ICIC Express Letters
Volume 11, Number 1, January 2017

ICIC International ©2017 ISSN 1881-803X
pp. 109–115

INCREASING CO-TRAINING WITH PREDICTION CONFIDENCE OF NEAREST NEIGHBOR UNLABELED SAMPLES

Xianchun Zou and Xitao Zou

College of Computer and Information Science
Southwest University
No. 2, Tiansheng Road, Beibei Dist., Chongqing 400715, P. R. China
zouxc@swu.edu.cn

Received June 2016; accepted September 2016

Abstract. *As an excellent semi-supervised learning paradigm, co-training usually needs at least two sufficient and redundant views on the training datasets, which is fulfilled by few training datasets. Besides, most co-training algorithms frequently face with the introduction of noisy labels, which is harmful to the efficiency of co-training. To decrease these problems, in this paper, we increase co-training algorithm with prediction confidence of neighboring unlabeled samples, Increase-CoTrain in short. On one hand, to avoid the sufficient and redundant views in co-training, we introduce entropy-based division of views into Increase-CoTrain. On the other hand, to alleviate the introduction of noisy labels in co-training, we define a strategy of comparing the prediction confidence with its nearest neighbor unlabeled samples when labeling an unlabeled sample. Experiments on several UCI datasets demonstrate the efficiency of Increase-CoTrain.*

Keywords: Co-training, Noisy labels, Prediction confidence of nearest neighbor unlabeled samples, Entropy-based division of views

1. Introduction.

Semi-supervised learning [1], which uses both labeled samples and unlabeled samples to train and strengthen classifiers, is one of the hottest topics in machine learning. Co-training [2] is an excellent semi-supervised learning paradigm. In 1997, Blum and Mitchell first put forward the original co-training algorithm. Generally speaking, the process of co-training is as follows: Co-training algorithm generates two classifiers on two sufficient and redundant views of samples. In each iteration, a classifier gives samples which have the highest labeling confidence to the other classifier; the latter refines itself with the expanding labeled samples and gives samples which have the highest prediction confidence to its collaborative classifier. Co-training algorithm stops after reaching the termination conditions.

Co-training can make full use of quantities of unlabeled samples and build classifiers with high classification accuracy. Especially on two sufficient and redundant views of training datasets, co-training can construct efficient classifiers with a few labeled samples and abundant unlabeled samples. However, there are few training datasets who have at least two sufficient and redundant views, which makes original co-training is not applicable to real-world scenes. To solve this problem, many researchers have worked out their solutions. Goldman and Zhou [3] improved original co-training, in which two classifiers are built on the single view of training dataset by using two different decision trees. Zhou and Li [4] proposed the famous tri-training algorithm. Tri-training does not require sufficient and redundant views, by using bootstrap sampling three times, this algorithm develops three sample subsets from the original labeled sample set, and then, tri-training trains three basic classifiers from these sample subsets. Wang et al. [5] presented a co-training algorithm based on random subspace of training samples, which trains classifiers on different random subspace of attributes of samples.

The introduction of noisy labels is another serious problem for co-training. Because of the small number of labeled samples in co-training, especially in the initial stages, basic classifiers built on these labeled samples are too weak to classify unlabeled samples correctly, which may label unlabeled samples incorrectly and bring in noisy labels for co-training. To minimize this issue, pools of criteria are carried out. Among them, calculating the prediction confidence and bringing the unlabeled samples with the highest prediction confidence to the main classifier in co-training is the most popular strategy. Zhou and Li [4] firstly evaluated the prediction confidence of unlabeled samples with voting by the auxiliary classifiers. Zhang et al. [6] drew a conclusion on methods of prediction confidence in co-training. Zou et al. [7] defined a novel formula to calculate the prediction confidence.

Above all, solutions to the problem of inexistence of sufficient and redundant views do not always work well, while methods of preventing from the introduction of noisy labels limit the improvement of co-training. To lessen the introduction of noisy labels and increase the efficiency of co-training, in this paper, we combine co-training with prediction confidence of nearest neighbor unlabeled samples and put forward a co-training algorithm, Increase-CoTrain in short. Increase-CoTrain builds two classifiers on two views of training samples. The two views are generated by entropy-based division. In each iteration of Increase-CoTrain, one classifier is selected as the main classifier while the other one is regarded as the auxiliary classifier in return. The auxiliary classifier labels unlabeled samples for the main classifier. Concretely, those unlabeled samples, which not only have the highest prediction confidence but also have small difference of the prediction confidence of its nearest neighbor unlabeled samples, are labeled by the auxiliary classifier and used to the reinforcement learning of the main classifier.

The rest of this paper is organized as follows. Section 2 detailedly depicts the entropy-based division of views and the measure of calculating the prediction confidence of unlabeled samples and the difference of the prediction confidence between them and their nearest neighbor unlabeled samples. Then, Section 3 defines the process of Increase-CoTrain. After that, Section 4 shows the experiments and Section 5 draws a conclusion of this paper.

2. **Research Methods.**

2.1. **Entropy-based division of views.**

To solve the problem that there are not sufficient and redundant views of training data in co-training, Du et al. [8] proposed the method of entropy-based divisions of views. The method at first calculates the entropy of each attribute of the training data, and then, sorts all the attributes in descending order of the value of their entropy. At last, all attributes with odd index are taken as a view, while attributes with even index are regarded as another view.

Let p_i represent the probability of samples belonging to class C_i in dataset D. The value of p_i is calculated by the following formula.

$$p_i = \frac{|c_{i,D}|}{|D|} \tag{1}$$

where $|c_{i,D}|$ denotes the number of samples belonging to class C_i in dataset D and $|D|$ is the total number of samples in D.

The expectation information of classification in dataset $|D|$ can be depicted as:

$$Info(D) = -\sum_{i=1}^{|C|} p_i \log_2(p_i) \tag{2}$$

where $|C|$ represents the number of classes of samples in dataset D.

Based on this, we define the entropy of each attribute as follows.

$$Info_A(D) = \sum_{j=1}^{|V|} \frac{|D_j|}{|D|} \times Info(D_j) \tag{3}$$

where $Info_A(D)$ is the entropy of attribute A in the dataset D. $|V|$ is the number of classes of samples in D under attribute A, $|V| \leq |C|$. D_j is the number of samples belonging to class C_j under attribute A.

2.2. **Prediction confidence of nearest neighbor unlabeled samples.** We assume that an unlabeled sample $x = \{a_1, a_2, \cdots, a_m\}$, and a_i $(i = 1, 2, \cdots, m)$ is an attribute of unlabeled sample x. The label of x belongs to $C = \{C_1, C_2, \cdots, C_n\}$. Afterwards, we can calculate the $p(C_1|x)$, $p(C_2|x)$, $\cdots$, $p(C_n|x)$. If $p(C_k|x) = \max\{p(C_1|x), p(C_2|x), \cdots, p(C_n|x)\}$, $k = 1, 2, \cdots, n$, then with a high probability, the label of x is C_k. In this paper, we define the prediction confidence of sample x as $h(x) = p(C_k|x)$. As a result, the key point is to work out $p(C_1|x)$, $p(C_2|x)$, $\cdots$, $p(C_n|x)$. As we all know, if each attribute of the training data is conditional independent, then we can acquire $p(C_i|x)$ by Bayes Theorem [6, 9].

$$p(C_i|x) = \frac{p(x|C_i)p(C_i)}{p(x)} \tag{4}$$

As the denominator in Equation (4) is a constant, the question is to acquire the value of $p(x|C_i)p(C_i)$. Because each attribute of the training data is conditional independent, we can work out $p(x|C_i)p(C_i)$ with the following equation.

$$\begin{aligned} p(x|C_i)p(C_i) &= p(a_1|C_i)p(a_2|C_i)\cdots p(a_m|C_i)p(C_i) \\ &= p(C_i)\prod_{j=1}^{m} p(a_j|C_i) \end{aligned} \tag{5}$$

While $p(C_i)$ can be estimated approximately by Equation (6).

$$p(C_i) = \frac{\sum\limits_{j=1}^{|D|} p(j, C_i)}{|D|} \tag{6}$$

where $|D|$ is the total number of samples in D, $p(j, C_i)$ denotes the probability of sample x_j belonging to class C_i.

Under the theory of clustering hypothesis, if an unlabeled sample x is labeled with a high prediction confidence, while its nearest unlabeled samples have lower prediction confidence, then it might hint that x is labeled incorrect. Thereby, in order to prevent the unlabeled samples from being incorrectly labeled, when labeling unlabeled sample x_i, the prediction confidence of the nearest neighbors should be taken into account. In another word, for each unlabeled sample $x_j \in neighbor_k(x_i)$, x_i and x_j need to fulfill Inequality (7).

$$|h(x_i) - h(x_j)| \leq \varepsilon \tag{7}$$

where $h(x_i)$ is the prediction confidence of unlabeled sample x_i. $\varepsilon \to 0$ is a threshold.

3. **Increase-CoTrain.** Based on the entropy-based division of views and prediction confidence of nearest neighbor unlabeled samples in Section 2, we put forward an increasing co-training algorithm, Increase-CoTrain in short. Increase-CoTrain firstly partitioned the attributes of training dataset into two views by entropy-based division of views. Then, in each view, it generates the initial labeled samples and unlabeled samples via the projection of the training dataset on the two views respectively. After that, Increase-CoTrain

trains two classifiers on the labeled samples of the two views. In each iteration of Increase-CoTrain, each classifier labels unlabeled samples which both have the highest prediction confidence but also have small difference of the prediction confidence of their nearest neighbor unlabeled samples. When the algorithm reaches the iteration time, the two classifiers are assembled as the final classifier. The detail description of Increase-CoTrain is depicted in Algorithm 1.

Algorithm 1 Increase-CoTrain: Increasing co-training with prediction confidence of nearest neighbor unlabeled samples

```
Input:
    $L$: Original labeled sample set;
    $U$: Unlabeled sample set;
    $Learner$: Learning algorithm;
    $IterNum$: the iteration time of algorithm;
    $k$: the number of the nearest neighbors of the choosing unlabeled sample;
    $\varepsilon$: threshold of the prediction confidence between the choosing unlabeled sample and its
    neighbor.
Output:
    Dividing the attributes of training dataset into two views $V_1$, $V_2$ by Equations (1), (2), (3)
    Splitting $L$, $U$ respectively into $V_1$, $V_2$ and generating $L_{V_1}$, $L_{V_2}$, $U_{V_1}$, $U_{V_2}$
    for $t = 1$ to $2$ do
        $classifier_t$ = Learner($L_{V_t}$)
    end for
    for $iter = 1$ to $IterNum$ do
        for $t = 1$ to $2$ do
            $ta$ = mod(t,2) + 1
            $U_{xy} = \emptyset$, $L_{xy} = \emptyset$
            for $i = 1$ to $|U_{V_{ta}}|$ do
                Labeling $x_i$ by $classifier_{ta}$ and adding $x_i$ with its label into $U_{xy}$
                Calculating the prediction confidence of $x_i$ by Equations (4), (5), (6)
            end for
            Removing samples with the highest prediction confidence from $U_{xy}$ to $L_{xy}$
            for $i = 1$ to $|L_{xy}|$ do
                $f$ = false
                for $j = 1$ to $|neighbor_k(x_i)|$ do
                    if $!(|h(x_i) - h(x_j)| \leq \varepsilon)$ then
                    $f$ = true
                    end if
                end for
                if $f$ == false then
                Taking $x_i$ and its label into $L_{V_t}$
                Deleting $x_i$ from $U_{V_{ta}}$
                end if
            end for
        end for
        for $t = 1$ to $2$ do
            $classifier_t$ = Learner($L_{V_t}$)
        end for
    end for
    $classifier \leftarrow$ Assemble($classifier_1$&$classifier_2$)
    return $classifier$
```

4. **Experiments.** In this part, we experiment on several datasets from UCI [10] to examine the performance of Increase-CoTrain. The detail information of each dataset is delineated in Table 1.

Table 1. Experimental data sets

data set	#features	#samples	#class	#pos/#neg
colic	22	368	2	63.0%/37.0%
hypothyroid	25	3163	2	4.8%/95.2%
sick	29	3772	2	6.1%/93.9%
wdbc	30	569	2	37.3%/62.7%

In experiments, for each experimental dataset, we randomly take 80% samples as training data while the rest as test data. In training data, 20% samples are selected as initial labeled data and the rest as unlabeled data. The learning algorithm to train classifiers for each algorithm in experiments is BP neural network. In experiments, to overcome the random results, each algorithm is trained 20 times on each dataset and the average results are regarded as the final experimental results. Furthermore, we set $IterNum = 6$ and $k = 20$ in experiments.

4.1. **Investigating sensitivity of Increase-CoTrain with threshold ε.** To investigate the sensitivity of Increase-CoTrain with threshold ε, we set $\varepsilon = 0.01$, 0.05, 0.10, 0.15 respectively. With each ε, we train Increase-CoTrain on each experimental dataset and record the corresponding classification error rate of classifiers trained on Increase-CoTrain in each iteration. The average classification error rates on the four experimental datasets are explicitly depicted in Figure 1.

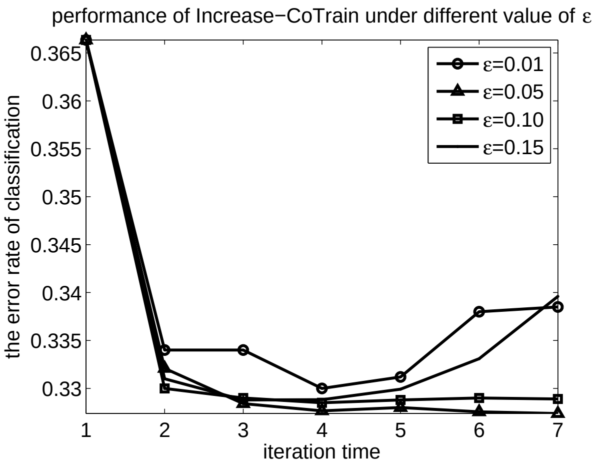

Figure 1. Performance of increase-CoTrain with different threshold ε

From Figure 1, it can be obviously observed that, when $\varepsilon = 0.05$ or $\varepsilon = 0.10$, classifiers trained on Increase-CoTrain can achieve a lower classification error rate. This fact can be attributed to the following reasons. In Increase-CoTrain, when ε is too small, it is difficult to seek out unlabeled samples which can fulfill Inequation (7); as a result, few unlabeled samples with highest prediction confidence can be labeled, which clearly prevent from strengthening the main classifier. On the contrary, when ε gets a big value, it is easy to fulfill Inequation (7), as long as a number of unlabeled samples with highest prediction confidence are labeled, many noisy labels may be introduced, which also makes bad effect on the performance of Increase-CoTrain.

4.2. **Comparing Increase-CoTrain with relative co-training algorithms.** In the second experiment, aiming at validating the performance of Increase-CoTrain, we compare it with co-training in paper [2], NoNeighbor-CoTrain (Increase-CoTrain without the consideration of the prediction confidence of the nearest neighbor unlabeled samples). We set $\varepsilon = 0.05$ and write down the classification error rate of each comparing algorithm in each iteration. The results on each of four datasets are presented in Figure 2.

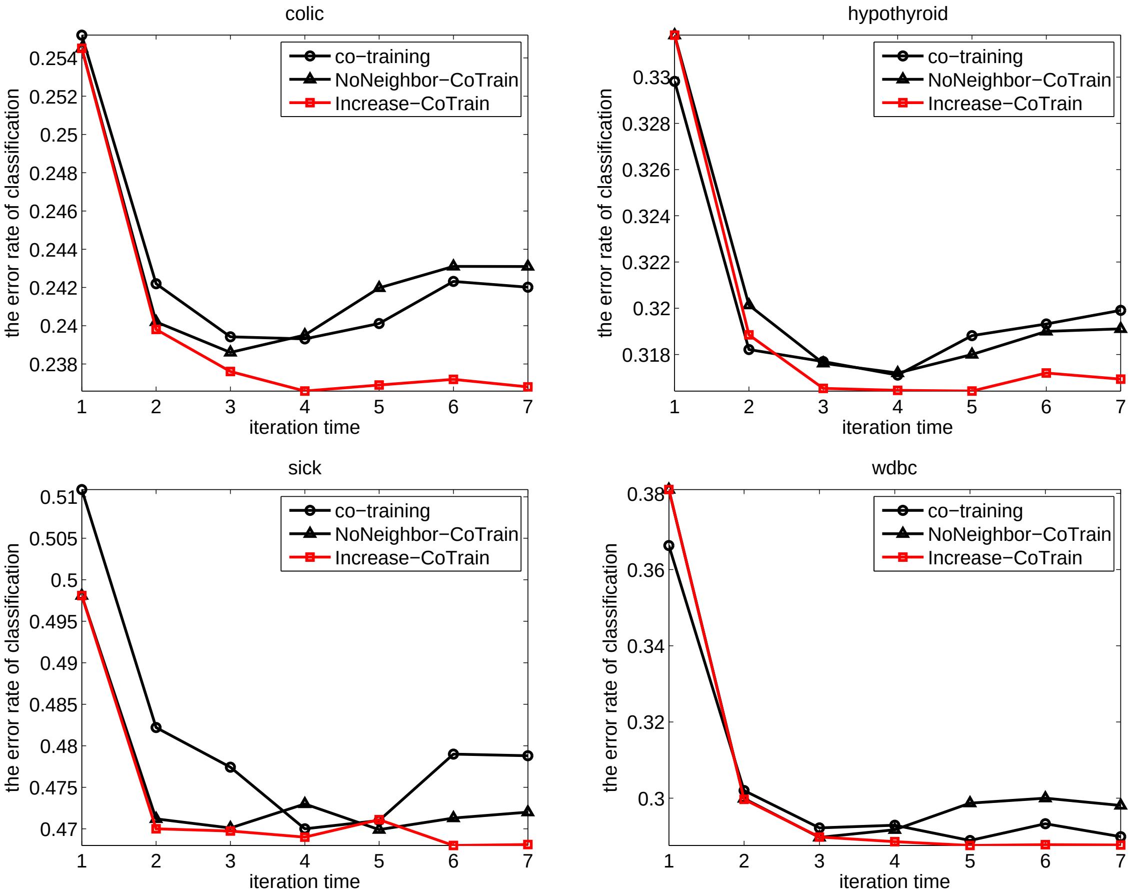

FIGURE 2. Classification error rate of co-training, NoNeighbor-CoTrain, Increase-CoTrain

From Figure 2, we can know that Increase-CoTrain always performs better than its comparing algorithms on classification accuracy. Meanwhile, with the increment of the $IterNum$, Increase-CoTrain is more stable than co-training and NoNeighbor-CoTrain, which demonstrates the efficiency of the introduction of the prediction confidence of the nearest neighbor unlabeled samples.

5. **Conclusion.** Above all, this paper is a discussion about co-training algorithm in machine learning. Specifically, in this paper, we introduce entropy-based division of views into co-training to solve the problem that most training datasets are lack of sufficient and redundant views; besides, the prediction confidence of the nearest neighbor unlabeled samples is taken into account in co-training to prevent the introduction of noisy labels. The experiments validate the efficiency of our proposed algorithm. As the algorithm is a bit complicate, our future work is to simplify the algorithm by finding novel methods.

Acknowledgment. The authors gratefully acknowledge the helpful comments and suggestions of the reviewers, which have improved the presentation.

REFERENCES

[1] X. Zhu, Semi-supervised learning literature survey, *Technical Report 1530*, Department of Computer Sciences, University of Wisconsin-Madison, 2008.

[2] A. Blum and T. Mitchell, Combining labeled and unlabeled data with co-training, *Proc. of the 11th Annual Conference on Computational Learning Theory*, pp.92-100, 1998.

[3] S. Goldman and Y. Zhou, Enhancing supervised learning with unlabeled data, *Proc. of the 17th International Conference on Machine Learning*, San Francisco, CA, pp.327-334, 2000.

[4] Z. Zhou and M. Li, Tri-training: Exploiting unlabeled data using three classifiers, *IEEE Trans. Knowledge and Data Engineering*, vol.17, no.11, pp.1529-1541, 2005.

[5] J. Wang, S. Luo and X. Zeng, A random subspace method for co-training, *Proc. of the International Joint Conference on Neural Networks*, pp.195-200, 2008.

[6] Y. Zhang, J. Wen, X. Wang and Z. Jiang, Semi-supervised learning combining co-training with active learning, *Expert Systems with Applications*, vol.41, no.5, pp.2372-2378, 2014.

[7] X. T. Zou, X. C. Zou, G. X. Yu and X. H. Fu, Enhancing co-training algorithms by sample representativeness, *Journal of Computational Information Systems*, vol.10, no.16, pp.6883-6890, 2014.

[8] J. Du, X. Ling and Z. Zhou, When does co-training work in real data? *IEEE Trans. Knowledge and Data Engineering*, vol.23, no.5, pp.788-799, 2011.

[9] K. Nigam, A. McCallum, S. Thrun et al., Text classification from labeled and unlabeled documents using EM, *Machine Learning*, vol.39, nos.2-3, pp.103-134, 2000.

[10] A. Asuncion and D. J. Newman, *UCI Repository of Machine Learning Databases*, School of Information and Computer Science, University of California, Irvine, 2007.