Confidence Estimation for Graph-based Semi-supervised Learning

Tao Guo
Visual Computing and Visual Reality Key Laboratory of Sichuan Province, Chengdu, China
Email: tguo35@gmail.com

Guiyang Li
College of Computer Science, Sichuan Normal University, Chengdu, China
Email: guiyang.li@gmail.com

***Abstract*—To select unlabeled example effectively and reduce classification error, confidence estimation for graph-based semi-supervised learning （CEGSL） is proposed. This algorithm combines graph-based semi-supervised learning with collaboration-training. It makes use of structure information of sample to calculate the classification probability of unlabeled example explicitly. With multi-classifiers, the algorithm computes the confidence of unlabeled example implicitly. With dual-confidence estimation, the unlabeled example is selected to update classifiers. The comparative experiments on UCI datasets indicate that CEGSL can effectively exploit unlabeled data to enhance the learning performance.**

***Index Terms*—graph, collaboration-training, confidence, classification, semi-supervised leaning,**

I. INTRODUCTION

Applications such as web search, pattern recognition, text classification, genetic research are examples where cheap unlabeled data can be added to a pool of labeled samples. In these applications, a large amount of labeled data should be available for building a model with good performance. During past decade, many supervised learning algorithms (e.g. J4.8, Bays and SVM) have been developed and extensively learned use labeled data. Unfortunately, it is often the case that there is a limited number of labeled data along with a large pool of unlabeled data in many practices [1]. It is noteworthy that a number of methods called semi-supervised learning have been developed for using unlabeled data to improve the accuracy of prediction [2]. It has received considerable attention in the machine learning literature due to its potential in reducing the need for expensive labeled data. Early methods in semi-supervised learning were using mixture models and extensions of the EM algorithm [3]. More recent approaches belong to one of the following categories: self-training, transductive SVMs, co-training, split learning, and graph-based methods [4].

Co-training is a prominent approach in semi-supervised learning proposed by Blum and Mitchell [5]. It requires two sufficient and redundant views to learning [6]. In this algorithm, it assumes that the description of each sample set can be divided into two distinct subsets. Each of the subsets is sufficient for learning if there is sufficient labeled example. Then the two subsets are conditionally independent given the class attribute. Two classifiers iteratively trained on one subset and they teach each other with a respective subset of unlabeled example and their highest confidence predictions. Since co-training requires two sufficient and redundant views, such a requirement can hardly be met in most scenarios [7]. Goldman and Zhou proposed an improved co-training algorithm [8]. It employs time-consuming cross validation technique to determine how to label the unlabeled examples and how to produce the final hypothesis [9]. In 2005, Zhou and Li proposed a new co-training style algorithm named tri-training [10]. It is easy to be applied to common data mining application. However, the performance of this algorithm goes degradation in some circumstances and exists three issues: (1) estimation for classification error is unsuitable. (2) excessively confined restriction introduce more classification noise. (3) differentiation between initial labeled example and labeled unlabeled example is deficient [11]. Zhan [12] proposed an algorithm called co-training semi-supervised active learning with noise filter. In this algorithm, three fuzzy buried Markov models are used to perform semi-supervised learning cooperatively. Some human-computer interactions are actively introduced to label the unlabeled sample at certain time. The experimental results show that the algorithm can effectively improve the utilization of unlabeled samples, reduce the introduction of noise samples and raise the accuracy of expression recognition. But human interaction will reduce the efficiency of the algorithm. In this paper, an explicit confidence estimation for graph-based semi-supervised learning algorithm (CEGSL) is proposed. This algorithm makes use of structure of sample data to calculate the classification probability of unlabeled example explicitly. Combining with co-training, this algorithm computes the confidence of unlabeled example implicitly with three classifiers and to select unlabeled example efficiently.

The rest of the paper is structured as follows: Section 2 describes graph-based semi-supervised learning. Section

doi:10.4304/jsw.7.6.1307-1314

3 introduces the proposed algorithms. Section 4 shows experimental and comparative results in different UCI data sets. Section 5 makes concludes.

II. GRAPH-BASED SEMI-SUPERVISED LEARNING

Graph-based semi-supervised learning algorithm makes use of example sets and similarity to create a diagram. The nodes in the graph correspond to example. The weight of edge represents similarity that connects two examples. Graph-based semi-supervised learning problem is a regular optimization problem. Definition of the problem includes the objective function needed to optimize and regular items defined by decision function. It solves the problem by optimizing the parameters of optimal model. Decision function for the model has two properties: (1) the output label from unlabeled example tries to match that from labeled example. (2) the whole graph satisfies smoothness. Graph-based semi-supervised learning algorithm uses the popular assumption directly or indirectly. The assumption requires similar labels in a small local region and it also reflects local smoothness of decision function. Under this assumption, a large number of unlabeled examples make the space of example more compact, thus it can indicate characteristic of local region more accurately and makes the decision function fit the data better.

The target function of graph-based semi-supervised learning algorithm includes two parts, loss function and regular items. Different algorithm selects different loss function and regular item. Zhu X J[13] proposed a semi-supervised learning algorithm with harmonic function of Gaussian random occasions in 2003. This method is a continuous relaxation method for discrete Markov. The loss function in objective function is a quadratic function with infinite weight. Regular item is a combinational Laplacian based on graph. Although a variety of graph-based semi-supervised learning algorithm set the objective function differently, they can be concluded to formula (1)

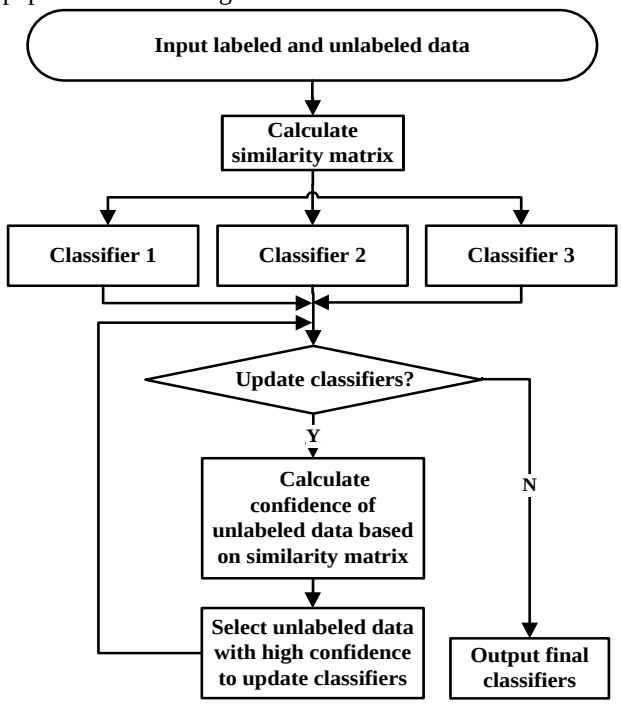
$$F(y)=\sum_{i=1}^{n}\sum_{j=1}^{n}w_{i,j}(y_i-y_j)^2 \qquad (1)$$

Where *y* represents prediction labels for unlabeled examples, $w_{i,j}$ represents matrix of weight in graph. The objective of graph-based semi-supervised learning is to optimize $F(y)$ and obtain optimal parameter of model.

III. CONFIDENCE ESTIMATION FOR GRAPH-BASED SEMI-SUPERVISED LEARNING

CEGSL algorithm combines the advantages of semi-supervised learning and collaboration-training algorithms. It uses three classifiers to perform collaborative training and compare the confidence of unlabeled examples implicitly. In order to select more reliable unlabeled example to join to training set, it makes use of structure information of examples to calculate the classification probability of unlabeled examples explicitly.

The flow diagram of the algorithm proposed in this paper is shown in Figure 1.



The training stage of CEGSL

A.. Description of CEGSL Algorithm

Given data set $R=\{\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_n\}$, it includes labeled and unlabeled examples. Assuming n_l in R are labeled examples, its data set $Y_l=\{\boldsymbol{y}_{l1}, \boldsymbol{y}_{l2,}\cdots, \boldsymbol{y}_{ln_l}\}$; $n_u=n-n_l$ are unlabeled examples and its data set $Y_u=\{\boldsymbol{y}_{u1}, \boldsymbol{y}_{u2,}\cdots, \boldsymbol{y}_{un_u}\}$.The entire data set $Y=\{Y_l, Y_u\}$. CEGSL algorithm consists of following steps. First, reading examples to built a graph with labeled and unlabeled examples as vertex and the similarity between examples as edge. Then, re-sampling labeled example set L with Bootstrap to built initialized training set for three classifiers. For each classifier, the other two classifiers are auxiliary classifiers in each iteration. They classify the examples which are in unlabeled example set U and put the identified examples and their labels into a buffer. The confidence is calculated explicitly using the graph. The unlabeled examples with high confidence are put into training set. The main classifier is adjusted until the classification errors of the three classifiers are not reduced. Finally, the algorithm is terminated. Figure 2 shows the procedure of CEGSL algorithm.

The labeled example used by CEGSL is defined as $L=\{(\boldsymbol{x}_1, \boldsymbol{y}_1),(\boldsymbol{x}_2, \boldsymbol{y}_2),\cdots,(\boldsymbol{x}_{|L|}, \boldsymbol{y}_{|L|})\}$, $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ represents that the label for example $\boldsymbol{x}_i$ is $\boldsymbol{y}_i$ ($\boldsymbol{y}_i \in \{1,-1\}$). A large number of unlabeled example is defined as $U=\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_{|U|}\}$, $|L| \square |U|$. Sampling the labeled example and initialized the three classifiers, we get three classifiers h_i （$1 \le i \le 3$）. The $buffer(i)$ （$1 \le i \le 3$）are used to save the unlabeled examples with same voting

by other two auxiliary classifiers. There are two requirements for terminating the algorithm: number of iterations is greater than specified number *K* or classifier error rata e_i increases.

input：labeled example set L，unlabeled example set U，iteration number K

output：final classifiers h_i （$1 \le i \le 3$）

1. calculating the similarity between any two examples in labeled and unlabeled example set
2. randomly sampling three data sets from labeled data set for initializing classifiers h_i
3. calculating p_i, q_i ,label z_i and confidence $|p_i - q_i|$ for each unlabeled example
4. for each classifier, the rest of two are used as auxiliary classifier to vote. The unlabeled data with same voting are put into $buffer(i)$
5. updating the classifiers with the unlabeled example, which has a high confidence $|p_i - q_i|$
6. terminating algorithm when number of iterations is greater than specified number *K* or classifier error rata e_i increases, otherwise returning to step 3

Figure 1. The procedure of explicit confidence estimation for graph-based semi-supervised learning algorithm

In the algorithm, step3 and stet4 keep the quality of selected unlabeled examples; step5 performs the selection of unlabeled example. When selecting the number of unlabeled examples, if more unlabeled examples are selected, it will increase the introduction of possibility of noise. If the selected example set is small, the convergence rate will be affected. After repeated experiments, the algorithm takes top 10% unlabeled examples to help the training of classifiers with better achievement. Also, the number of iterations *K* is set to 20 in this experiment. Since the calculation for classification error of unlabeled example is more different, this paper assumes that there is a same distribution for labeled and unlabeled examples. The classification error rate e_i is defined as number of error classification for labeled examples/number of same labeled examples. Similarity S_{ij} is defined as: $\boldsymbol{S}_{i,j} = \exp(\|x_i - x_j\|^2 / \sigma^2)$, in which σ is a constant and the RBF is used to calculate the similarity.

B. Graph-based explicit confidence estimation for unlabeled examples

Graph-based semi-supervised learning is an important breach in the research of semi-supervised learning. Representative algorithm includes Label Propagation Algorithm [14] and Graph Mincut Algorithm[15].It uses graph to present the relationship between data, nodes in the graph present examples and edges between nodes present the similarity between examples. Then, the algorithm searches the labels for unlabeled example by minimizing the labels and the inconsistency of the graph. The inconsistency is defined as:

$$\mathrm{F(y)}=\sum_{i=1}^{n}\sum_{j=1}^{n}\boldsymbol{S}_{i,j}(y_i - y_j)^2 = Y^T L Y \qquad (2)$$

Where $\boldsymbol{S}_{i,j}$ is the similarity matrix with $n*n$. L represents non-normalized graph Laplacian. For a graph constructed by labeled and unlabeled examples, the label for unlabeled example is calculated by minimizing $\mathrm{F}(\boldsymbol{y})$.

Since the regular graph-based semi-supervised learning can only calculate the labels for unlabeled example directly, this paper modifies the algorithm by referencing [15].The target function $F(\boldsymbol{S}, \boldsymbol{y})$ includes two parts, one is used for calculating the inconsistence $F_l(\boldsymbol{S}, \boldsymbol{y})$ between labeled and unlabeled examples, the other is used to compute the inconsistency $F_u(\boldsymbol{S}, \boldsymbol{y}_u)$ between unlabeled examples. There are two criteria needed to be satisfied when distributing label for unlabeled example: (1) the two unlabeled examples with high similarity have the same label. (2) the unlabeled examples own same label with the labeled example when they have high similarity with labeled example. The inconsistency $F_u(\boldsymbol{S}, \boldsymbol{y})$ is defined as:

$$F_u(\boldsymbol{S}, \boldsymbol{y}_u) = \sum_{i,j=1}^{n_u}\boldsymbol{S}_{i,j}(\boldsymbol{y}_{iu} - \boldsymbol{y}_{ju})^2 \qquad (3)$$

The inconsistency $F_l(\boldsymbol{S}, \boldsymbol{y})$ between labeled and unlabeled examples is defined as:

$$F_l(\boldsymbol{S}, \boldsymbol{y}) = \sum_{i=1}^{n_l}\sum_{j=1}^{n_u}\boldsymbol{S}_{i,j}(\boldsymbol{y}_{il} - \boldsymbol{y}_{ju})^2 \qquad (4)$$

then, the target function is defined as:

$$F(\boldsymbol{S}, \boldsymbol{y}) = F_l(\boldsymbol{S}, \boldsymbol{y}) + CF_u(\boldsymbol{S}, \boldsymbol{y}_u) \qquad (5)$$

Where C is a constant and used to evaluate the importance of F_u . When minimizing $F(\boldsymbol{S}, \boldsymbol{y})$, a suitable label can be found. $h(\boldsymbol{x}_i)$ represents prediction label for $\boldsymbol{x}_i$, then the target function is:

$$\min F(\boldsymbol{S}, y) \;\; s.t. \;\; h(\boldsymbol{x}_i) = \boldsymbol{y}_{li}, i = 1, 2, \cdots, n_l \qquad (6)$$

Put formula (4) and (5) into (6), the target function is expressed as formula (7)

$$\min F(\boldsymbol{S}, \boldsymbol{y}) = \min \sum_{i=1}^{n_l}\sum_{j=1}^{n_u}\boldsymbol{S}_{i,j}(\boldsymbol{y}_{il} - \boldsymbol{y}_{ju})^2 + C\sum_{i,j=1}^{n_u}\boldsymbol{S}_{i,j}(\boldsymbol{y}_{iu} - \boldsymbol{y}_{ju})^2$$

$$s.t. \quad h(\boldsymbol{x}_i) = \boldsymbol{y}_{li}, i = 1, 2, \cdots, n_l \qquad (7)$$

To calculate the confidence of unlabeled example, formula (7) is modified to (8)

$$\min F(\boldsymbol{S}, \boldsymbol{y}) = \sum_{i=1}^{n_u} (p_i + q_i) \qquad (8)$$

Where

$$p_i = \sum_{j=1}^{n_l} \boldsymbol{S}_{i,j} (h_i - \boldsymbol{y}_j)^2 \partial(\boldsymbol{y}_j, 1) + \frac{C}{2} \sum_{j=1}^{n_u} \boldsymbol{S}_{i,j} (h_i - h_j)^2 \qquad (9)$$

$$q_i = \sum_{j=1}^{n_l} \boldsymbol{S}_{i,j} (h_i - \boldsymbol{y}_j)^2 \partial(\boldsymbol{y}_j, -1) + \frac{C}{2} \sum_{j=1}^{n_u} \boldsymbol{S}_{i,j} (h_i - h_j)^2 \qquad (10)$$

When $\boldsymbol{x} = \boldsymbol{y}$, $\partial(\boldsymbol{x}, \boldsymbol{y})$=1 or $\partial(x, \boldsymbol{y})$=0.

p_i and q_i are calculated through formula (9) and (10). p_i and q_i represents confidences belonging to different labels for unlabeled example $\boldsymbol{x}_i$ respectively. The label of unlabeled example is computed by using $sign(p_i - q_i)$. The confidence for this label is $|p_i - q_i|$.

IV. EXPERIMENTS AND ANALYSIS

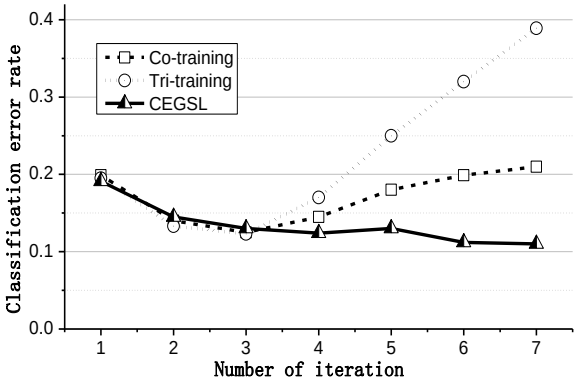
Four UCI data sets are used in this experiment. Detailed information on these data sets are tabulated in Table I .The data set used in the comparative experiments includes two sets of credit card data sets-Australian and German; two sets of medical diagnostic data set-breast-cancer and diabetes.

TABLE I. *Basic* INFORMATION *for data sets*

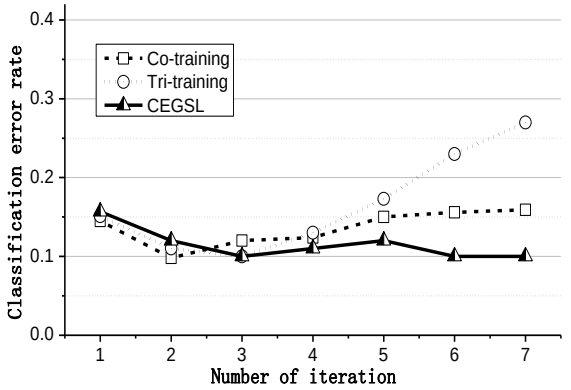
	Australian	German	Breast-cancer	Diabetes
size	690	1000	699	768
attribute	14	20	11	8
class	2	2	2	2

For each data set, about 25% data are kept as test examples while the rest are used as the pool of training examples. L and U are partitioned under different unlabeled rates including 20%, 40%, 60%, 80%. For example, assuming a set contains 1000 examples, 250 examples are used as test examples. The rest of 750 examples are kept as training examples. When the unlabeled rate is 20%, 600 examples are put into L with their labels while the remaining 150 examples are put into U without their labels. The experiment will compare the performance under different percentage of training data.

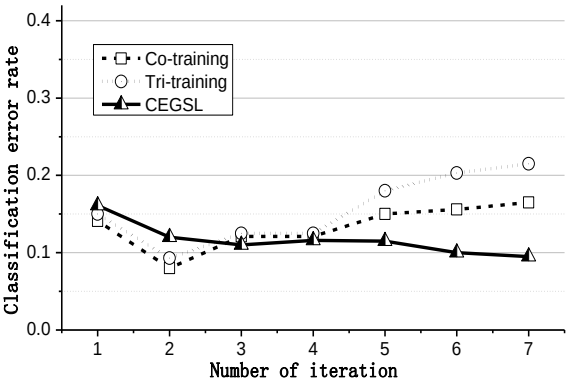
The experiment includes two groups. It takes BP neural networks and ID3 decision tree as a classifier respectively, the performance of CEGSL algorithm is compared with two semi-supervised learning algorithms, i.e. Tri-training and Co-training.



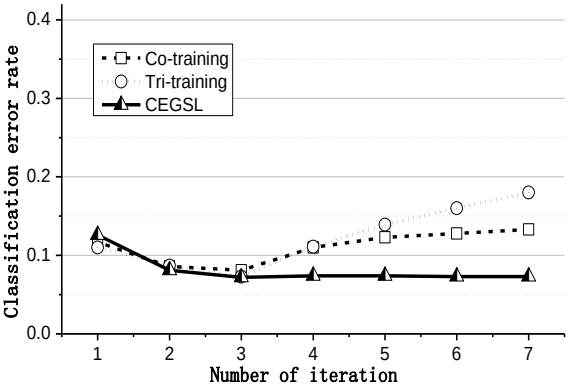
(a) 80% unlabel rate



(b) 60% unlabel rate

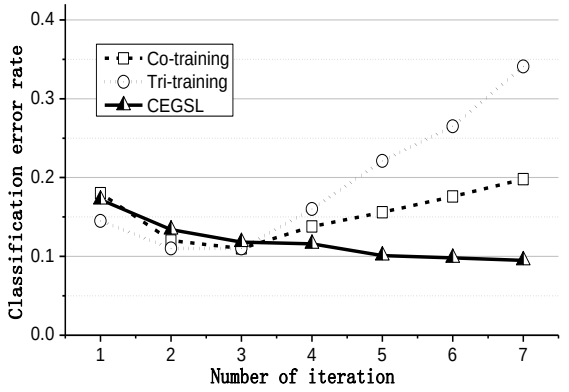


(c) 40% unlabel rate

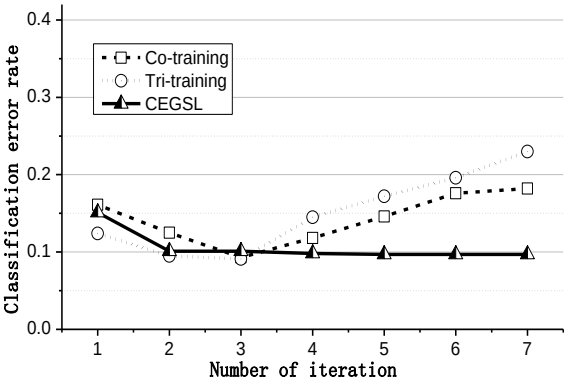


(d) 20% unlabel rate

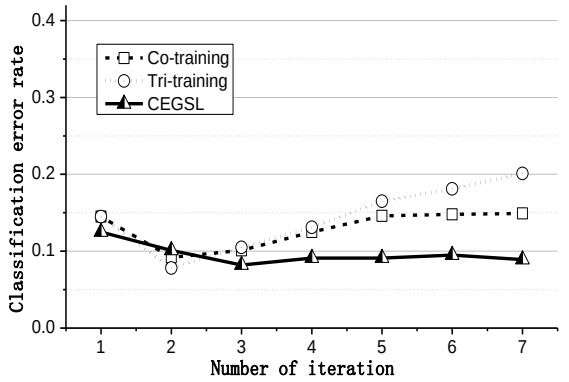
Figure 2. Average classification error rate comparison with BP neural network



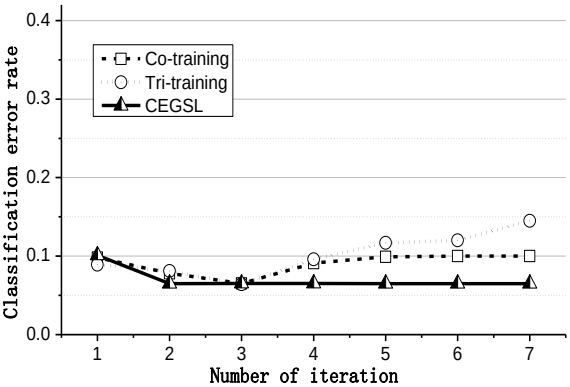
(a) 80% unlabel rate



(b) 60% unlabel rate



(c) 40% unlabel rate



(d) 20% unlabel rate

Figure 3. Average classification error rate comparison with ID3 decision tree

Figure 3 and Figure 4 give the plots of the average classification error rates versus the learning iterations before the algorithm stops. The error rates of the compared algorithms are also depicted in Figure 3 and Figure 4. The semi-supervised learning algorithms, three single classifiers with BP neural network and ID3 decision tree are trained from only the labeled training examples, i.e. L. The average error rate of the single classifiers is shown as a vertical line in each figure, the iteration number for each algorithms is shown as a horizontal line.

In detail, Figure 3(a)~(d) show the average of classification error rate during the iterative process when BP neural networks is used under all data sets. From the results, we may see that the tri-training can effectively reduce the classification error only in the first two or three rounds. With the further iterations, the classification error rate has a greater increasing. Since there is no effective way to prevent the introduction of noise data, the noise data will continue to accumulate during the iteration of the algorithm. Therefore, it will give a negative impact on tri-training, especially in the case of less labeled example [16]. Moreover, when the co-training is used, the introduction of noise data can be prevented in a certain extent by using the 10 cross-validation. Figure 3 reveals that on all the subfigures, the final hypotheses generated by CEGSL are better than the initial hypotheses. Comparing with the other two algorithms, the final hypotheses of CEGSL are almost always better after first two or three iterations.

When ID3 decision tree is used, Figure 4 (a)~(d) also show the average of classification error rate during the iterative process. It could be observed from the figures that the line of CEGSL is always below those of the other compared algorithms after first two or three rounds. But, the error rate of CEGSL keeps on decreasing when utilizing more unlabeled example, and converges quickly within just a few learning iterations. From subfigure (a) to (d), CEGSL keeps comparable with all the classifiers under all the unlabel rates

From Figure 3 to Figure 4, we may found that on all the subfigures, the final hypotheses generated by CEGSL are better than the initial hypotheses. It confirms that CEGSL can effectively exploit unlabeled examples to enhance the learning performance.

The comparative results are also summarized in Table II to Table V, which present the classification error rate of hypothesis, the final hypothesis generated by CEGSL and the improvement of the latter over the former under 80% , 60%, 40%, 20% unlabel rate. The biggest improvements achieved by each algorithm have been boldfaced in Tables.

Tables II to VI show that CEGSL algorithm can effectively improve the hypotheses with BP neural network and ID3 decision tree under all the unlabel rates. In fact, if the improvements are averaged across all the data sets, classifiers and unlabel rates, it can be found that the average improvement of CEGSL is 5.33% with BP neural network and 4.65% with ID3 decision tree. It is impressive that with all the classifiers and under all the

unlabel rates, CEGSL achieved the biggest average improvement. Moreover, Tables II to V also show that if the algorithms are compared through counting the number of winning data sets, CEGSL is almost always the winner.

In detail, under 80% unlabel rate, CEGSL has 4 wining data sets when BP neural network is used; when ID3 decision tree is used CEGSL has 3 wining data sets while co-training has 1 wining data set. Under 60% unlabel rate, when BP neural network and ID3 decision tree are used, CEGSL has 3 wining data sets respectively while Tri-training has 1 wining data set respectively; Under 40% unlabel rate, CEGSL has 4 wining data sets when BP neural network is used; when ID 3 decision tree is used, CEGSL has 3 wining data sets while co-training has 1 wining data sets. Under 20% unlabel rate, when BP neural network is used, CEGSL only has 2 wining data sets while co-training has 2 wining data sets; when ID3 decision tree is used CEGSL has 3 winging data sets and co-training has 1 wining data set.

TABLE II. THE CLASSIFICATION ERROR RATES OF THE INITIAL AND FINAL HYPOTHESES AND THE CORRESPONDING IMPROVEMENTS OF CEGSL, TRI-TRAINING AND CO-TRAINING UNDER 80% UNLABEL RATE

Data set	BP								
	CEGSL			Tri-training			Co-training		
	initial	final	improv	initial	final	improv	initial	final	improv
Australian	18.62	13.72	**4.9**	16.83	15.27	1.56	17.26	14.28	2.98
German	22.16	17.28	**4.9**	17.92	16.22	1.7	19.37	16.33	3.04
Breast-cancer	19.27	12.22	**7.1**	18.27	16.38	1.89	18.22	14.57	3.65
Diabetes	14.26	9.77	**4.5**	13.21	10.37	2.84	15.37	11.26	4.11
average	18.58	13.25	**5.33**	16.56	14.56	2.00	17.56	14.11	3.45
Data set	ID3								
	CEGSL			Tri-training			Co-training		
	initial	final	improv	initial	final	improv	initial	final	improv
Australian	17.25	13.72	3.5	17.35	14.37	2.98	18.26	14.26	**4**
German	20.01	14.08	**5.9**	18.33	15.33	3	16.27	14.23	2.04
Breast-cancer	18.97	12.36	**6.6**	19.21	17.39	1.82	17.35	14.27	3.08
Diabetes	12.31	9.77	**2.5**	12.67	10.27	2.4	12.36	11.75	0.61
average	17.14	12.48	**4.65**	16.89	14.34	2.55	16.06	13.62	2.43

TABLE III. THE CLASSIFICATION ERROR RATES OF THE INITIAL AND FINAL HYPOTHESES AND THE CORRESPONDING IMPROVEMENTS OF CEGSL, TRI-TRAINING AND CO-TRAINING UNDER 60% UNLABEL RATE

Data set	BP								
	CEGSL			Tri-training			Co-training		
	initial	final	improv	initial	final	improv	initial	final	improv
Australian	15.23	9.27	**6**	14.53	13.37	1.16	16.25	12.97	3.28
German	17.16	11.39	**5.8**	20.55	18.66	1.89	19.33	17.66	1.67
Breast-cancer	15.79	12.63	**3.2**	15.79	13.28	2.51	16.76	15.32	1.44
Diabetes	10.33	7.95	2.4	11.27	8.25	**3.02**	11.37	9.26	2.11
average	14.63	10.31	**4.32**	15.54	13.39	2.15	15.93	13.80	2.13
Data set	ID3								
	CEGSL			Tri-training			Co-training		
	initial	final	improv	initial	final	improv	initial	final	improv
Australian	15.33	10.33	**5**	15.07	14.09	0.98	15.37	13.98	1.39
German	16.89	12.67	4.2	21.97	17.38	**4.59**	18.39	16.27	2.12
Breast-cancer	17.21	10.27	**6.9**	16.33	12.33	4	19.25	14.33	4.92
Diabetes	10.31	7.95	**2.4**	10.78	9.72	1.06	12.36	10.97	1.39
average	14.94	10.31	**4.63**	16.04	13.38	2.66	16.34	13.89	2.46

TABLE IV. THE CLASSIFICATION ERROR RATES OF THE INITIAL AND FINAL HYPOTHESES AND THE CORRESPONDING IMPROVEMENTS OF CEGSL, TRI-TRAINING AND CO-TRAINING UNDER 40% UNLABEL RATE

Data set	BP								
	CEGSL			Tri-training			Co-training		
	initial	final	improv	initial	final	improv	initial	final	improv
Australian	12.53	9.28	**3.3**	11.27	10.28	0.99	12.76	11.27	1.49
German	16.79	11.98	**4.8**	18.25	15.26	2.99	17.62	16.27	1.35
Breast-cancer	14.28	9.63	**4.7**	14.38	13.72	0.66	15.73	12.37	3.36
Diabetes	10.03	6.72	**3.3**	9.26	7.05	2.21	9.68	8.29	1.39
average	13.41	9.40	**4.01**	13.29	11.58	1.71	13.95	12.05	1.90
Data set	ID3								
	CEGSL			Tri-training			Co-training		
	initial	final	improv	initial	final	improv	initial	final	improv
Australian	12.62	9.87	2.8	10.73	8.37	2.36	12.33	9.29	**3.04**
German	14.38	10.26	**4.1**	17.28	15.38	1.9	17.95	14.27	3.68
Breast-cancer	15.35	11.29	**4.1**	15.79	14.27	1.52	15.28	14.39	0.89
Diabetes	9.27	6.79	**2.5**	10.32	8.27	2.05	11.37	9.37	2
average	12.91	9.55	**3.35**	13.53	11.57	1.96	14.23	11.83	2.4

TABLE V. THE CLASSIFICATION ERROR RATES OF THE INITIAL AND FINAL HYPOTHESES AND THE CORRESPONDING IMPROVEMENTS OF CEGSL, TRI-TRAINING AND CO-TRAINING UNDER 20% UNLABEL RATE

Data set	BP								
	CEGSL			Tri-training			Co-training		
	initial	final	improv	initial	final	improv	initial	final	improv
Australian	12.39	10.27	**2.1**	13.76	12.53	1.23	11.92	10.22	1.7
German	13.05	9.38	**3.7**	14.32	12.09	2.23	14.38	12.75	1.63
Breast-cancer	11.76	10.34	1.4	12.68	11.06	1.62	12.25	10.29	**1.96**
Diabetes	9.26	7.28	2	9.59	9.07	0.52	10.39	8.17	**2.22**
average	11.62	9.32	**2.30**	12.59	11.19	1.40	12.24	10.36	1.88
Data set	ID3								
	CEGSL			Tri-training			Co-training		
	initial	final	improv	initial	final	improv	initial	final	improv
Australian	10.75	7.39	**3.4**	12.97	11.29	1.68	11.25	10.27	0.98
German	13.27	10.28	**3**	13.28	12.05	1.23	12.33	11.79	0.54
Breast-cancer	11.82	9.25	**2.6**	12.77	10.75	2.02	13.59	11.52	2.07
Diabetes	7.25	6.27	1	8.95	8.03	0.92	9.68	7.89	**1.79**
average	10.77	8. 30	**2.48**	12	10.53	1.46	11.71	10.37	1.35

In Table II, under 80% unlabeled rate, the average error rate of corresponding improvements for CEGSL algorithm is 5.33% when BP neural network is used. It is better than Tri-training (2.0%) and Co-training (3.45%). Similarly, when ID3 decision tree is used as classifier, CEGSL algorithm not only has higher error rate of corresponding improvement for German, Breast-cancer, and Diabetes than Tri-training and Co-training, but also the final error rate (12.48%) is better than Tri-training (14.34%) and Co-training (13.62%).

In Table III, under 60% unlabeled rate, the average error rate of corresponding improvements for CEGSL algorithm is 4.32% when BP neural network is used. The improvement of average error is higher than Tri-training (2.15%) and Co-training (2.13%).

In Table IV, under 40% unlabeled rate, the classifiers get enough labeled data for learning. Both of classifiers become more stronger. Therefore, the initial error rates for three algorithms are decreased. This causes the improvement of classification precision becomes smaller. CEGSL, Tri-training and Co-training only get

improvement of 3.35%, 1.96%, 2.4% respectively. Under these circumstances, CEGSL still expresses better performance.

In Table V, under 20% unlabeled rate, the original labeled data can train a strong classifier and the performance of unlabeled data is decreased. The improvements of CEGSL, Tri-training, and Co-training only reach 2.48%， 1.46%， 1.35% respectively. The CEGSL shows greater achievement.

V. CONCLUSIONS

In this paper, the CEGSL algorithm is proposed. This algorithm combines graph-based semi-supervised learning and collaboration-training algorithms. It makes use of structure information of sample data to calculate the classification probability of unlabeled example explicitly. This algorithm is facilitated with good efficiency and generalization ability because it can effectively select sample data to label and use multiple classifiers to help to perform the final hypothesis. Experiments on UCI datasets prove the efficiency of this algorithm. CEGSL is worth studying in determination of classification error rate in future work. Its applicability is wide because it does not requires sufficient and redundant views. Moreover, using statistical techniques to further identify and deal with noise data can be researched in the future.

ACKNOWLEDGMENT

This work was sponsored by the Visual Computing and Visual Reality Key Laboratory of Sichuan Province in China (No. PJ201102).

REFERENCES

[1] I. Cohen, F. G. Cozman, N. Sebe, M. C. Cirelo, T. S. Huang. Semisupervised learning of classifiers: Theory, algorithm, and their application to human-computer interaction[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2004, 26(12): 1553-1567.

[2] Z.-H. Zhou. Learning with unlabeled data and its application to image retrieval. In Proceedings of the 9th Pacific Rim International Conference on Artificial Intelligence, pages 5–10(2006).

[3] Jimin Li, A Novel Semi-supervised SVM Based on Tri-training for Intrusion Detection. JOURNAL OF COMPUTERS, VOL. 5, NO. 4, APRIL (2010).

[4] Kurt Driessens1, Peter Reutemann2, Using Weighted Nearest Neighbor to Benefit from Unlabeled Data. Encyclopedia of Machine Learning 2010: 857-862(2010).

[5] A. Blum, T. Mitchell. Combining labeled and unlabeled example with co-training[C]. In: Proceedings of the 11th Annual Conference on Computational Learning Theory (COLT'98), Wisconsin, MI, 1998, pp:92-100.

[6] Zhu X. Semi-supervised learning literature survey [R]. Technical Report 1530, Department of Computer Sciences, University of Wisconsin at Madison, Madison, WI, Jul.2008.

[7] D. Zhou, O. Bousquet, T. Lal etc. Learning with local and global consistency [C]. Advances in Neural Information Processing Systems(NIPS), Cambridge, MA:MIT Press, 2004, 16, pp:321- 328.

[8] S. Goldman, Y. Zhou. Enhancing supervised learning with unlabeled example[C]. In: Proceedings of the 17th International Conference on Machine Learning (ICML'00), San Francisco, CA, 2000, pp:327-334.

[9] X.J. Zhu. Z. Ghahramani. Semi-supervised learning using Gaussian fields and Harmonic functions[C]. In: Proceedings of International Conference of Machine Learning(ICML'03), Washington DC, 2003, pp:912-919.

[10] Zhou Z H, Li M. Tri-training: Exploiting unlabeled example using three classifiers [J].IEEE Transactions on Knowledge and Data Engineering， 2005， 17(11), pp:1529-1541.

[11] Tao Guo, Guiyang Li, Improved Tri-Training with Unlabeled example [C]. International Conference on Nanotechnology and Computer Engineering (CNCE), 2011.

[12] Yongzhao Zhan， Yabi Cheng. Co-Training Semi-Supervised Active Learning Algorithm with Noise Filter[J]. Pattern Recognition and Artificial Intelligence, 2009,22(5):750-755

[13] Zhu X J. Semi-supervised learning with graphs [D].USA: Carnegie Mellon University. 2006.1-89.

[14] Blum A, Chawla S. Learning from labeled and unlabeled example using graph mincuts [C]. in: Proceeding of 18th International Conference on Machine Learning. 2001.

[15] Pavan K, Rong J. SemiBoost: Boost for Semi-supervised Learning [J].IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009, 31(11):2000-2014.

[16] Zhou Z H, Wang Jue, Machine learning and application [M], Beijing, Tsinghua University Press, 2007, PP:259-275

Tao Guo received the M.S. degree from computer science and computer engineering at University of Arkansas of USA in 2001. She is an associate professor in the College of Computer Science, Sichuan Normal University, China. Her current areas of interest include data mining and bioinformatics. Email:tguo35@gmail.com

Guiyang Li received the Ph.D. degree from computer science of Sichuan University of China in 2009. He is an associate professor in the College of Computer Science, Sichuan Normal University, China. He is actively involved in the development of network security. His current areas of interest include artificial immune computation, network security.Email:guiyang.li@gmail.com