal{Z}}\mathcal{Z}}$$

6 Computational Complexity

Algorithm: Transfer Kernel Learning	
Compute $\mathcal{K}_{\mathcal{Z}}, \mathcal{K}_{\mathcal{X}}, \mathcal{K}_{\mathcal{Z}\mathcal{X}}$ by kernel k	$O(d(m+n)^2)$
Eigendecompose $\mathcal{K}_{\mathcal{X}}$ for $\{\Lambda_{\mathcal{X}}, \Phi_{\mathcal{X}}\}$	$O(rn^2)$
Extrapolate for source eigensystem $\overline{\Phi}_{\mathcal{Z}}$	$O(rmn)$
Solve QP problem for eigenspectrum λ	$O(rn^2 + r^3)$

The first line just does matrix multiplications of the order $O(n^2), O(m^2) and O(mn)$. The second line computes r eigenvectors each taking a computational time

$O(n^2)$. In the same way, all the other calculations can be evaluated. Overall Complexity of the algorithm: $O(d+r)(m+n)^2$.

7 Approximate Error Analysis

Nystrom Approximation Error is expressed as:

$$\epsilon_{Nys} = \|K_{\mathcal{Z}} - K_{\mathcal{Z}\mathcal{X}} K_{\mathbb{X}}^{-1} K_{\mathcal{X}\mathcal{Z}}\|_{\mathcal{F}}$$

Error in the Transfer Kernel Learning procedure:

$$\epsilon_{tkl} = \|\overline{\Phi}_{\mathcal{Z}} \Lambda \overline{\Phi}_{\mathcal{Z}}^{T} - K_{\mathcal{Z}}\|_{\mathcal{F}}$$

Following is the final result we get for theoretical error analysis:

$$\epsilon_{TKL} \leq \epsilon_{Nys} \leq 4m \sqrt[2]{C_k mn\epsilon} + C_k mn\epsilon \|K_{\mathcal{X}}^{-1}\|_{\mathcal{F}}$$

The first inequality is obvious following from the consequence that we ϵ_{Nys} is minimized for hyperparamters to give a minimum ϵ_{tkl}. The second inequality is the result of the work by Jin et al. wherein he proved the bound on Nystrom Approximation Error[6].

8 Dataset

We evaluate the implementation on text as well as image datasets.

8.1 Text

Two benchmark datasets were taken to evaluate the model- 20 newsgroup and reuters. That data is essentially a bag of words model each having the counts of words, processed using TF-IDF. Both the datasets were further divided into subcategories: 20-newsgroup into four categories: rec, sci, main and talk. Each of these were further sub-divided into categories which are actually different domains the data is taken from. Reuters data is composed similarly from 4-5 categories, further sub divided into domains from different data. The source is compiled by taking two categories from one primary category and two from other. The target is constructed similarly, and it becomes a binary classification problem between the two categories.

8.2 Image

Image datasets are obtained from the following sources: Amazon(A), Webcam(W) and Caltech(C). Each of them is further divided into sub-categories. SURF features are taken for the image datasets, and a multilabel classification problem is constructed across any two of the domains.

9 Results

9.1 Text dataset

Tuned Parameters: $C = 10, \zeta = 2$

Dataset	SVM	TKL
orgs vs people	69.24	76.40
orgs vs place	63.71	75.11
comp vs rec	85.51	92.44
comp vs sci	74.23	86.42
Out-sample	67.20	80.23

Table 1: Accuracies

9.2 Image dataset

Tuned Parameters: $C = 1.2, \zeta = 4$

Dataset	SVM	TKL
Amazon vs Caltech	62.90	69.80
Amazon vs Webcam	53.71	54.14
Caltech vs Webcam	48.43	57.45
Out-sample	50.23	54.64

Table 2: Accuracies

9.3 Plots

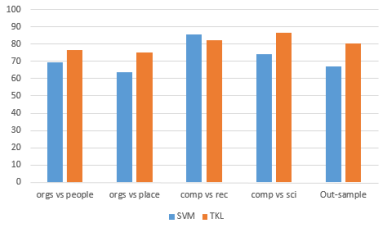

Fig. 1. Text Dataset.

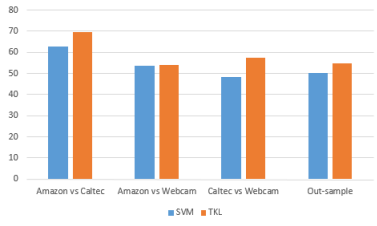

Fig. 2. Image Dataset.

10 Conclusion and Future Work

The results depict a marked increase in performance between traditional SVMs and SVMs equipped domain-invariant kernel(TKL). Also, when tested outside the data domain, TKL still outperforms the traditional SVMs. The aim of the project was to study and implement state of the art techniques in transfer kernel learning, and evaluate it against standard domain-irrespective methods. This was achieved emulating the work of Zhang, wherein the distribution mismatch between training and testing data is reduced in the RKHS using Nystrom method. The domain-invariant kernels were plugged into an SvM and were evaluated on cross domain benchmark text and image datasets. The result shows a marked increase in the accuracy of classification when compared to standard SVM. They also perform better than SVM in case of out-of-sample data points.

Future improvements to this involve modifying/removing the power-law imposition on the eigenvalues in the QP. We can test for non-power law distributions between the eigenvalues as well. Also, instead of taking the top r eigenvectors in every data(irrespective of size and dimensionality), we can consider that as a hyperparamter which can be further tuned with respect to a dataset.

References

1. V. N. Vapnik and V. Vapnik, *Statistical learning theory*, vol. 1. Wiley New York, 1998.
2. 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.
3. L. Duan, I. W. Tsang, and D. Xu, "Domain transfer multiple kernel learning," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 3, pp. 465–479, 2012.
4. K. Zhang, V. Zheng, Q. Wang, J. Kwok, Q. Yang, and I. Marsic, "Covariate shift in hilbert space: A solution via sorrogate kernels," in *Proceedings of the 30th International Conference on Machine Learning*, pp. 388–395, 2013.
5. M. Long, J. Wang, J. Sun, and P. S. Yu, "Domain invariant transfer kernel learning," 2015.
6. K. Zhang, I. W. Tsang, and J. T. Kwok, "Improved nyström low-rank approximation and error analysis," in *Proceedings of the 25th international conference on Machine learning*, pp. 1232–1239, ACM, 2008.
7. S. J. Pan, X. Ni, J.-T. Sun, Q. Yang, and Z. Chen, "Cross-domain sentiment classification via spectral feature alignment," in *Proceedings of the 19th international conference on World wide web*, pp. 751–760, ACM, 2010.