Self-Paced Co-training

Fan Ma [1] **Deyu Meng** [*1] **Qi Xie** [1] **Zina Li** [1] **Xuanyi Dong** [2]

Abstract

Co-training is a well-known semi-supervised learning approach which trains classifiers on two different views and exchanges labels of unlabeled instances in an iterative way. During co-training process, labels of unlabeled instances in the training pool are very likely to be false especially in the initial training rounds, while the standard co-training algorithm utilizes a "draw without replacement" manner and does not remove these false labeled instances from training. This issue not only tends to degenerate its performance but also hampers its fundamental theory. Besides, there is no optimization model to explain what objective a co-training process optimizes. To these issues, in this study we design a new co-training algorithm named self-paced co-training (SPaCo) with a "draw with replacement" learning mode. The rationality of SPaCo can be proved under theoretical assumptions utilized in traditional co-training research, and furthermore, the algorithm exactly complies with the alternative optimization process for an optimization model of self-paced curriculum learning, which can be finely explained in robust learning manner. Experimental results substantiate the superiority of the proposed method as compared with current state-of-the-art co-training methods.

1. Introduction

Semi-supervised learning (SSL) aims to implement learning on both labeled and unlabeled data through fully considering the supervised knowledge delivered by labeled data and unsupervised data structure under unlabeled ones (Zhu, 2011). Co-training (Blum & Mitchell, 1998) is one of the most classical and well known SSL approaches that train classifiers on two views and exchanges labels of unlabeled instances in an iterative way. In the recent years, co-training has been attracting much attention attributed to both its wide applications, like web classification and visual detection (Xu et al., 2009), and rational theoretical supports (Blum & Mitchell, 1998; Balcan et al., 2004; Wang & Zhou, 2010; 2013).

However, there are still some limitations existing in current co-training investigation. Specifically, although there are multiple theoretical results to support the rationality of the co-training regimes, most of them require a strong pre-assumption that the false pseudo-labeled instances can be correctly recognized during training by classifiers or pseudo labels of unlabeled instances predicted in each iteration are of a high confidence extent. Based on such high-confidence assumption, most of current co-training algorithms add pseudo-labeled samples into training without replacement. Nevertheless, in most real cases such assumption is too subjective to be satisfied, especially in the early learning stage of a co-training algorithm. The learned classifiers might be not able to confidently distinguish certain samples, or precisely pseudo-annotate them with an expected accuracy requirement. This not only inclines to degenerate the performance of co-training since those false pseudo-labeled involved in training have no chance to be rectified in the latter training process on account of such "draw without replacement" manner, but also might make the basic assumption under theoretical support of co-training incorrect.

Another critical issue in most of current co-training methods is on their absence of an optimization model that can measure the performance and explain the intrinsic mechanism under the co-training implementation. As formally defined in (Mitchell, 1997), the performance measure is one of the necessary elements for a machine learning method. It is thus meaningful to explore whether there exists such an optimization model, which can finely interpret the co-training implementation as the process of solving this model. Such model also should be helpful in revealing more insights underlying co-training.

To address the aforementioned issues, a new co-training method, called self-paced co-training (SPaCo) is proposed in this study. The method differs from the previous co-training regimes mainly in two aspects: Firstly, it utilizes a

[1]Xi'an Jiaotong University, Xi'an, China [2]University of Technology Sydney, Sydney, Australia. Correspondence to: Deyu Meng <dymeng@xjtu.edu.cn>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017.

"draw with replacement" learning manner. In the method, an unlabeled instance having been added into the training pool is likely to be removed if classifiers in later training rounds identify it as a low-confidence annotated one. Besides, the pseudo label of an unlabeled instance has chance to be rectified based on the prediction knowledge obtained by classifiers in later training rounds. Secondly, the SPaCo method employs a serial mode to update the classifiers of two views in co-training implementation instead of the parallel mode commonly adopted by previous methods. Under such amelioration, the new method can be proved to still guarantee the theoretical effectiveness under ϵ-expansion assumption in the traditional co-training theory, while more importantly, such series implementation exactly complies with the alternative optimization algorithm on solving a concise optimization model. Besides, it is substantiated that the new method can attain evidently better performance beyond current state-of-the-art co-training methods on average of our experiments, which further verifies the rationality of the proposed SPaCo algorithm.

In summary, this work makes the following contributions:

- A new co-training method, called SPaCo, is proposed by adopting the "draw with replacement" learning manner. Similar to traditional co-training strategies, the theoretical soundness of the new method can also be proved under the commonly specified ϵ-expansion assumption for pseudo-label confidence.
- The implementation process of the new method exactly complies with an alternative updating algorithm for a underlying optimization model. This model corresponds to a self-paced curriculum learning paradigm, which helps reveal more "easy-to-hard" insights under the co-training implementation.
- Through this derived model, the effectiveness of the proposed SPaCo method can be finely interpreted as: the method implements robust learning regimes in both views under the regularization that the robust loss forms in two views are closely related. This understanding provides a natural explanation for the effectiveness mechanism under such co-training strategy, without any requirement of subjective assumptions for pseudo-label confidences.
- Experimental results on multiple text classification and person re-ID data substantiate the superiority of the SPaCo method as compared with current state-of-the-art co-training methods.

The rest of the paper is organized as follows. We first briefly introduce related work in Section 2, and then present the proposed SPaCo algorithm and its underlying model in Section 3. More theoretical explanations on its insights are also provided in this part. We then present the experimental results and finally give a conclusion remark.

2. Related Work

2.1. Co-training methods

The traditional co-trainig method (Blum & Mitchell, 1998) builds classifiers for different views and exchanges predictions of high-confidence unlabeled data to augment the training set in two views in each training round. Afterwards, multiple advancements have been developed. These co-training variants can be roughly categorized into two paradigms. One paradigm is to follow the iterative training process of co-training but to label unlabeled samples using different methods with certain confidence criterion in each iteration (Goldman & Zhou, 2000; Brefeld & Scheffer, 2004; 2006; Zhou et al., 2007; Li & Zhou, 2007; Zhang & Zhou, 2011). The other is to embed extra information from other views as a regularization term into the learning objective (Sindhwani et al., 2005; Sindhwani & Rosenberg, 2008; Ye et al., 2015) and turn the semi-supervised multi-view problem into a new optimization problem.

2.2. Other methods related to co-learning

Recently, there are multiple other methods proposed aiming to simultaneously learn classifiers on two different views or multi-views. While different from co-training approaches, these methods directly pseudo-label all unlabeled samples and involve them into the training process. A typical approach along this line is Co-EM (Nigam & Ghani, 2000), which iteratively updates labels of unlabeled data based on the posterior class probability calculated by naive Bayesian learners, and updates the classifiers on all of them. Several other methods directly encode the unknown labels of unlabeled data and classifier parameters on two views into a model, and simultaneously calculate all these variables through solving this model. Typical methods of this category include Co-MR (Sindhwani & Rosenberg, 2008), which deduces a co-regularization kernel by exploiting two Reproducing Kernel Hilbert Spaces defined over the same input space, and RANC (Ye et al., 2015), which assumes predictions for unlabeled data under different views are consistent with each other and enforces an affixed rank constraint on optimization function of each view.

2.3. Co-training theory development

The rationality of co-training is supported by a series of related theoretical analyses. E.g., (Blum & Mitchell, 1998) showed that class on two views is learnable in the PAC model with classification noise when the features of two views are independent given the class. To further relax the assumption for co-training, (Abney, 2002) provided a

weaker view-independence condition that brings about the success of co-training. Afterwards, (Balcan et al., 2004) introduced the ϵ-expansion assumption, which is a confidence assumption on pseudo labeled positive samples, further relaxing the condition of guaranteeing the effectiveness of a co-training strategy. Later, (Wang & Zhou, 2010) made a new analysis of co-training on label propagation strategy designed for co-training.

Despite providing theoretical support for current co-training methods, all these theories include some subjective assumptions like the independence between classifiers of different views and the confidence extent of pseudo-labels of unlabeled samples obtained by the algorithm. These assumptions, however, are not only hard to be justified in real applications, but also not very intuitive to be easily understood by common co-training users, which might possibly keep it from being more extensively utilized in practice.

2.4. Self-paced learning

(Bengio et al., 2009) proposed a learning paradigm called *curriculum learning*(CL), in which a model is learned by gradually including samples from easy to complex in training so as to increase the entropy of training samples. Afterwards, self-paced learning (Kumar et al., 2010) is proposed to embed curriculum design as a regularization term into the learning objective. Due to its generality, the SPL theory has been widely applied to various tasks, such as object tracking (Supancic & Ramanan, 2013), image classification (Jiang et al., 2015), and multimedia event detection (Jiang et al., 2014a;b). Especially, the SPL paradigm has been integrated into the system developed by CMU Informedia team, and achieved the leading performance in challenging TRECVID MED/MER competition organized by NIST in 2014 (Yu et al., 2014).

Let $L(y_i, g(\mathbf{x}_i, \mathbf{w}))$ denote the loss function which calculates the cost between the ground truth label y_i and the estimated one $g(\mathbf{x}_i, \mathbf{w})$. Here $\mathbf{w}$ represents the model parameter inside the decision function g. The SPL model considers a weighted loss term for all samples and a general self-paced regularizer with respect to sample weights, expressed as:

$$\min_{\mathbf{w},\mathbf{v}\in[0,1]^n} \mathbf{E}(\mathbf{w}, \mathbf{v}; \lambda) = \sum_{i=1}^{n} (v_i L(y_i, g(\mathbf{x}_i, \mathbf{w})) + f(v_i, \lambda)), \tag{1}$$

where λ is the age parameter for controlling the learning space, and $f(v, \lambda)$ represents the self-paced regularizer (SP-regularizer briefly). By jointly learning the model parameter $\mathbf{w}$ and the latent weight $\mathbf{v}$ using alternative optimization strategy with gradually increasing age parameter, more samples can be automatically included into training from easy to complex in a purely self-paced way.

The recent development of SPL includes that (Jiang et al., 2015) improved SPL as a more effective self-paced curriculum learning (SPCL) regime by embedding useful loss prior knowledge into the model and analyzed that this regime is analogous to rational instructor-student-collaborative learning mode of human teaching. And multiple researches (Zhang et al., 2015; Zhao et al., 2015; Pi et al., 2016) showed that SPL worked well when dealing with real data. Besides, (Meng & Zhao, 2015; Ma et al., 2017) proved that the optimization problem of SPL solved by the alternative optimization algorithm is equivalent to a robust loss minimization problem solved by a majorization-minimization algorithm. This work first reveals the insightful understanding of the robust learning mechanism under SPL.

3. Self-paced Co-training

In this section, we introduce the details of our proposed framework, Self-Paced Co-training (SPaCo) model. We first present the mathematical form of this model, and then introduce the alternative optimization algorithm for solving this model. It is then evident that the algorithm finely complies with a co-training strategy in a "draw with replacement" mode and a series implementation manner. Some properties of the algorithm will be detailedly analyzed afterwards. We then provide the theoretical support under ϵ-expansion assumption, and show that the effectiveness of the algorithm can be proved under the conventional routine of co-training. Finally, a new interpretation on the mechanism of this method will be presented from the viewpoint of self-paced learning.

3.1. SPaCo Model

We first present the following SPaCo model, which extends the self-paced learning optimization model (1) to two view scenarios, by introducing importance weights of two views $v_k^{(1)}, v_k^{(2)}$, $(k = l+1, \cdots, l+u)$, together with the corresponding hard self-paced regularizer $f(v, \lambda) = \lambda v$ as proposed in (Jiang et al., 2014a):

$$\begin{aligned} &\min_{\substack{\mathbf{w}^{(j)}, y_k, v_k^{(j)} \in [0,1] \\ j=1,2; k=l+1,\cdots,l+u}} E(\mathbf{w}^{(j)}, v_k^{(j)}, y_k; \lambda^{(j)}, \gamma) = \\ &\sum_{j=1}^{2}\sum_{i=1}^{l} L(y_i, g^{(j)}(\mathbf{x}_i^{(j)}; \mathbf{w}^{(j)})) + \frac{1}{2}\sum_{j=1}^{2} ||\mathbf{w}^{(j)}||_2 \\ &+ \sum_{j=1}^{2}\sum_{k=l+1}^{l+u} (v_k^{(j)} L(y_k, g^{(j)}(\mathbf{x}_k^{(j)}; \mathbf{w}^{(j)})) - \lambda^{(j)} v_k^{(j)}) \\ &- \gamma (\mathbf{v}^{(1)})^T \mathbf{v}^{(2)}, \end{aligned} \tag{2}$$

where l and u denote the number of labeled and unlabeled instances, respectively. $\mathbf{x}_i^{(j)}$ is the i^{th} sample ($i = 1, \cdots, l+u$) under j^{th} view ($j = 1, 2$), and y_i is the common label of $\mathbf{x}_i^{(j)}$ for every j. $v_k^{(j)}$ denotes the weight of

$\mathbf{x}_k^{(j)}$ where $k = l+1, ..., l+u$. $\mathbf{v}^{(j)}$ is an u-dimensional vector preserving all the weights of unlabeled instances under j^{th} view where its k^{th} element is $v_{l+k}^{(j)}$. $\mathbf{w}^{(j)}$ represents parameters of j^{th} classifier trained on j^{th} view. $\lambda^{(j)}$ is the age parameter controlling the training scale in each iteration with respect to j^{th} view, and γ is the parameter adjusting influence from the other view when one view is going to add more training samples.

The above SPaCo model actually corresponds to the sum of SPL model under two views plus a regularization term $(\mathbf{v}^{(1)})^T\mathbf{v}^{(2)}$. This inner product encodes the relationship of "sample easiness degree" between two views. The new co-regularizer delivers the basic assumption under co-training that different views share common knowledge of pseudo-labeled sample confidence (an unlabeled sample is likely to be labeled correctly or wrongly simultaneously for both views), and thus this inner product enforces the weight penalizing the loss of one view similar to that of the other view. This finely accords to the idea of SPCL and complies with an instructor-student-collaborative learning manner under a specific co-training curriculum.

3.2. AOS algorithm for solving (2)

The alternative optimization strategy (AOS) can be readily adopted to solve this SPaCo model. The optimization process are shown as follows[1]:

Initialization: The first step is to initialize parameters of model. $\mathbf{v}^{(1)}$ and $\mathbf{v}^{(2)}$ are zero vectors in R^u. $\lambda^{(1)}$ and $\lambda^{(2)}$ are initialized with small values to allow a few unlabeled instances into training for the first iteration. γ is set as a specific value in the whole training process. Two classifiers are simultaneously trained on labeled samples to get initial losses of both labeled and unlabeled instances.

Update $v_k^{(3-j)}(j = 1, 2)$: The physical meaning of this step is to prepare confident unlabeled instances (with non-zero $v_k^{(3-j)}$ values) for the training on the j^{th} view. This is known as the process of picking confident instances from one view in the traditional co-training algorithm. By calculating the derivative of Eq. (2) with respect to $v_k^{(3-j)}$, we can get

$$\frac{\partial E}{\partial v_k^{(3-j)}} = L_k^{(3-j)} - \lambda^{(3-j)} - \gamma v_k^{(j)}. \tag{3}$$

Then we can get the closed-form updating equation for v_i^{3-j} as follows:

$$v_k^{(3-j)*} = \begin{cases} 1, L_k^{(3-j)} < \lambda^{(3-j)} + \gamma v_k^j, \\ 0, otherwise. \end{cases} \tag{4}$$

[1]For the ease of description, we only present the optimization process under one view since parameters under the other view are optimized in the same way.

In the first iteration, all the $v_k^{(j)}$s are zeros according to the initialization. Thus unlabeled instances are selected only from the $(3-j)^{th}$ view. In other words, unlabeled instances with loss values less than $\lambda^{(3-j)}$ will be seen as confident instances.

Update $v_k^{(j)}$: The goal of this step is to formally define which unlabeled instances will be feeded into the training pool of $j^{(th)}$ view. The optimization process for $v_k^{(j)}$ is the same as previous step, but unlabeled instances selected in this step will be directly employed for training in j^{th} view.

From Eq. (4), we can easily observe that confident instances from the other view (picked in the previous step) possess higher chance than other instances to be selected for training.

Update $\mathbf{w}^{(j)}$: This step aims to train a classifier by virtue of the labeled and pseudo-labeled samples in the training pool of the j^{th} view. By setting the loss as well-known hinge loss, we can directly choose SVM to train the expected classifier. In this case, Eq. (2) degenerates to the standard SVM optimization problem as:

$$\min_{\mathbf{w}^{(j)}} \frac{1}{2}||\mathbf{w}^{(j)}||_2 + \sum_{i=1}^{l} L_i^{(j)} + \sum_{k=l+1}^{l+u} v_k^{(j)} L_k^{(j)}, \tag{5}$$

where $L_t^{(j)} = L(y_t, g(\mathbf{x}_t^{(j)}, \mathbf{w}^{(j)})), t = 1, \cdots, l+u$. This problem can be readily solved by any off-the-shelf SVM toolbox (Jiang et al., 2014a). For the cross entropy loss, we can employ deep learning network to train the expected classifier, and thus our model is not constrained within one single classification algorithm.

Update y_k: The next step is to update pseudo-labels of training samples by solving the following minimization sub-problem:

$$y_k = \underset{y_k}{\operatorname{argmin}} \sum_{j=1}^{2} v_k^{(j)} L(y_k, g(\mathbf{x}_k^{(j)}; \mathbf{w}^{(j)})). \tag{6}$$

It is easy to prove that the global optimum of the above problem can be obtained by directly setting the pseudo-label y_i of a training sample as the weighted sum of prediction value under two classifiers.

Once pseudo-labels of training samples are refreshed, $\lambda^{(1)}, \lambda^{(2)}$ are enlarged to allow more instances with lager loss values into the training pool in the next iteration. Then we repeat the above optimization process with respect to each variable under different views until there is no more available unlabeled instances or the preset largest iteration number is reached.

The entire process of this alternative optimization algorithm is summarized in Algorithm 1. It can be seen that

such optimization process exactly corresponds to the traditional co-traing algorithm with some reasonable adjustments. Assisted by this model, such a co-training algorithm possesses all of the necessary elements a formal machine learning method should have.

Algorithm 1 Alternative Optimization Algorithm for Solving SPaCo Model

Input: samples $x_1^{(1)}, \cdots, x_{l+u}^{(1)}, x_1^{(2)}, \cdots, x_{l+u}^{(2)}$, labels $y_1, ..., y_l$, parameters $\lambda^{(1)}, \lambda^{(2)}, \gamma$, and max_round.
Output: $\mathbf{w}^{(1)}, \mathbf{w}^{(2)}$.
Initialize $\mathbf{v}^{(1)}$, $\mathbf{v}^{(2)}$, $\lambda^{(1)}, \lambda^{(2)}$, and γ
Update $\mathbf{w}^{(1)}, \mathbf{w}^{(2)}$
training_round = 1
while not converge || training_round $<$ max_round **do**
  for $j \leftarrow 1$ to 2 **do**
    Update $v_k^{(3-j)}$: Prepare confident instances from $(3-j)^{th}$ view for training on j^{th} view
    Update $v_k^{(j)}$: Add unlabeled instances into the training pool of j^{th} view based on $L_k^{(j)}$ and $\gamma v_k^{(3-j)}$
    Update $\mathbf{w}^{(j)}$: Train a classifier (SVM for instance) on training pool of j^{th} view
    Update y_k: Find optimal pseudo label for each of selected unlabeled instances by solving Eq. (6)
    Augment $\lambda^{(1)}, \lambda^{(2)}$
  end for
end while
Return $\mathbf{w}^{(1)}, \mathbf{w}^{(2)}$

From Algorithm 1, we can easily observe that it has a very similar implementation scheme as traditional co-training methods. Specifically, it also iteratively trains classifiers on two views by exchanging labels of unlabeled instances. Yet beyond that, the proposed algorithm complies with an optimization implementation on a underlying self-paced learning model. This model thus tends to provide some novel insightful understandings on the intrinsic effectiveness mechanism under the co-training approach, which will be analyzed in Sec. 3.5.

3.3. Algorithm analysis

The standard co-training method (Blum & Mitchell, 1998) requires to simultaneously train classifiers of both views, and then select highly confident samples to label for each view and feed them into the training pool of the other view. The proposed SPaCo algorithm, as listed in Algorithm 1, mainly differs from traditional co-training methods in the following three-fold aspects.

First, instead of "draw without replacement" mode as conventional, the SPaCo algorithm utilizes a "draw with replacement" manner. The algorithm does not consistently keep the previously selected training pool unchanged, while a confidence sample in the pool has certain chance to be thrown out from it when the loss value of a sample is larger than a preset threshold $\gamma + \lambda$. Note that this confidence threshold is larger than that (λ) set for samples not in the training pool, implying that we still more prefer to keep the samples in the pool than those not in it. Also, when we set γ as an extremely large value, then it is easy to deduce that the algorithm will degenerate to a traditional "draw without replacement" method since the loss values of any samples in the training pool will be smaller than the threshold and will thus be definitely selected in the next round.

Second, instead of the parallel training way as conventional, the new algorithm uses a serial manner for training the classifiers of two views. This not only will make this algorithm fully comply with the alternative updating strategy for solving an optimization model, but also leads to better performance than traditional parallel model methods (see experiment part). This might possibly be due to the fact that the serial way can better guarantee the reliability of added high-confidence pseudo-label samples based on the updated while not the non-updated classifiers as in parallel way.

Third, when updating the training pool in one view, besides feeding into high-confidence samples justified by the other view, the new algorithm might add into the pool a few high-confidence samples which obtain very small loss values calculated on the current view. This is expected to make the algorithm use more reliable high-confidence knowledge from the predicted knowledge by current classifiers.

3.4. Learnable theory of SPaCo under ϵ-expansion assumption

Similar to the theoretical support for traditional co-training methods, we want to prove that the SPaCo algorithm is a PAC learning algorithm under the certain ϵ-expansion condition as utilized in (Balcan et al., 2004). First we give the definition of the ϵ-expansion condition:

Definition 1 *(Balcan et al., 2004) Let X^+ denote the positive region and D^+ denote the distribution over X^+, and $X_i (i = 1, 2)$ is the training data set in the i^{th} view. For $S_1 \subseteq X_1$ and $S_2 \subseteq X_2$, the D^+ is ϵ-expanding if the following inequality holds:*

$$P(\boldsymbol{S}_1 \oplus \boldsymbol{S}_2) \geq \epsilon \, min(P(\boldsymbol{S}_1 \wedge \boldsymbol{S}_2), P(\bar{\boldsymbol{S}_1} \wedge \bar{\boldsymbol{S}_2})), \quad (7)$$

where $P(\boldsymbol{S}_1 \wedge \boldsymbol{S}_2)$ denotes the probability of examples for being confident in both views, and $P(\boldsymbol{S}_1 \oplus \boldsymbol{S}_2)$ denotes the probability of examples for being confident in only one view.

We can then prove the following theorem for the proposed SPaCo algorithm. The proof is presented in supplementary material.

Theorem 1 *Let ϵ_{fin} and δ_{fin} be the desired accuracy and confidence parameters. Suppose that the serial ϵ-expanding condition is satisfied in each training round, and then we can achieve the error rate ϵ_{fin} with probability $1-\delta_{fin}$ by running the SPaCo for $N = O(\frac{1}{\epsilon} log \frac{1}{\epsilon_{fin}} + \frac{1}{\epsilon} \cdot \frac{1}{p_init})$ rounds, each time running algorithm A_1 and algorithm A_2 with accuracy and confidence parameters set to $\frac{\epsilon \cdot \epsilon_{fin}}{8}$ and $\frac{\delta_{fin}}{2N}$, respectively.*

Therefore, the rationality of the new algorithm can also be supported by the traditional theoretical means.

3.5. Co-robust-loss interpretation for SPaCo rationality

Based on the SPaCo model (2) underlying Algorithm 1, we can get some new insights underlying the co-training regimes.

(Meng & Zhao, 2015) has proved that the optimization problem (1) of SPL is closely related to a robust loss minimization problem. Such understanding can be utilized in this study to present a new understanding for the effectiveness insight underlying this co-training strategy. Specifically, in the SPaCo model (2), there is a separate SPL objective function for each view, which means that there implicitly exists a robust loss for training the classifier of each view on pseudo-labeled samples. However, such robust losses for different views are not independent while closely related to each other since a sample should be labeled correctly or wrongly for any view of data representation. Thus in SPaCo model (2), the co-training curriculum regularization actually encodes such relationship between robustness of different views. That is, through consistently exchanging pseudo-labels justified in different views, the robust loss functions of both views are enforced to be related by such regularization term. This guarantees a sound learning manner for the co-training process.

Note that such a explanation for the effectiveness of the SPaCo algorithm can be easily understood and requires no subjective assumptions on pseudo-label confidences or two-view independence. It is thus expected to facilitate a better extension of co-training paradigms to general users.

4. Experimental results

To validate the performance of the proposed SPaCo method, we first employ six text classification data sets derived from three real-world domains, where each data set is associated with two naturally partitioned or artificially generated views. Besides, we also apply our method to the person re-identification task, which is a popular research topic in the field of computer vision.

Table 1. Statistics of Utilized Data Sets

Data set	Number of examples	Attributes		Postive proportion
		view 1	view 2	
course	1051	344	42	21.88%
ads12	3279	45	49	14.04%
ads13	3279	45	47	14.04%
ads23	3279	49	47	14.04%
NG1	800	303	334	50%
NG2	800	303	334	50%

4.1. Text Classification Experiments

Datasets: We employ the following six data sets, all having been used for testing in the previous co-training literatures.

Course data: This data set contains 1,051 home pages collected from web sites of Computer Science departments of Cornell University[2]. These pages are manually labeled as course or noncourse, each with a page-based view (words appearing in the page itself) and a link-based view (words appearing in hyperlinks pointing to it). Among all homepages, course homepages (22%) correspond to positive examples while all others are negative examples.

Advertisement data: This data set contains advertising images in web pages[3]. Each image is described from multiple views, such as image properties, image caption, words occurring in the image source's url, words occurring in the affiliated web page's url and words occurring in the image anchor's url. By using the words of different areas, we create data sets named ads12, ads13 and ads23.

Newsgroup data: This data set is related to 16 newsgroups postings from the Mini-Newsgroup data[4]. Each group consists of 100 postings randomly drawn from the 1000 postings in the original 20-Newsgroup data. The 16 chosen newsgropus are divided into four groups, and we create NG1 and NG2 data sets based on a partition strategy (Zhang & Zhou, 2011).

Each utilized data set contains two classes. Table 1 summarizes the statistics of these data sets. For each data set, 25% of the data are retained as test examples while the rest are used as training examples, i.e., including both labeled and unlabeled examples. Three experiments are conducted on these six data sets with different number of labeled in-

[2]Data available at http://www.cs.cmu.edu/afs/cs/project/theo-20/www/data/

[3]Data available at https://archive.ics.uci.edu/ml/datasets/Internet+Advertisements

[4]Data available at http://www.cs.cmu.edu/afs/cs/project/theo-11/www/ naive-bayes/mini_newsgroup.tar.gz.

Table 2. Accuracy comparison of 6 competing methods on 6 testing data sets. Each result is averaged from 100 trials with independently sampled labeled samples. The best result in each series is highlighted in bold.

		SelfTrain	Cotrain	CoEM	CoMR	Cotrade	RANC	SPaCo
k=1	course	0.8034	0.8820	0.7884	0.7975	0.8904	0.7297	**0.9121**
	ng1	0.5521	0.6067	0.5679	0.5148	0.5792	**0.6170**	0.5970
	ng2	0.5775	0.5900	0.5988	0.5211	0.6199	0.6177	**0.6243**
	ad12	0.8733	0.9057	0.8635	0.8653	0.8874	**0.9336**	0.9253
	ad13	0.9082	0.9160	0.8705	0.8664	0.9212	0.9096	**0.9227**
	ad23	0.8975	0.8920	0.8621	0.8631	0.9049	0.9035	**0.9093**
	average	0.7687	0.7987	0.7585	0.7381	0.8005	0.7852	**0.8151**
k=2	course	0.8303	0.9086	0.8217	0.8145	0.9270	0.7878	**0.9315**
	ng1	0.5826	0.6353	0.6253	0.5260	**0.6530**	0.6475	0.6255
	ng2	0.6033	0.6467	0.6475	0.5422	0.6865	0.6658	**0.7036**
	ad12	0.8947	0.9057	0.8716	0.8690	0.9002	**0.9374**	0.9346
	ad13	0.9124	0.9243	0.8883	0.8720	0.9267	0.9118	**0.9320**
	ad23	0.9021	0.9109	0.8677	0.8665	0.9152	0.9084	**0.9207**
	average	0.7877	0.8219	0.7870	0.7484	0.8348	0.8098	**0.8413**
k=3	course	0.8525	0.9141	0.8651	0.8365	0.9298	0.8248	**0.9322**
	ng1	0.6509	0.7058	0.6724	0.5627	**0.7072**	0.7050	0.6827
	ng2	0.6858	0.6970	0.7212	0.5853	0.7573	0.7279	**0.7701**
	ad12	0.8986	0.9105	0.8812	0.8750	0.9011	0.9349	**0.9356**
	ad13	0.9208	0.9312	0.9077	0.8816	0.9255	0.9226	**0.9441**
	ad23	0.9014	0.9108	0.8753	0.8728	0.9089	0.9176	**0.9219**
	average	0.8183	0.8449	0.8205	0.7690	0.8550	0.8388	**0.8644**

stances. To simulate real-world cases where labeled samples are rarely available, only a small number of instances are randomly selected as labeled data. Among the training samples, we choose 2^k positive and $3 \cdot 2^k$ negative, $k = 1, 2, 3$, labeled instances for *Course*, 2^k positive and $6 \cdot 2^k$ negative labeled examples for *Advertisement*, and $2 \cdot 2^{k+1}$ positive and $2 \cdot 2^k$ negative labeled examples for *Newsgroup*, based on their different data size. On each data set, three series of experiments are implemented (i.e., for $k = 1, 2, 3$), and each experiment series contains 100 trials, with independently sampled labeled samples.

Experiment setting: The performance of SPaCo is compared with six current semi-supervised learning algorithms, including: SelfTrain (Scudder, 1965), the most conventional SSL method, the co-training method (Blum & Mitchell, 1998), the most conventional co-training method, CoTrade (Zhang & Zhou, 2011), one state-of-the-art co-training method, CoEM (Nigam & Ghani, 2000), CoMR (Sindhwani & Rosenberg, 2008) and RANC (Ye et al., 2015), representing the state-of-the-art for solving two-view co-learning problem. SVM is employed as base classifier in text classification task for all the iterative training methods.

For SelfTrain, two classifiers expand their training pool by selecting the "most confident" samples they thought by themselves in each training round. The standard co-training and Cotrade select the "most confident" samples justified by the other view. To avoid introducing too much noise, each classifier of SelfTrain and standard co-training only selects 1p(ositive) 3n(egative) for the *course* data, 1p 6n for the *ads* data, and 2p 2n for the *newsgroup* data. These three algorithms terminate when no more examples are available for training. Instead of augmenting training pool step by step, CoEM estimates class probability in every training round and employs these pseudo labels to re-estimate class probability after each iteration.

For CoMR, each data set adpots an unified representation by merging the two views, and the regularization parameters γ_1, γ_2 varied on a gird of values $(10^{(-6)}, 10^{(-4)}, 10^{(-2)}, 1, 10, 100)$ where the results from optimal configuration are reported. RANC embeds the rank constraints into the optimized function of multi-view learning object (Ye et al., 2015) and proposes two different ways to solve the problem. The ADMM method (Boyd et al., 2011) is adopted in this paper to get the solution.

For our SPaCo algorithm, instead of tuning λ directly, we increase the number of nonzero element of $\mathbf{v}^{(j)}$ in each training round. Besides, to judge unlabeled instances based on two views, we easily set γ as 1 throughout all our experiments. Its setting actually is not sensitive to the final performance of our algorithm.

Experimental results: The average accuracy over all 100

trails obtained by each competing method on each data set is shown in Table 2. From the table, we can easily observe that the proposed SPaCo method can attain the best (13 out of 18) or the second best (3 out of 18) performance among all competing methods. In average, SPaCo acquires an evident better performance than other competing methods under different sizes of initialized labeled examples. This verifies the superiority of the proposed method on these co-training problems.

4.2. Person Re-identification Experiments

The person re-identification (re-ID) task is usually viewed as an image retrieval problem, aiming to match pedestrians from the gallery (Zheng et al., 2016). Specifically, given a person-of-interest (query), a person re-ID method aims to determine whether the person has been observed by cameras.

Dataset: We employ the Market-1501 set in our experiment. This data set contains 32,668 detected person bounding boxes of 1,501 identities (Zheng et al., 2015). Images of each identity is captured by six cameras at most, and two cameras at least. According to the data set setting, training set contains 12936 cropped images of 751 identities and testing set contains 19,732 cropped images of 750 identities. They are directly detected by Deformable Part Model (DPM) instead of hand-drawn bboxs, which is closer to the realistic setting. Each identity may have multiple images under each camera. We use the provided fixed training and test set, under both the single-query and multi-query evaluation settings.

In the experiments, 20% instances of training data with their labels are chose with labels, the rest of data are treated as unlabeled instances. Instead of directly selecting labeled samples from the whole data, we randomly sample 20% labeled samples for each class. We implemented the experiments two times under two randomly sampled training data, and their average is reported as the final result.

Experiment setting: Like the state-of-art Person re-ID model proposed by (Zheng et al., 2016), three different deep learning networks, including Alexnet, Googlenet, and Vggnet, respectively, are used to generate multi-view features for Market-1501 data set. The employed model is a new siamese network that simultaneously computes identification loss and verification loss (Zheng et al., 2016). We treat this new loss as the optimized loss in our model, and thus the re-ID task can be well handled using the SPaCo algorithm. Two combinations, Alexnet with Googlenet and Googlenet with Vggnet, are adopted in our experiments.

Over all experiments, parameters of each model are set following the training setting as (Zheng et al., 2016).

Via using co-training and self-train algorithms as comparison methods, the effectiveness of SPaCo method is validated. The training process on re-ID task is the same as the process on text classification task. Cotrade is not adopted to re-ID task since it can only handle two class problems. CoMR and RANC are also not included since they are not trained in an iterative way, and hard to be applied to the re-ID task. For the SPaCo algorithm, in every iteration, the number of selected unlabeled instances is ranged from 1000 to 2000. The lower bound is to guarantee sufficient instances for each class and the higher bound is to avoid introducting too many noisy samples.

Table 3. Rank-1 accuracy comparison of 3 competing methods on Market-1501 data set

Method	Alexnet & Googlenet			Vggnet & Googlenet		
	View1	View2	Final	View1	View2	Final
Base	0.3693	0.5371	0.5537	0.5581	0.5371	0.5811
SelfTrain	0.5118	0.6315	0.6603	0.6223	0.6315	0.6707
Cotrain	0.5353	0.6392	0.6686	0.6133	0.6625	0.6726
SPaCo	**0.5391**	**0.6657**	**0.6785**	**0.6301**	**0.6647**	**0.6825**

Experimental results: From Table 3, it is seen that rank-1 accuracies of all methods are improved since more samples are used for training. When Alexnet and Googlenet networks are adopted, SPaCo achieves the highest rank-1 accuracy not only on final result, but also on two view results, respectively. Specifically, our model achieves 66.57% rank-1 accuracy, evidently better than other competing methods. For the Vggnet and Googlenet view experiment, our SPaCo algorithm achieves a 68.26% rank-1 accuracy, which is also the best among all competing methods.

5. Conclusion and Future Work

We have proposed an improved co-training algorithm which trains the classifiers under two views of data in a serial way and alternatively updates the pseudo labels of unlabeled data to improve the performance of classifiers in each training round. We represent the algorithm with an self-paced co-training (SPaCo) model, and the optimized strategy for solving this model is consistent with the training process of an improved co-training algorithm. Experimental results verify the advantage of SPaCo beyond current co-training methods.

Research directions in our future work include designing more self-paced regularizers for SPaCo considering their different capacities on noise data. Besides, since there are many multi-view other than two view data sets in practice, we need to develop a more general SPaCo regimes to deal with multi-view tasks.

Acknowledgements

This research was supported by the China NSFC project under Grant No. 61373114, 61661166011, 11690011, 61603292, Macau Science and Technology Development Funds under Grant No. 003/2016/AFJ from the Macau Special Administrative Region of the People's Republic of China, and the National Grand Fundamental Research 973 Program of China under Grant No. 2013CB329404.

References

Abney, Steven. Bootstrapping. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pp. 360–367. Association for Computational Linguistics, 2002.

Balcan, Maria-Florina, Blum, Avrim, and Yang, Ke. Co-training and expansion: Towards bridging theory and practice. In *Advances in neural information processing systems*, pp. 89–96, 2004.

Bengio, Yoshua, Louradour, Jérôme, Collobert, Ronan, and Weston, Jason. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pp. 41–48. ACM, 2009.

Blum, Avrim and Mitchell, Tom. Combining labeled and unlabeled data with co-training. In *Proceedings of the eleventh annual conference on Computational learning theory*, pp. 92–100. ACM, 1998.

Boyd, Stephen, Parikh, Neal, Chu, Eric, Peleato, Borja, and Eckstein, Jonathan. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.

Brefeld, Ulf and Scheffer, Tobias. Co-em support vector learning. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 16. ACM, 2004.

Brefeld, Ulf and Scheffer, Tobias. Semi-supervised learning for structured output variables. In *Proceedings of the 23rd international conference on Machine learning*, pp. 145–152. ACM, 2006.

Goldman, Sally and Zhou, Yan. Enhancing supervised learning with unlabeled data. In *ICML*, pp. 327–334, 2000.

Jiang, Lu, Meng, Deyu, Mitamura, Teruko, and Hauptmann, Alexander G. Easy samples first: Self-paced reranking for zero-example multimedia search. In *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 547–556. ACM, 2014a.

Jiang, Lu, Meng, Deyu, Yu, Shoou-I, Lan, Zhenzhong, Shan, Shiguang, and Hauptmann, Alexander. Self-paced learning with diversity. In *Advances in Neural Information Processing Systems*, pp. 2078–2086, 2014b.

Jiang, Lu, Meng, Deyu, Zhao, Qian, Shan, Shiguang, and Hauptmann, Alexander G. Self-paced curriculum learning. In *AAAI*, volume 2, pp. 2694–2700, 2015.

Kumar, M Pawan, Packer, Benjamin, and Koller, Daphne. Self-paced learning for latent variable models. In *Advances in Neural Information Processing Systems*, pp. 1189–1197, 2010.

Li, Ming and Zhou, Zhi-Hua. Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, 37(6): 1088–1098, 2007.

Ma, Zilu, Liu, Shiqi, and Meng, Deyu. On convergence property of implicit self-paced objective. *CoRR*, abs/1703.09923, 2017. URL http://arxiv.org/abs/1703.09923.

Meng, Deyu and Zhao, Qian. What objective does self-paced learning indeed optimize? In *arXiv:1511.06049*, 2015.

Mitchell, Tom. *Machine Learning*. McGraw Hill, 1997.

Nigam, Kamal and Ghani, Rayid. Analyzing the effectiveness and applicability of co-training. In *Proceedings of the ninth international conference on Information and knowledge management*, pp. 86–93. ACM, 2000.

Pi, Te, Li, Xi, Zhang, Zhongfei, Meng, Deyu, Wu, Fei, Xiao, Jun, and Zhuang, Yueting. Self-paced boost learning for classification. In *IJCAI*, 2016.

Scudder, H. Probability of error of some adaptive pattern-recognition machines. *IEEE Transactions on Information Theory*, 11(3):363–371, 1965.

Sindhwani, Vikas and Rosenberg, David S. An rkhs for multi-view learning and manifold co-regularization. In *Proceedings of the 25th international conference on Machine learning*, pp. 976–983. ACM, 2008.

Sindhwani, Vikas, Niyogi, Partha, and Belkin, Mikhail. A co-regularization approach to semi-supervised learning with multiple views. In *Proceedings of ICML workshop on learning with multiple views*, pp. 74–79. Citeseer, 2005.

Supancic, James S and Ramanan, Deva. Self-paced learning for long-term tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2379–2386, 2013.

Wang, Wei and Zhou, Zhi-Hua. A new analysis of co-training. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pp. 1135–1142, 2010.

Wang, Wei and Zhou, Zhi-Hua. Co-training with insufficient views. In *ACML*, pp. 467–482, 2013.

Xu, Qian, Hu, Derek Hao, Xue, Hong, Yu, Weichuan, and Yang, Qiang. Semi-supervised protein subcellular localization. *BMC bioinformatics*, 10(1):1, 2009.

Ye, Han-Jia, Zhan, De-Chuan, Miao, Yuan, Jiang, Yuan, and Zhou, Zhi-Hua. Rank consistency based multi-view learning: A privacy-preserving approach. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pp. 991–1000. ACM, 2015.

Yu, S., Jiang, L., Mao, Z., Chang, X. J., Du, X. Z., Gan, C., Lan, Z. Z., Xu, Z. W., Li, X. C., Cai, Y., Kumar, A., Miao, Y., Martin, L., Wolfe, N., Xu, S. C., Li, H., Lin, M., Ma, Z. G., Yang, Y., Meng, D. Y., Shan, S. G., Sahin, P. D., Burger, S., Metze, F., Singh, R., Raj, B., Mitamura, T., Stern, R., and Hauptmann, A. Cmu-informediaTRECVID 2014. In *TRECVID*, 2014.

Zhang, Dingwen, Meng, Deyu, and Han, Junwei. Co-saliency detection via a self-paced multiple-instance learning framework. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015.

Zhang, Min-Ling and Zhou, Zhi-Hua. Cotrade: confident co-training with data editing. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41 (6):1612–1626, 2011.

Zhao, Qian, Meng, Deyu, Jiang, Lu, Xie, Qi, Xu, Zongben, and Hauptmann, Alexander G. Self-paced learning for matrix factorization. In *AAAI*, pp. 3196–3202, 2015.

Zheng, L., Yang, Y., and Hauptmann, A. G. Person Re-identification: Past, Present and Future. *ArXiv e-prints*, October 2016.

Zheng, Liang, Shen, Liyue, Tian, Lu, Wang, Shengjin, Wang, Jingdong, and Tian, Qi. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1116–1124, 2015.

Zheng, Zhedong, Zheng, Liang, and Yang, Yi. A discriminatively learned CNN embedding for person re-identification. *CoRR*, abs/1611.05666, 2016.

Zhou, Zhi-Hua, Zhan, De-Chuan, and Yang, Qiang. Semi-supervised learning with very few labeled training examples. In *Proceedings of the national conference on artificial intelligence*, volume 22, pp. 675. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2007.

Zhu, Xiaojin. Semi-supervised learning. In *Encyclopedia of machine learning*, pp. 892–897. Springer, 2011.