Marginalized Denoising Autoencoder via Graph Regularization for Domain Adaptation

Yong Peng[1], Shen Wang[2], and Bao-Liang Lu[1,3,⋆]

[1] Center for Brain-Like Computing and Machine Intelligence,
Department of Computer Science and Engineering,
Shanghai Jiao Tong Unviersity, Shanghai 200240 China
[2] Department of Electrical Engineering and Computer Science,
University of Michigan, Ann Arbor, 48109 USA
[3] MoE-Microsoft Key Lab. for Intelligent Computing and Intelligent Systems,
Shanghai Jiao Tong Unviersity, Shanghai 200240 China
bllu@sjtu.edu.cn

Abstract. Domain adaptation, which aims to learn domain-invariant features for sentiment classification, has received increasing attention. The underlying rationality of domain adaptation is that the involved domains share some common latent factors. Recently neural network based on Stacked Denoising Auto-Encoders (SDA) and its marginalized version (mSDA) have shown promising results on learning domain-invariant features. To explicitly preserve the intrinsic structure of data, this paper proposes a marginalized Denoising Autoencoders via graph Regularization (GmSDA) in which the autoencoder based framework can learn more robust features with the help of newly incorporated graph regularization. The learned representations are fed into the sentiment classifiers and experiments show that the GmSDA can effectively improve the classification accuracy when comparing with some state-of-the-art models on the cropped Amazon benchmark data set.

Keywords: Domain Adaptation, Marginalized Denoising Autoencoder, Graph Regularization.

1 Introduction

Sentiment analysis [9] aims to determine the attitude of a speaker or a writer with respect to some topic or the overall contextual polarity of a document which is now a popular application; however, it often suffers from cross domain learning curse. To solve this problem, one solution is domain adaptation, which can build classifiers that are robust to mismatched distributions [1] [8] [12]. This presents a major difficult in adapting predictive models. Recent work has investigated techniques for alleviating the difference: instance re-weighting [8], sub-sampling from both domains [5] and learning joint target and source feature representations[4] [7] [12] [6].

⋆ Corresponding author.

M. Lee et al. (Eds.): ICONIP 2013, Part II, LNCS 8227, pp. 156–163, 2013.

Learning domain-invariant features is under the assumption that there is a domain invariant feature space [4] [2] [3] where the source and target domains have the same or similar marginal distributions and the posterior distribution of the labels are the same across domains. A deep model (SDA: stacked denoising autoencoders) was proposed in [7] to learn domain-invariant features. Denoising autoencoder [10] is a single layer neural network, whose output aims at reconstructing the partially corrupted input. Denoizers can be used as building block to construct deep architecture. The linearized version of SDA: marginalized stacked denoising autoencoder (mSDA)[6], is computational economy with a closed form solution and has few hyper-parameters to tune.

In real applications, the data is likely to reside on a low-dimensional ambient space. It has been shown that the geometrical information of the data is important for pattern recognition. Though deep model has show promising performance on domain adaption, it does not explicitly considers the intrinsic structure of data. To compensate this drawback and simultaneously harness the great power of feature learning of deep architecture, we propose a graph regularized marginalized SDA, which considers the local manifold structure of the data. The graph regularization term can be seen as a smooth operator for making the learned features vary smoothly along the geodesics of the data manifold.

The remainder of this paper is organized as follows. Brief review on mSDA is given in section 2. The proposed model, marginalized Denoising Autoencoders via graph Regularization (GmSDA), is introduced in section 3. Section 4 evaluates our method on a benchmark composed of reviews of 4 types of Amazon products and section 5 is conclusion.

Notation and Background. We assume the data originates from two domains, source S and target T. We samples data $\mathcal{D}_{\mathcal{S}} = \{\mathbf{x}_1, \cdots, \mathbf{x}_{n_S}\} \in \mathbb{R}^d$ with ground truth label $L_S = \{y_1, \cdots, y_{n_S}\}$. For target domain, only data without labels $\mathcal{D}_{\mathcal{T}} = \{\mathbf{x}_{n_S+1}, \cdots, \mathbf{x}_n\} \in \mathbb{R}^d$ are available. We do not assume that both use identical features and pad all input vectors with zeros to make both domain have same dimensionality d. The goal is to learn a classifier $h \in \mathcal{H}$ with labeled data $\mathcal{D}_{\mathcal{S}}$ and unlabeled data $\mathcal{D}_{\mathcal{T}}$ to predict labels T of data in $\mathcal{D}_{\mathcal{T}}$.

2 Marginalized Denoising Autoencoders

mSDA is a linearized version of SDA, in which the building block of mSDA is a single layer denoising autoencoder. Given data points $\mathcal{D} = \{\mathbf{x}_1, \cdots, \mathbf{x}_U\} \in \mathbb{R}^d$, where $\mathcal{D} = \mathcal{D}_{\mathcal{S}} \bigcup \mathcal{D}_{\mathcal{T}}$, corruption is applied to them by random feature removal. Then each feature has a probability p to be set to 0. Denote the corrupted version of $\mathbf{x}_i$ as $\tilde{\mathbf{x}}_i$. Reconstruction of corrupted input using mapping $\mathbf{W} : \mathbb{R}^d \to \mathbb{R}^d$ is equal to minimizing the squared reconstruction loss:

$$\frac{1}{2n} \sum_{i=1}^{n} ||\mathbf{x}_i - \mathbf{W}\tilde{\mathbf{x}}_i||^2. \tag{1}$$

We can incorporate the bias into the mapping $\mathbf{W} = [\mathbf{W}, \mathbf{b}]$ with slightly modifying the feature as $\mathbf{x}_i = [\mathbf{x}_i; 1]$. And we assume that the constant feature is

never corrupted. Considering a low variance, the m times passes over the input are implied to corrupt different feature each time. Then the problem becomes to solve the $\mathbf{W}$ which aims to minimize the overall squared loss:

$$\frac{1}{mn}\sum_{j=1}^{m}\sum_{i=1}^{n}||\mathbf{x}_i - \mathbf{W}\tilde{\mathbf{x}}||^2. \tag{2}$$

By defining the design matrix $\mathbf{X} = \{\mathbf{x}_1, \cdots, \mathbf{x}_n\} \in \mathbb{R}^{d\times n}$, its m-times repeated version as $\overline{\mathbf{X}} = [\mathbf{X}, \cdots, \mathbf{X}]$ and corrupted version of $\overline{\mathbf{X}}$ as $\tilde{\mathbf{X}}$, (1) can be reduced to

$$\frac{1}{2mn}\mathrm{Tr}\left[(\overline{\mathbf{X}} - \mathbf{W}\tilde{\mathbf{X}})^T(\overline{\mathbf{X}} - \mathbf{W}\tilde{\mathbf{X}})\right], \tag{3}$$

whose solution can be expressed as the closed form solution for ordinary least squares:

$$\mathbf{W} = \mathbf{P}\mathbf{Q}^{-1} \quad \text{with} \quad \mathbf{Q} = \tilde{\mathbf{X}}\tilde{\mathbf{X}}^T \quad \text{and} \quad \mathbf{P} = \bar{\mathbf{X}}\tilde{\mathbf{X}}^T. \tag{4}$$

Let $m \to \infty$, denoising transform $\mathbf{W}$ can be effectively computed with infinitely many copies of noise data. By the weak law of large numbers, $\mathbf{P}$ and $\mathbf{Q}$ converge to their mean values when $m \to \infty$. Then the mapping $\mathbf{W}$ can be expressed as:

$$\mathbf{W} = \mathbb{E}[\mathbf{P}]\mathbb{E}[\mathbf{Q}]^{-1} \quad \text{with} \quad \mathbb{E}[\mathbf{Q}] = \sum_{i=1}^{n}[\tilde{\mathbf{x}}_i\tilde{\mathbf{x}}_i^T]. \tag{5}$$

Off-diagonal entries in $\tilde{\mathbf{x}}_i\tilde{\mathbf{x}}_i^T$ are uncorrupted with the probability $(1-p)^2$, while for diagonal entries, this holds with probability $1-p$. Denote a vector $\mathbf{q} = [1-p, \cdots, 1-p]^T \in \mathbb{R}^{d+1}$, where $\mathbf{q}_\alpha$ and $\mathbf{q}_\beta$ represent the probabilities of no corruption happen to the feature α and β respectively. Defining the scatter matrix of the original uncorrupted input as $\mathbf{S} = \mathbf{X}\mathbf{X}^T$, the mean of $\mathbf{Q}$ can be expressed as:

$$\mathbb{E}[\mathbf{Q}]_{\alpha,\beta} = \begin{cases} \mathbf{S}_{\alpha\beta}\mathbf{q}_\alpha\mathbf{q}_\beta & \text{if } \alpha \neq \beta \\ \mathbf{S}_{\alpha\beta}\mathbf{q}_\alpha & \text{if } \alpha = \beta. \end{cases} \tag{6}$$

Similarly, the mean of $\mathbf{P}$ can be expressed as

$$\mathbb{E}[\mathbf{Q}]_{\alpha,\beta} = \mathbf{S}_{\alpha\beta}\mathbf{q}_\beta. \tag{7}$$

Then the reconstruction mapping $\mathbf{W}$ can be computed directly. This is the algorithm of the marginalized denoising autoencoder (mDA) [6].

Usually, the nonlinearity and the deep architecture is beneficial to feature learning. The nonlinearity is injected through the nonlinear quashing function $h(\cdot)$ after the reconstruction mapping $\mathbf{W}$ is computed. To perform the layer-wise stacking, several mDA layers are stacked by feeding the output of the $(t-1)$-th mDA (after the squashing function) as the input into the t-th layer mDA.

3 Marginalized SDA with Graph Regularization

In this section, we present our graph regularized marginalized Stacked Autoencoder (GmSDA) model.

3.1 General Graph Regularization Framework

As described in [11], a general class of graph regularization algorithms described by the following optimization problem: given the data $\mathcal{D} = \{\mathbf{x}_1, \cdots, \mathbf{x}_U\} \in \mathbb{R}^d$, we need to find a transformed representation $f(\mathbf{x}_i)$ w.r.t. $\mathbf{x}_i$ by minimizing

$$\sum_{i,j=1}^{U} \mathcal{L}(f(x_i, \alpha), f(x_j, \alpha), W_{ij}) \tag{8}$$

w.r.t. λ, subject to *Balance constraint*.

This type of optimization problem has the following main notations: $f(\mathbf{x}) \in \mathbb{R}^n$ is the embedding one trying to learn from a given example $\mathbf{x} \in \mathbb{R}^d$. It is parameterized by λ. In many techniques $f(\mathbf{x}_i) = f_i$ is a lookup table where each example i is assigned an independent vector f_i. $\mathcal{L}$ is a loss function between pairs of examples. Each element W_{ij} in $\mathbf{W}$ specifies the similarity or dissimilarity between samples $\mathbf{x}_i$ and $\mathbf{x}_j$. A balance constraint is often required for certain object functions so that a trivial solution is not reached.

3.2 Marginalized SDA with Graph Regularization

We propose the mSDA based deep learning system with graph regularization to learn domain-invariant features, which are used for training a linear SVM sentiment classifier. Our method can maximize the empirical likelihood (by DA) [10] and preserve the geometric structure (by graph regularization) simultaneously.

Considering a graph with N vertices where each vertex corresponds to a data point in the data set. The edge weight matrix $\mathbf{S}$ is usually defined as follows:

$$S_{ij} = \begin{cases} 1, & \text{if } \tilde{\mathbf{x}}_i \in \mathcal{N}_p(\tilde{\mathbf{x}}_j) \text{ or } \tilde{\mathbf{x}}_j \in \mathcal{N}_p(\tilde{\mathbf{x}}_i) \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

$\mathcal{N}_p(\mathbf{x}_i)$ denotes the set of p nearest neighbors of $\mathbf{x}_i$. Let f_i and f_j be the transformed representation (embedding) corresponding to $\tilde{\mathbf{x}}_i$ and $\tilde{\mathbf{x}}_j$ respectively, where $f_i = \mathbf{W}\tilde{\mathbf{x}}_i$, $f_j = \mathbf{W}\tilde{\mathbf{x}}_j$, we hope to preserve the local structure of data by minimizing the following equation:

$$\frac{1}{2} \sum_{i,j=1}^{n} ||f_i - f_j||^2 S_{ij} = \mathrm{Tr}(\mathbf{W}\tilde{\mathbf{X}}\mathbf{L}\tilde{\mathbf{X}}^T\mathbf{W}^T), \tag{10}$$

where $\mathbf{L}$ is the graph Laplacian, which can be obtained by $\mathbf{L} = \mathbf{D} - \mathbf{S}$. $\mathbf{D}$ is a diagonal matrix whose entries are column (or row, since $\mathbf{S}$ is symmetric) sums of $\mathbf{S}$, $D_{ii} = \sum_j S_{ij}$.

By integrating this graph regularization term into the objective function of mDA, we can get the objective function of the building block for our model:

$$\arg\min_{\mathbf{W}} \frac{1}{2mn} \mathrm{Tr}\left[(\bar{\mathbf{X}} - \mathbf{W}\tilde{\mathbf{X}})^T(\overline{\mathbf{X}} - \mathbf{W}\tilde{\mathbf{X}})\right] + \mathrm{Tr}(\mathbf{W}\tilde{\mathbf{X}}\mathbf{L}\tilde{\mathbf{X}}^T\mathbf{W}^T), \tag{11}$$

which can be solved analytically

$$\mathbf{W} = \mathbf{P}(\mathbf{Q} + \lambda\tilde{\mathbf{X}}\mathbf{L}\tilde{\mathbf{X}}^T)^{-1}, \tag{12}$$

where $\mathbf{P}$ and $\mathbf{Q}$ have the same definition in Eq.(3) and λ represents the parameter to balance the contribution of the graphic regularization. Follow the marginalized configuration in Section 2. We can solved $\mathbf{W}$ in closed form as in Eq.(4). The whole process of our GmSDA model is summarized in Algorithm 1.

Algorithm 1. mSDA via Graph Regularization (GmSDA)

Input: Data point $\mathcal{D} = \{\mathbf{x}_1, \cdots, \mathbf{x}_U\} \in \mathbb{R}^d$, where $\mathcal{D} = \mathcal{D}_\mathcal{S} \bigcup \mathcal{D}_\mathcal{T}$, number of the layer l, corruption level p, number of nearest neighbors k, parameter λ to balance the contribution of the graph regularizer
Output: hidden representation of l layer $\mathbf{h}^l$
Construct a weighted graphic S by KNN in Binary style;
Compute the graph Laplacian $\mathbf{L}$;
Initialize $\mathbf{X}^0 = D$;
for $t \leftarrow 1$ ***to*** l **do**
  Compute $\tilde{\mathbf{X}}^{t-1}\tilde{\mathbf{X}}^{t-1T}$, $\overline{\mathbf{X}}^{t-1}\tilde{\mathbf{X}}^{t-1T}$ and $\tilde{\mathbf{X}}^{t-1}\mathbf{L}\tilde{\mathbf{X}}^{t-1T}$;
  Solve $\mathbf{W}^t$ according to Eq.(12);
  Compute $\mathbf{h}^t = \tanh(\mathbf{W}^t\mathbf{X}^{t-1})$;
  Define $\mathbf{X}^t = [\mathbf{X}^{t-1}; \mathbf{h}]$;
end
return $\mathbf{h}^l$

To apply GmSDA to domain adaptation, we first learn feature representation in an unsupervised fashion on the whole set including source domain and target domain data. Then the output of all layers, after squashing function $\texttt{tanh}(\mathbf{W}^t\mathbf{h}^{t-1})$, are combined with original features $\mathbf{h}_0$ to form new representations. Finally a linear SVM is trained on the new features.

4 Experiments

We evaluate GmSDA on the reduced *Amazon reviews* benchmark dataset [4] together with several other related algorithms. This data set is more controllable and contains review from 4 type of domains: books, DVDs, electronics and Kitchen appliances. For computational reasons, we followed the convention of [7] and [6], considering only binary classification problem: whether a review is positive or negative. The data is preprocessed as the setting in [4][6]. Our experiments are using the first 5000 features.

We followed the experimental configuration in [6]: training a linear SVM on the raw bag-of-words feature from the labeled source domain and test it on target domain as the baseline. PCA (as another baseline) is used to project the entire data set on to a low dimensional subspace where dense features are learned. Another three type of features are also used to train a linear SVM: structural correspondence learning (SCL) [4], 1-layers SDA [7] and mSDA[6].

We use *Transfer Loss*, *Transfer Ratio* and *Transfer Distance* [7] as metrics for evaluating the performance of models.

There are 4 parameters in GmSDA: the corruption level p, number of layers l, number of nearest neighbors k and the balance parameter λ. k and λ are set as 30 and 0.01 respectively in our experiments. p was selected with 5-fold cross validation on the labeled data on source domain, following the setup in [6]. Near optimal value p is obtained by this cross validation process for each domain.

Figure 1 displays the transfer loss across the twelve domain adaptation tasks. The GmSDA outperforms all the compared models, achieving the best performance. For some tasks, the transfer loss has negative results which denotes that the learned features from source domain can train a better classifier than the one trained on the original target domain. It is worth noticing that, GmSDA achieves a lower transfer loss in ten out of twelve tasks than mSDA, indicating that the learned features bridge the gap between domains.

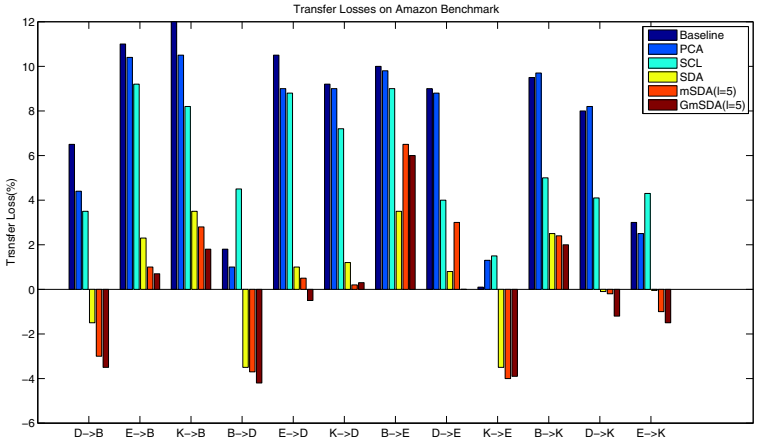

Fig. 1. Transfer losses on the Amazon benchmark of 4 domains: Books(B), DVDs(D), Electronics(E) and Kitchen(K) by different methods

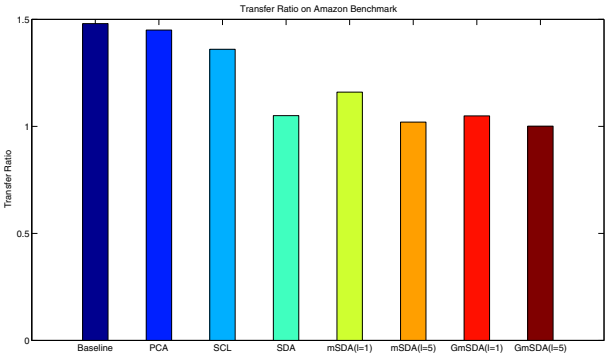

Fig. 2. Transfer ratios of algorithms on the Amazon benchmark

Figure 2 shows the transfer ratio for different methods and here we consider different layers of deep architectures as well. Compared with other methods, denoising autoencoder framework achieves better performance. In this framework, the deep architectures outperform the shallow ones and GmSDA get the best results. We can conclude that: 1). sharing the unsupervised pre-training across all domains is beneficial; 2). preserving the geometric structure is helpful to learn domain-invariant features; 3). deep architecture is better than the shallow one.

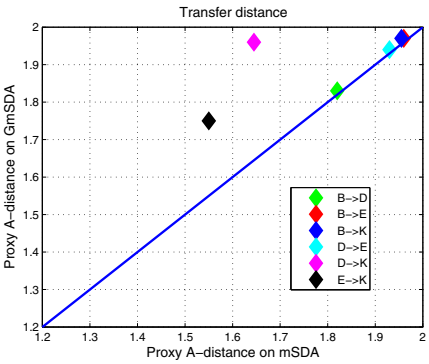

Fig. 3. Transfer distance: GmSDA vs. mSDA on the Amazon benchmark

Figure 3 shows the PAD of GmSDA and mSDA. All the points located beyond the blue line. It denotes that GmSDA features have bigger transfer distance than mSDA feature, which means it will be easier to distinguishing two domains with GmSDA features. We explain this effect through the fact that GmSDA is regularized with graph. With the help of the graph regularization, geometrical structure is exploited and the local invariance is considered, resulting a generally better representation. This helps both tasks, distinguishing between domains and sentiment analysis.

5 Conclusion

In this paper, we propose the mSDA based deep learning system with graph regularization. It can learn domain-invariant features which are suitable for sentiment classification. With help of the deep DA framework, we can maximize the empirical likelihood. Similarly, incorporating the graph regularization into mSDA, we can preserve the geometric structure to incorporate prior knowledge. This overcomes the shortcomings of most existing domain adaptation methods which focus only one aspect of the data or shallow framework. We compare our proposed approach against deep learning baselines over the reduced Amazon review benchmark. The experiments prove that our approach significantly outperforms all the baselines.

Acknowledgments. This work was supported partially by the National Natural Science Foundation of China (Grant No.61272248), the National Basic Research Program of China (Grant No.2013CB329401) and the Science and Technology Commission of Shanghai Municipality (Grant No.13511500200).

References

1. Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., Vaughan, J.W.: A theory of learning from different domains. Mach. Learn. 79(1-2), 151–175 (2010)
2. Blitzer, J., Dredze, M., Pereira, F.: Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In: ACL (2007)
3. Blitzer, J., Foster, D., Kakade, S.: Domain adaptation with coupled subspaces. In: AISTATS (2011)
4. Blitzer, J., McDonald, R., Pereira, F.: Domain adaptation with structural correspondence learning. In: EMNLP, pp. 120–128 (2006)
5. Chen, M., Weinberger, K.Q., Chen, Y.: Automatic feature decomposition for single view co-training. In: ICML (2011)
6. Chen, M., Xu, Z., Weinberger, K.Q., Sha, F.: Marginalized stacked denoising autoencoders. In: ICML (2012)
7. Glorot, X., Bordes, A., Bengio, Y.: Domain adaptation for large-scale sentiment classification: A deep learning approach. In: ICML (2011)
8. Huang, J., Smola, A.J., Gretton, A., Borgwardt, K.M., Scholkopf, B.: Correcting sample selection bias by unlabeled data. In: NIPS (2007)
9. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Found. Trends Inform. Retrieval 2(1-2), 1–135 (2008)
10. Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P.A.: Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. J. Mach. Learn. Res. 11, 3371–3408 (2010)
11. Weston, J., Ratle, F., Collobert, R.: Deep learning via semi-supervised embedding. In: ICML, pp. 1168–1175 (2008)
12. Xue, G.R., Dai, W., Yang, Q., Yu, Y.: Topic-bridged plsa for cross-domain text classification. In: SIGIR (2008)