Multi-Source Tri-Training Transfer Learning

Yuhu CHENG†, Nonmember, Xuesong WANG†(a), Member, and Ge CAO†, Nonmember

SUMMARY A multi-source Tri-Training transfer learning algorithm is proposed by integrating transfer learning and semi-supervised learning. First, multiple weak classifiers are respectively trained by using both weighted source and target training samples. Then, based on the idea of co-training, each target testing sample is labeled by using trained weak classifiers and the sample with the same label is selected as the high-confidence sample to be added into the target training sample set. Finally, we can obtain a target domain classifier based on the updated target training samples. The above steps are iterated till the high-confidence samples selected at two successive iterations become the same. At each iteration, source training samples are tested by using the target domain classifier and the samples tested as correct continue with training, while the weights of samples tested as incorrect are lowered. Experimental results on text classification dataset have proven the effectiveness and superiority of the proposed algorithm.

key words: transfer learning, Tri-Training, multi-source, text classification

1. Introduction

Transfer learning is a machine learning method for cross-domain learning, with the aim to seek the useful knowledge from one or more source domains related to a target domain and use it in helping the target task learning [1]. It breaks through the requirement of the conventional machine learning that training and target samples must be subject to the same distribution and can promote the performance of machine learning using a large number of labeled samples with the distribution being different from but similar to that of target samples. Generally, the effect of transfer learning is associated with the correlation between source domain and target domain. The higher such correlation, the better the transfer effect is; otherwise, the correlation is weak, thus being likely to result in a negative transfer. To avoid any possible negative transfer, many scholars have proposed different transfer learning algorithms, such as multi-source TrAdaBoost (MTrA) [2] and SLW [3]. The main idea of MTrA is that the training samples come from different source domains. At each iteration, MTrA selects a source domain that is closely related to the target domain to train the weak classifier. SLW is a novel two-phase framework to effectively transfer knowledge from multiple sources even when there exist irrelevant sources and imbalanced class distributions, where the label propagation idea is used. As noted in [4], the label propagation method is not applicable for high-dimensional data, such as 20_newsgroups data. Therefore, the SLW can not deal with the classification of high-dimensional data.

In semi-supervised learning, neither the expert labeling method in supervised learning nor the cluster learning means of unsupervised learning are used, but both labeled and unlabeled samples available are used for learning [5]. As the machine learning theory continues to develop, it is impossible to meet the needs of classifier training merely by relying on a limited number of labeled samples and it becomes extremely important to make use of unlabeled samples for co-training. Fujino et al. [6] designed a semi-supervised classifier (MHLE) trained by using unlabeled samples drawn by the same distribution as testing samples, and presented a semi-supervised classification method to deal with the transfer learning problem. The MHLE needs to designate a discriminative and generative model for a specific problem beforehand, which requires much domain knowledge.

Semi-supervised learning mainly follows with interest in how to assist labeled samples in classifier training by making use of unlabeled samples, without considering other possible impacts of relevant knowledge. Therefore, when labeled samples and unlabeled samples show a major difference in distribution, it is easy to result in a large error. Since transfer learning focuses on how to make effective use of labeled samples in relevant source domains, without considering the distribution of unlabeled samples in the target domain, and is severely dependent on the distribution of target training samples, it is easily observed with overfitting phenomena of classifier on target training samples. By integrating semi-supervised learning and transfer learning technologies, taking into account the distribution of unlabeled samples in the target domain as well as making use of labeled samples in relevant source domains for co-training, it is possible to reduce the possibility of improper sample selection due to difference of sample distribution as well as avoid the phenomena of overfitting, thus promoting the learning performance of classifier to a great extent.

2. Multi-Source Tri-Training Transfer Learning

TrAdaBoost [7] provides transfer learning with an excellent transfer mean: according to the distribution of target training samples, screen out the source domain samples with similar distribution. Yet, when the number of target training samples is limited, the classifier obtained by co-training with
source domain samples may easily reach the test accuracy of 100% on target training samples and thus enter a state of overfitting, while the screened source domain samples are severely dependent on the distribution of target training samples, showing a poor capacity of generalization. Furthermore, in multi-source transfer learning, due to the overfitting effect of classifier, it is extremely difficult to define which source domain will be much helpful for target task learning and thus it is impossible to accurately select any appropriate source domain and source training samples. Tri-Training [8] can avoid frequent use of statistical test technique for prediction of sample labels, but if the initial classifier is relatively weak, a large error will result in prediction of unlabeled samples, thus causing noise to the training of the third classifier. In another word, Tri-Training is relatively dependent on initial classifier and sample distribution. Being enlightened by the idea of co-training, a multi-source Tri-Training transfer learning (MST3L) algorithm is proposed. By introducing the idea of co-training into transfer learning, it is possible to resolve the noise impact likely to be caused by the prediction error of unlabeled samples as well as reduce the possibility for overfitting and additionally avoid any negative transfer.

Figure 1 shows the flow chart of MST3L where $D_{S_k} = \{(x_i^{S_k}, y_i^{S_k})\}_{i=1}^{n_{S_k}}$ and $D_T = \{(x_i^{T}, y_i^{T})\}_{i=1}^{n_T}$ are the $k$th source and target training set, $n_{S_k}$ and $n_T$ are the numbers of samples in corresponding sets. Compared with Tri-Training, MST3L no longer has merely two initial training sets, but has the same number as that of source domains. In another word, the number of weak classifiers is also the same as that of source domains, i.e., $N$ number of weak classifiers. Weak classifiers are used to predict labels of unlabeled samples in the target testing set. Such a target testing sample having the same predicted label is selected as a high-confidence sample and then is added into the target training set to construct a new training set. The new training set is trained to obtain a target domain classifier, which is used to test the source training sets, select samples classified correctly for adding to the next iteration and reduce the weight of samples with classification errors: i.e., select the samples helpful for target task learning for training and reduce the impact of unhelpful samples on target task learning. We contrast two training sets obtained at two successive iterations, i.e., determine whether $D_{T(t)}$ is the same as $D_{T(t-1)}$. The initial $D_{T'}$, i.e., $D_{T(0)}$, is set as the target training set $D_T$. If it is not the same, it indicates source domains still contain samples that are not related to target task learning and it is necessary to further reduce the weight of such samples according to $w_{h(t)} = w_{h(t-1)} \frac{1}{1 + ||h(x_i^{S_k}) - h(x_i^{T})||}$; otherwise, it indicates the algorithm converges and the learning process is ended.

3. Theoretic Analysis

3.1 Negative Transfer

The goal of transfer learning is to transfer helpful knowledge from $D_{S_k}$ to $D_T$ so as to obtain a better target domain classifier. It can be intuitively known that the source samples helpful for target task learning should satisfy the following criterions: 1) they have high similarity with the target training samples; 2) these class labels predicted by the target classifier are the same as a high-confidence sample and add it into $D_T$. Thus, a new target training set $D'_T$ is constructed, which can be used to train $h$. 5. Determine whether $D_{T(t)}$ is the same as $D_{T(t-1)}$. If yes, the learning process is ended. Otherwise, turn to 6.

6. Update weights of source training samples according to $w_{h(t)} = w_{h(t-1)} \frac{1}{1 + ||h(x_i^{S_k}) - h(x_i^{T})||}$; otherwise, it indicates the algorithm converges and the learning process is ended.

Algorithm 1: Multi-Source Tri-Training Transfer Learning

**Input:** source training sets $D_{S_1}, D_{S_2}, \ldots, D_{S_N}$, target training set $D_T$, target testing set $U$, number of samples in corresponding set $n_{S_k}$, $n_T$ and $n_U$, maximum iterations $M$

**Output:** target domain classifier $h$

1. $t \leftarrow 0$, $D_{T(0)} = D_T, w_{h(0)}$.
2. $D_h = D_{S_k} \cup D_T, k = 1, 2, \ldots, N$.
3. Train $N$ number of weak classifiers $h_1, h_2, \ldots, h_N$ using $D_1, D_2, \ldots, D_N$ respectively, and use these trained weak classifiers to predict the labels of the target testing samples.
4. Select such a sample whose label predicted by all weak classifiers is the same as a high-confidence sample and add it into $D_T$. Thus, a new target training set $D'_T$ is constructed, which can be used to train $h$.
5. Determine whether $D_{T(t)}$ is the same as $D_{T(t-1)}$. If yes, the learning process is ended. Otherwise, turn to 6.
6. Update weights of source training samples according to $w_{h(t)} = w_{h(t-1)} \frac{1}{1 + ||h(x_i^{S_k}) - h(x_i^{T})||}$.
7. $t \leftarrow t + 1$.
8. If $t < M$, return to 2. Otherwise, the learning process is ended.
For the source sample $x^S_i$, $D^S_T$ can be divided into two parts, i.e., $D^S_T = D^s_{T(0)} \cup D^s_{T(1)}$. If the class label of $x^S_i$ is the same as that of $x^S_j$, there is $x^S_i \in D^s_{T(0)}$, i.e., $2u^S_{ij} - 1 = 1$. If the class label of $x^S_i$ is different from that of $x^S_j$, there is $x^S_i \in D^s_{T(1)}$, i.e., $2u^S_{ij} - 1 = -1$. Equation (1) can be rewritten as:

$$\beta^S_i = \exp\left[ \sum_{x^S_i \in D^s_{T(0)}} s_{ij}(2u^S_{ij} - 1) + \sum_{x^S_i \in D^s_{T(1)}} s_{ij}(2u^S_{ij} - 1) \right]$$

$$= \exp\left[ \sum_{x^S_i \in D^s_{T(0)}} s_{ij} - \sum_{x^S_i \in D^s_{T(1)}} s_{ij} \right] = \exp(s^S_i - s^S_j)$$

(2)

where, $s^S_i = \sum_{x^S_i \in D^S_T} s_{ij}$, $s^S_j = \sum_{x^S_j \in D^S_T} s_{ij}$. Because $\exp()$ is a monotonically increasing function, there is $\beta^S_i \in (1, e]$ when $s^S_i > s^S_j$, meaning that the classifying result of $x^S_i$ is consistent with most samples in the target setting $D^S_T$, i.e., $x^S_i$ plays a positive transfer on the target task learning. There is $\beta^S_i \in [e^{-1}, 1)$ when $s^S_i < s^S_j$, meaning that the classifying result of $x^S_i$ is inconsistent with most samples in $D^S_T$, i.e., $x^S_i$ plays a negative transfer on the target task learning.

Definition 4: We apply the classification inconsistency between $x^S_i$ and $x^S_j$ to measure the training error on $x^S_i$:

$$loss(x^S_i) = \exp\left[ \sum_{j=1}^{n_T} s_{ij}(1 - 2u^T_{ij}) \right]$$

(3)

In order to avoid negative transfer, $m_k$ number of samples having stronger transfer capability are selected from $D_{S_k}$ and are added into $D_T$ to minimize $F_{S_k}$. Therefore, the loss function of $h$ on $\{x^S_{i1}\}_{i=1}^{m}$ is:

$$F_{S_k} = \sum_{i=1}^{m_k} loss(x^S_{i1}) = \sum_{i=1}^{m_k} \exp\left[ \sum_{j=1}^{n_T} s_{ij}(1 - 2u^T_{ij}) \right]$$

(4)

Theorem 1: For $x^S_i$, the training error loss is inversely proportional to its transfer capability, i.e., $loss(x^S_i)$ is inversely proportional to $\beta^S_i$.

Proof: According to definition 3, Eq. (3) is rewritten as:

$$loss(x^S_i) = \exp\left[ \sum_{j=1}^{n_T} s_{ij}(1 - 2u^T_{ij}) \right]$$

$$= \exp\left[ \sum_{x^S_i \in D^s_{T(0)}} s_{ij}(1 - 2u^T_{ij}) + \sum_{x^S_i \in D^s_{T(1)}} s_{ij}(1 - 2u^T_{ij}) \right]$$

$$= \exp[-(s^S_i - s^S_j)] = \frac{1}{\beta^S_i}$$

(5)

It can be easily known from above that $loss(x^S_i)$ is inversely proportional to $\beta^S_i$.

Theorem 2: at each iteration, transferring $m_k$ samples having stronger transfer capability from $D_{S_k}$ to $D_T$ can minimize the training error loss of $h$ on $D^S_{T(0)}$.

Proof: Based on Eqs. (4) and (5), we can obtain:

$$F_{S_k} = \sum_{i=1}^{m_k} loss(x^S_{i1}) = \sum_{i=1}^{m_k} \frac{1}{\beta^S_i}$$

(6)

At each iteration, we arrange all source training samples in a descending order according to the transfer capability and select the former $m_k$ samples to add into the target training set. For the $m_k$ number of samples, there is $\beta^S_i \in (1, e]$. Hence, Eq. (6) can be minimized.

3.2 Time Complexity

Suppose the time complexity of training the $k$th weak classifier and the target domain classifier are $C_{U_k}$ and $C_h$, that the time complexity of testing a sample in $U$ using the $k$th weak classifier is $U_{U_k}$, and the time complexity of updating a weight is $C_{w_k}$, then the time complexity of MST3L is $M \sum_{k=1}^{N} (C_{U_k} + n_UU_{U_k} + C_h + n_SC_{w_k})$. If the number of training samples on each source domain is about the same, the time complexity of MST3L approximately is $NM(C_{U_k} + n_UU_{U_k} + C_h + n_SC_{w_k})$.

4. Experimental Results and Analysis

4.1 Experimental Dataset

A commonly used UCI machine learning dataset is selected to test the performance of MST3L, with the compared algorithms as Tri-Training and MTrA. 20 newsgroups is a group of text classification data, containing 20 classes. Each class contains 1,000 newsgroup documents of the same class. Many classes of newsgroups have an overlapped part.

Simply, for the experiment, 2 newsgroups on different topics (motorcycles VS med) are selected as target domains for binary classification study. In experiment, set the maximum iterations as 15, select weighted support vector machine (SVM) as weak classifier and kernel function as Gaussian kernel; penalty factor and kernel width are selected with 5-fold cross-validation and initial weight of samples is taken as average weight, of which the weights of source and target domain samples are set as $1/2n_{s_k}$ and $1/2n_T$.

4.2 Impact of Number of Target Training Samples on Algorithm Performance

The experiment includes two parts. The first part tests the impact of changing number of target training samples on algorithm performance when the source domain remains unchanged. As for MST3L and MTrA, we select two source domains with the same topic as the target domain: autos VS electronics, baseball VS crypt. The number of target training samples is set respectively as 40, 60, 80, 100, 120, 140, 160, 180 and 200. In another word, the number of training samples for such classes as motorcycles and med is 20, 30, 40, 50, 60, 70, 80, 90 and 100, respectively. As a semi-supervised learning algorithm, Tri-Training needs no source domain. In addition, since the training samples of Tri-Training are obtained by sampling at random with Bagging method (where Bagging is used to select at random
Table 1  Comparison of classification accuracy.

| Number of target training samples | Tri-Training (%) | MTrA (%) | MST3L (%) |
|----------------------------------|-----------------|----------|-----------|
| 40                               | 90.9634         | /        | 94.3481   |
| 60                               | 91.4249         | /        | 94.0790   |
| 80                               | 92.1622         | /        | 94.7924   |
| 100                              | 94.3465         | 92.9860  | 94.9900   |
| 120                              | 94.8951         | 93.2972  | 95.0575   |
| 140                              | 94.8776         | 93.8229  | 95.0583   |
| 160                              | 95.3630         | 94.7112  | 95.6159   |
| 180                              | 95.6104         | 94.5560  | 95.6246   |
| 200                              | 95.0990         | 94.8461  | 95.7767   |

Table 2  Settings of source domain for different group experiments.

| Group 1 experiment | Group 2 experiment | Group 3 experiment | Group 4 experiment |
|--------------------|--------------------|--------------------|--------------------|
| auto vs. electronics; baseball vs. crypt | auto vs. electronics; baseball vs. crypt; ice hockey vs. space | auto vs. electronics; baseball vs. crypt; ice hockey vs. space | auto vs. electronics; baseball vs. crypt; ice hockey vs. space; pc.hardware vs. religion.misc; guns vs. for sale |

4.3 Impact of Source Domain on Algorithm Performance

The second part of the experiment examines the impact of different source domain settings on the performance of transfer learning algorithms in the circumstance that the target training set remains unchanged. This part of experiment includes four groups. Table 2 shows the settings of source domain used for each group of experiment. From Table 2, it may be observed that disregarding the number of target training samples, MST3L can achieve a correct result and shows the highest classification accuracy among the three algorithms.

Group 3 adds one source domain on a different topic from the target domain to examine the effectiveness of algorithm in case of any source domain with less correlation with the target domain. Based on Group 3, Group 4 adds another source domain on a topic different from the target domain.

During the experiment, the number of target training samples remains unchanged as 120, while the other parameter settings are the same as the first part of experiment. Figure 2 shows the curve of classification accuracy obtained by four groups of experiments. From Fig. 2, it may be observed: (1) Under the same experimental conditions, the classification accuracy of MST3L is higher than that of MTrA. (2) Due to the relatively high correlation of different source domains with the target domain, they can provide more useful knowledge. Therefore, for MST3L as well as MTrA, the classification accuracy of Group 2 is higher than that of Group 1. (3) From Figs. 2 (b)–(d), it may be observed that the classification accuracy of MTrA shows no changes. This is because Group 3 and Group 4 contain source domains on a topic different from the target domain, showing relatively high difference in data distribution and less similarity. When MTrA selects the source domain with the maximum correlation, the added source domain does not play role in helping the target task learning. In another word, the added source domain does not play role in helping the target task learning. (4) From Figs. 2 (b) and (c), it may be observed that even though Group 3 has one more source domain, relatively different from the target domain and on a different topic, than Group 2, MST3L can still select, through co-training, appropriate source domain samples to co-help the target task learning so that the classification accuracy is improved. It also indicates that as compared with MTrA, MST3L restrains the incidence of negative transfer to a certain extent. (5) From Figs. 2 (c) and (d), it may be observed that when the number of source domains with a major
difference from the target domain increases to 2, the added sample set, i.e., guns VS for sale, fails to play a role in promoting the target task learning and the classification accuracy of MST3L in Group 4 is the same as in Group 3. (6) It is easily seen from Figs. 2(a)–(d) that about after 7 iterations, the classification accuracies of MST3L under different experimental setting tend to stabilize, i.e., MST3L is able to converge.

5. Conclusions

Transfer learning stresses on the use of source domain knowledge, while semi-supervised learning focuses on the use of unlabeled samples. From a different aspect, the two methods resolve the problem that the limited number of labeled samples in machine learning cannot train a reliable classifier. Yet, the two methods are highly dependent on the distribution of target training samples. In other words, when there is a major difference between labeled samples and unlabeled samples by distribution, the learning effect is not so ideal. To resolve such problem as severe dependence on the distribution of target training samples, a multi-source Tri-Training transfer learning algorithm is proposed. On the one hand, the Boosting idea of transfer learning is used to select appropriate labeled samples from source domains. On the other hand, the co-training idea of semi-supervised learning is used to select from target domains the unlabeled samples with the distribution consistent with that of target training samples. Experimental results on text classification dataset indicate that MST3L can achieve a more excellent learning effect by making adequate use of available and unknown resources.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (61273143), and by the Fundamental Research Funds for the Central Universities (2013RC12).

References

[1] S.J. Pan and Q. Yang, “A survey on transfer learning,” IEEE Trans. Knowl. Data Eng., vol.22, no.10, pp.1345–1359, 2010.
[2] Y. Yao and G. Doretto, “Boosting for transfer learning with multiple sources,” Comput. Vis. Pattern Recognit., pp.1855–1862, 2010.
[3] L. Ge, J. Gao, H.Q. Ngo, K. Li, and A.D. Zhang, “On handling negative transfer and imbalanced distributions in multiple source transfer learning,” Proc. SIAM International Conference on Data Mining, 2013.
[4] F. Wang and C.S. Zhang, “Label propagation through linear neighborhoods,” Proc. 23rd International Conference on Machine Learning, pp.985–992, 2006.
[5] X. Zhu, Semi-supervised learning literature survey, Technical Report 1530, University of Wisconsin-Madison, 2006.
[6] A. Fujino, N. Ueda, and M. Nagata, “A robust semi-supervised classification method for transfer learning,” Proc. 19th International Conference on Information and Knowledge Management and Co-located Workshops, pp.379–388, 2010.
[7] W. Dai, Q. Yang, G. Xue, and Y. Yu, “Boosting for transfer learning,” Proc. 24th International Conference on Machine Learning, pp.193–200, 2007.
[8] Z.H. Zhou and M. Li, “Tri-training: exploiting unlabeled data using three classifiers,” IEEE Trans. Knowl. Data Eng., vol.17, no.11, pp.1529–1541, 2005.