IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 7, 2020, accepted January 21, 2020, date of publication February 3, 2020, date of current version February 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2971229

Metric-Based Semi-Supervised Regression

CHIEN-LIANG LIU, (Member, IEEE), AND QING-HONG CHEN
Department of Industrial Engineering and Management, National Chiao Tung University, Hsinchu 30010, Taiwan

Corresponding author: Chien-Liang Liu (clliu@mail.nctu.edu.tw)

This work was supported in part by the Ministry of Science and Technology, Taiwan, under Grant MOST 107-2221-E-009-109-MY2 and Grant MOST 107-2218-E-009-005.

ABSTRACT Regression problems are present in many industrial applications, and many supervised learning algorithms have been devised over decades. However, available labeled examples are limited in some application settings; meanwhile, enormous unlabeled examples are relatively easy to collect. Thus, this work proposes a simple but effective method to cope with semi-supervised regression problems. We propose to use deep neural networks to develop our proposed method as deep learning has shown promising results in recent years. Our proposed method is a metric-based approach, and the goal is to learn an embedding space by metric learning with few labeled examples and enormous unlabeled examples. The regression estimation of the target data point is performed on the new space. We generate an artificial dataset based on several criteria to investigate whether the proposed model could make accurate predictions on the data samples that have specific properties. The experimental results point that our proposed model could capture the trend of a non-linear function and normally predict well even though this dataset comprises extreme outliers. Moreover, we conduct experiments on four datasets and compare our proposed work with several alternatives. The experimental results indicate that our proposed method achieves promising results. Besides performance evaluation, detailed analysis about our proposed method is also provided in this work.

INDEX TERMS Semi-supervised regression, metric learning, siamese network, embedding space.

I. INTRODUCTION

Regression problems are present in many industrial applications, including the prediction of product quality [22], aesthetic quality assessments [4], condition monitoring [9], and analysis of wafer probe test data [31]. Moreover, sampling inspection is a popular way in the industry, in which the operators tend to estimate the overall product qualities with limited inspected products based on statistical inference. In this case, the inspected products comprise outputs, but the outputs for most products are missing.

In machine learning, supervised learning has been widely applied to many application domains, but acquiring sufficient training data is still a time-consuming and expensive task. Nevertheless, data labeling is always performed manually, and requires the involve of domain experts in some domains such as medical diagnosis. In contrast, semi-supervised learning [5] is another learning method that is in-between supervised learning and unsupervised learning. In many application settings, a few labeled examples are available, and enormous unlabeled examples are relatively easy to collect. In the aforementioned sampling inspection, the products that have inspection information belong to labeled data, and the number of quantities is limited owing to cost and time consideration. In this case, semi-supervised learning may help to train a robust and accurate model by using a small amount of labeled data along with enormous unlabeled data.

Recently, deep learning is one of the most important topics in machine learning as it has shown promising results in many application domains. Traditional machine learning algorithms normally rely on hand-crafted features, making it difficult to achieve the goal of end-to-end learning. Moreover, the extracted features are not easy to generalize to other data samples. In contrast, the key idea behind deep learning is to consider feature extraction as a learning problem, and use deep neural networks to learn hierarchical and discriminative feature representations from data. The deep neural networks always involve enormous model parameters, explaining why deep learning normally requires enormous labeled data to train the models. Therefore, using limited labeled data to train deep learning models has become an important and active research topic in recent years [8], [32], [34].

The associate editor coordinating the review of this manuscript and approving it for publication was Hualong Yu.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see http://creativecommons.org/licenses/by/4.0/

In machine learning, metric learning [36] is to learn a similarity function from data, so that the function could be used to measure how similar two data points are. The success of deep learning inspires researchers to apply deep learning technique to learn an embedding space [38], and the similarity measurement on the new space could preserve the similarity relationship between two data points. It is apparent that the learning for the embedding space is crucial to the subsequent tasks. However, these approaches were found mostly in computer vision. Moreover, none of these approaches focused on regression problem.

This work proposes a metric-based semi-supervised regression algorithm (MSSR) based on metric learning. Deep learning could learn good feature representations from data, explaining why the proposed model combines metric learning and deep neural networks to devise our proposed method. Learning an embedding space is the key step, and we use siamese network [18] as the base model in our proposed algorithm. Notably, the original siamese network is designed for image verification and categorization tasks, and we devise several mechanisms to enable the proposed method to use siamese network to deal with semi-supervised regression problems. First, we propose a hybrid approach to estimate the similarity of data points. Second, we propose to use random sampling to deal with the imbalanced problems presented in the data. Finally, we propose to use kNN to make predictions for the data points projected to the embedding space that is learned by the siamese network. The reason to use kNN is that it is a non-parametric method, and it only relies on nearest neighbors to make predictions, making it possible to reduce the influence brought by extreme outliers. We generate an artificial dataset based on several criteria to evaluate our proposed method, and the experimental results show that our proposed model could predict well even though the artificial dataset comprises extreme outliers. Moreover, most existing semi-supervised learning methods focus on classification problems, and the methods on semi-supervised regression are relatively limited. Thus, this work focuses on regression problems that are commonly occurred in many industrial applications to develop a semi-supervised learning method.

The contributions of this paper are listed as follows. First, we propose a semi-supervised learning model to cope with regression problems. To the best of our knowledge, this is the first work attempting to use deep learning to learn an embedding space to deal with regression problem. Second, we compare our proposed algorithm with several alternatives on four datasets, and the experimental results indicate that our proposed work outperforms state-of-the-art methods. Finally, detailed analysis about our proposed algorithm is also provided in this work.

The rest of this paper is organized as follows. Section II presents related surveys and techniques. Section III briefly introduces siamese network as the proposed work relies on siamese network to learn an embedding space. Section IV presents the proposed algorithm. Section V and VI show the experimental results and detailed discussions. The conclusions and future work are presented in Section VII.

II. RELATED WORK

Acquiring sufficient labeled data is difficult in many application domains, inspiring many researchers to devise semi-supervised learning algorithms to attack this problem [2], [12], [26], [41]. Notably, semi-supervised learning could not be applied to all problems as it is based on two assumptions [5]. First, points that are close to each other in a manifold are more likely to share the same label. The second one is cluster assumption; data tend to form individual clusters based on their characteristics, meaning that the data points in the same cluster are more likely to share the same label. Based on the two assumptions, semi-supervised learning has been successfully applied to many application domains.

Among these methods, generative methods are based on one or more generative models that are in charge of the generation of all data points. Therefore, one can connect the unlabeled data points with the learning targets through the parameters of the underlying models, and the label information of the unlabeled data can be regarded as missing parameters of the model. The generative methods assume that the models depend on unobserved latent variables, so the use of expectation-maximization (EM) algorithm [7] is a typical approach to iteratively find maximum likelihood estimates of parameters in the models. Different generative models are available depending on the characteristics of the data points, such as mixture of Gaussian [30], mixture of experts [6], and Naive Bayes [27].

The co-training [35], [42] assumes that features could be split into two sets, and the two sets are conditional independent given the class. Given the two sets, co-training trains two classifiers with the two sets. Each classifier could make predictions on unlabeled data, and teaches the other classifier with the unlabeled examples that it could predict with high confidence. Repeat the process of learning from each other until convergence, one could obtain a robust predictive model.

Semi-Supervised Support Vector Machine (S3VM) [1] is a generalization of support vector machines in the manner of semi-supervised learning. The most famous of the semi-supervised support vector machines is TSVM (Transductive Support Vector Machine) [16], which attempts to consider various possible label assignments for unlabeled samples. Besides S3VM and TSVM, Melacci and Belkin [21] proposed a semi-supervised method called Laplacian SVM (LapSVM), which explores the manifold structure of data through the Laplacian matrix of the graph. In addition to the original SVM hinge loss and ℓ_2 norm of the model parameters, Laplacian Eigenmaps was considered as the smoothness regularization term. Xu *et al.* [37] extended least-squares support vector regression (LS-SVR) to propose a semi-supervised LS-SVR that only solves a convex linear system in the training phrase, but additional estimation of the label for each sample in the training set is required.

The graph based method maps the data to a graph to reveal the structure within the data points, wherein the node corresponds to the data, and the edge between nodes corresponds to the similarity between the data. One of the assumptions behind semi-supervised learning is that the internal structure of the labeled data and unlabeled data revealed by the classification function should be smooth. Consequently, many methods viewed the smoothing property as constraints of the original optimization problems [16], [17]. Notably, similar data points are expected to share similar labels based on smoothness assumption, and the goal is to find an assignment of labels for unlabeled examples, so that the overall error of label assignments on the graph is minimized. Miyato *et al.* [24] devised a new regularization method based on virtual adversarial loss, which measures local smoothness of the conditional label distribution, and proposed a novel training method called virtual adversarial training that could deal with semi-supervised learning problems. Label propagation is a semi-supervised learning approach that propagates labels of the labeled data points to the unlabeled data points [43], but it may be influenced by the outliers to mislead the propagation. Gong *et al.* [11] devised a novel propagation scheme via teaching-to-learn and learning-to-teach to explicitly manipulate the propagation sequence.

Semi-supervised learning concerns how to improve performance via the usage of unlabeled data, and many methods have been developed to alleviate such a fundamental challenge for semi-supervised classification. The importance of regression problems has inspired more and more researchers to devoted to devising semi-supervised regression algorithms. Zhao *et al.* [39] proposed a semi-supervised sparse Bayesian regression model to cope with the training dataset with partial missing outputs. They treated missing outputs as random variables and performed variational inference to seek the optimal approximate posteriors over the uncertain variables. However, the time complexity of their proposed method is $O((N+1)^3)$ as it involves three inverse computations in the algorithm, making it difficult to cope with large dataset. Li *et al.* [20] presented an algorithm called SAFER (SAFE semi-supervised Regression), which attempted to maximize the performance gain based on the assumption that the weights of semi-supervised regression learners are from a convex set. This could be formulated as a saddle-point convex-concave optimization, and they considered safe semi-supervised regression as a geometric projection problem to devise the algorithm. Experimental results indicated that SAFER improved safeness of semi-supervised regression. Jean *et al.* [15] proposed a semi-supervised deep kernel learning method, which is a semi-supervised regression model based on minimizing predictive variance in the posterior regularization framework. The proposed method combines hierarchical representation learning of neural networks with probabilistic modeling capabilities of Gaussian processes.

In recent years, deep learning has become a popular research topic in machine learning. Therefore, many researchers have proposed to combine the deep learning architecture and the assumption of semi-supervised learning to devise semi-supervised classification algorithms. Rasmus *et al.* [28] proposed a ladder network architecture, which comprises encoder and decoder networks. Encoder network input comprises two paths; one is the clean input data, while the other one is to add Gaussian noise to each layer of the input data. The goal is to enable the decoder to de-noise, giving a base to increase the difficulty of the decoder, and prevent from learning a trivial network. Iscen *et al.* [14] combined deep neural networks and label propagation to devise an iterative semi-supervised classification method, in which they used a deep neural network to learn feature representations and used the graph-based approach for transductive learning [40] to generate pseudo-labels for unlabeled data.

Laine and Aila [19] proposed two model architectures, namely $\prod$-model and Temporal ensembling. Notably, $\prod$-model is similar to the Temporal ensembling. The experimental results showed that both models could greatly improve performance, while Temporal ensembling is slightly better than $\prod$-Model. Both models are based on ensembling learning. Temporal ensembling uses exponential moving average (EMA) techniques to combine the predictions of each epoch. Tarvainen and Valpola believed that this will result in poor processing speed when dealing with large datasets. Therefore, they proposed the Mean Teacher model [33], which averages the weight of the model instead of predicting the result.

III. PRELIMINARY

The proposed method is a metric-based approach, and we use siamese network to perform metric learning to learn an embedding space from data. Thus, this section briefly introduces siamese network. Siamese network was first introduced to solve signature verification problem [3], in which the network requires two images as the inputs.

In computer vision, deep learning has shown that it could learn hierarchical and discriminative feature representations from data, giving a base for many researchers to apply deep neural networks to siamese network. For example, koch *et al.* [18] explored 20-way and one-shot classification task using the Omniglot dataset, in which their proposed model used convolutional neural network to learn feature representation. As mentioned above, siamese network requires a pair of data points to be the inputs, so each training instance could be represented as a tuple (x_i, x_j, c), where x_i and x_j are the inputs of the model, while c is the label of the pair. If x_i and x_j belong to the same character class, then the label $c = 0$; otherwise, the label $c = 1$. The aforementioned problem is a binary classification, so they imposed a regularized cross-entropy objective function as the loss function.

Schroff *et al.* [29] presented a model called FaceNet to learn an embedding space from face images, so that each face could be mapped to a compact Euclidean space where face similarity could be estimated. The FaceNet used a triplet network to train the model, which comprises three inputs,

including anchor, positive, and negative data points. The anchor and positive belong to the same class, while anchor and negative are in different classes. The goal is to minimize the distance between the anchor and the positive, and maximize the distance between the anchor and the negative.

Both siamese network and triplet network require the similarity between inputs to train the model. In a typical setting for classification problems, the similarity of the data points could be obtained from label information based on whether two data points are in the same class. However, this setting is not so straightforward in regression as the labels are real values, and it is difficult to determine whether two real values are similar except their values are identical.

Moreover, unlabeled data are relatively easy to collect, but the aforementioned methods could not benefit from unlabeled data in the course of model training. For the regression problems that satisfy the cluster and manifold assumptions of semi-supervised learning, they normally could benefit from enormous unlabeled data. Therefore, this work focuses on developing a metric-based method for semi-supervised regression problems.

IV. PROPOSED METHOD

This section introduces the notations, the generation of similar and dissimilar pairs, and training as well as prediction of the proposed method.

A. NOTATION

This section introduces the notations that will be used in this work. This work focuses on semi-supervised regression, so we assume that a few labeled examples and enormous unlabeled examples are available at hand, and they are the elements of the training set. Consequently, the training set comprises labeled examples $X_L = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, where $x_i \in \mathbb{R}^d$ is the input data and $y_i \in \mathbb{R}$ is the corresponding label, and unlabeled examples $X_U = \{x_{n+1}, \ldots, x_m\}$. Besides, the test set $X_T = \{x_1, \ldots, x_t\}$, only comprising feature vectors, is used for model evaluation. The training set and test set are illustrated in Fig. 1.

In the training phase, we use X_L and X_U to generate pairwise pairs, since the training of siamese network requires two inputs. Once the model training is completed, one could obtain a mapping function that could project each input data point x to a new embedding space.

B. PROBLEM SPECIFICATION

Given the training set, the goal is to learn a regressor h, so that the prediction of an input data $x \in X_L$ could be accurate, namely, $h(x) \approx y$. Additionally, for any two points, x_i and x_j, in the training set, if x_i and x_j are similar, the two points in the embedding space should be similar, namely, $h(x_i) \approx h(x_j)$. In the setting of supervised learning, all the data points in the training set comprise labels, and the number of training examples is sufficient to train a regressor, explaining why many state-of-the-art regression methods have been devised in the last decades. However, in semi-supervised learning, the number of available labeled examples in the training set is insufficient to train a robust regressor, so it is important to consider unlabeled data X_U in the training process.

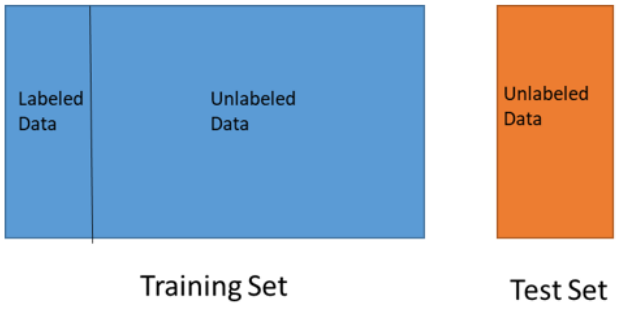


FIGURE 1. Data samples in the training set and test set.

Compared with most previous works in regression problems, we consider to learn a discriminative feature space, and then construct a regressor in the new space. As a result, the embedding space should preserve similarity relationship of data points, making it possible to learn an accurate regressor in the new space. Notably, the training set in this work comprises labeled and unlabeled examples, so the training process is different from the aforementioned metric-based methods.

C. PROCESSING OF SIMILAR AND DISSIMILAR PAIRS

This work uses the siamese network to learn an embedding space, but different from the previous works, this work focuses on regression problem and the training involves unlabeled data. We propose a hybrid approach to estimate the similarity of data points as it uses different schemes for labeled and unlabeled data when estimating the similarity. The proposed hybrid approach comprises two parts. The first part considers the relationship between labeled data and labeled data. In regression problems, the labels are real values, so it is natural to use the absolute value of difference as the base of the similarity measurement. However, it is not easy to determine a threshold value for the measurement of similarity. Thus, we select the top 5% as the similar pairs. It is apparent that other percentages could be used, so we conduct experiments to explore the performances when using different percentages of pairs as the similar pairs.

The second part considers the relationship between the remaining combinations, namely, the combination of labeled data as well as unlabeled data, and the combination of unlabeled data as well as unlabeled data. The similarities for the combinations belonging to the second part are estimated based on Euclidean distance. Similarly, we use the top 5% of the pairs as another source of the training data. Notably, Euclidean distance could be replaced by any distance function that measures the distance of two data points.

Because we only have a small amount of labeled data, the number of pairs generated in the first part is relatively small compared to the second part. Moreover, the pairs from

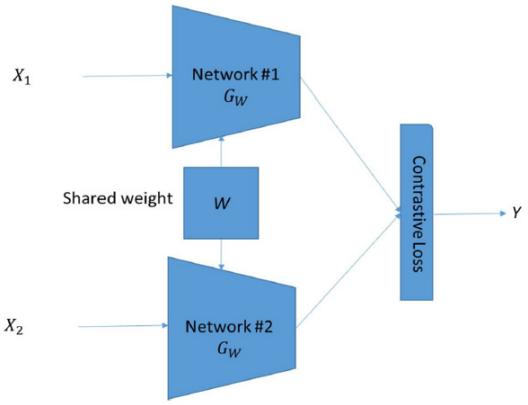


FIGURE 2. Siamese network architecture.

the first part is expected to be more trustable than those of the second part as the similarity in the first part is based on label information. Therefore, we propose to use over-sampling method to increase the number of pairs in the first part until the number is equal to the number of pairs in the second part. Then we merge the pairs in the two parts. Once the merging step is completed, it is expected that the number of dissimilar pairs is much larger than that of the similar pairs. Thus, we use under-sampling technique to reduce dissimilar pairs until the sizes for similar and dissimilar pairs are balanced.

D. MODEL TRAINING

Once the similarity estimation for the data points in the training set is completed, one could use the pairs as the inputs of siamese network to train the model. The siamese neural network is designed as two identical networks that are connected by a distance layer through the final layer, which is trained to predict whether the two inputs are similar or not. The architecture of the siamese network is illustrated in Fig. 2.

Notably, the architectures for the two sub-networks are identical, and the weights as well as biases are shared by the two sub-networks. The main idea behind siamese network is to learn an embedding space via the relationship between two inputs. This work uses contrastive loss [13] as shown in Equation (1) as the loss function.

$$(1-Y)\frac{1}{2}(D_w)^2 + Y\frac{1}{2}\max(0, \alpha - D_w)^2, \tag{1}$$

where D_w is the euclidean distance between the outputs of the siamese networks as defined in Equation (2), $Y \in \{0, 1\}$ is the ground truth for the indication of similar or not, and α is a margin which defines dissimilar pairs that are beyond this margin will not contribute to the loss. In Equation (2), G_w is the neural network within the siamese network. The goal of the contrastive loss is to keep dissimilar pairs away from at least the distance of margin size, and make similar pairs to be as close as possible.

$$D_w = \sqrt{\{G_w(X_1) - G_w(X_2)\}^2} \tag{2}$$

The training for the proposed MSSR algorithm is listed in Algorithm 1. Inputs comprise labeled dataset X_L and unlabeled dataset X_U, and output is a model that could calculate the dissimilarity of two data points. The first step is to generate pairs P_L from X_L based on the absolute difference of the labels, and pairs $P_U = S_U \cup D_U$, where S_U and D_U are similar and dissimilar pairs based on Euclidean distance, from X_L and X_U as listed in Line 1-2 of Algorithm 1. As mentioned above, the pairs in P_L are obtained based on labels, so they are more informative than those in P_U. Therefore, we perform over-sampling on P_L to balance the sizes of P_L and P_U as shown in Line 3.

Next, we split the new P_L into similar pairs S_L and dissimilar pairs D_L. Subsequently, we merge the similar pairs S_L and S_U to obtain a new similar set called $S_{L\cup U}$. The two steps are listed in Line 4-5. Next, we merge the dissimilar pairs D_L and D_U to obtain a new dissimilar set called $D_{L\cup U}$. It is apparent that the size of $D_{L\cup U}$ is much bigger than $S_{L\cup U}$ as the number of dissimilar pairs is expected to be much more than that of similar pairs. Thus, we perform under-sampling on $D_{L\cup U}$ to balance the sizes of $D_{L\cup U}$ and $S_{L\cup U}$ as shown in Line 6. Finally, we use the obtained similar and dissimilar pairs to train the siamese network as presented in Line 7.

Algorithm 1 Training Algorithm

Input: Labeled dataset
$X_L = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, where
$x_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$, and unlabeled dataset
$X_U = \{x_{n+1}, \ldots, x_m\}$.

Output: A neural network that could output dissimilarity of two inputs.

1 Generate pairs P_L from labeled set X_L based on the absolute difference of labels. In other words, similarity of $x_i \in X_L$ and $x_j \in X_L \wedge j \neq i$ is defined as $|y_i - y_j|$. We keep the top 5% of the pairs as the similar pairs, and the remaining as dissimilar pairs.
2 Generate pairs P_U from labeled set X_L and unlabeled set X_U based on Euclidean distance. We keep the top 5% of the pairs as the similar pairs S_U, and the remaining as dissimilar pairs called D_U.
3 Perform over-sampling on P_L until $|P_L| = |P_U|$.
4 Split the new P_L into similar pairs called S_L and dissimilar pairs called D_L.
5 Merge S_L and S_U and call the new set $S_{L\cup U}$.
6 Merge D_L and D_U, and perform under-sampling until the size is the same as $S_{L\cup U}$. The new set for dissimilar pairs is called $D_{L\cup U}$.
7 Use $S_{L\cup U}$ and $D_{L\cup U}$ to train the Siamese network, in which the labels for the pairs in $S_{L\cup U}$ are 0, while the labels for the pairs in $D_{L\cup U}$ are 1.

E. MODEL PREDICTION

Once the learning for embedding space is completed, one could project the data points to the new space, and perform verification or categorization on the new space. As compared

with previous works, this work focuses on regression, and we argue that regression could also benefit from the embedding space, since two data points that share similar features are expected to have similar regression outcomes.

The siamese network could output a dissimilarity value for the input pair. Based on the output of siamese network, we propose to use kNN to estimate the regression values of the target data as kNN is a non-parametric method that relies on training data to make predictions. The parametric models that use mean squared error as the loss function tend to give more weight to outliers, whereas kNN could alleviate the influence brought by the outliers when the hyper-parameter k is appropriate.

The prediction for the proposed MSSR algorithm is listed in Algorithm 2. The inputs involve the siamese network that is the model trained with Algorithm 1, the labeled dataset X_L, the target data point x, and a hyper-parameter k. The first step is to obtain n dissimilarity values between all the labeled examples and the target data point x. Then, we convert dissimilarity values to similarity values. Additionally, we normalize the similarity to the range of [0, 1]. In the final step, one could use average or weighted average to determine the regression value of the target sample.

Algorithm 2 Prediction Algorithm

Input: The trained Siamese network h, labeled dataset $X_L = \{(x_1, y_1), \ldots, (x_n, y_n)\}$, where $x_i \in \mathbb{R}^d$ and $y_i \in \mathbb{R}$, a target data point x, and a hyper-parameter k.

Output: Estimated regression outcome

```
Initialize array d
for i ← 1 to n do
    d[i] ← h(x_i, x)
end
Transform the dissimilarity in d into similarity, d[i] ← 1/d[i]
Use kNN technique to estimate the regression of x based on X_L, d[i], and k
```

V. EXPERIMENTS

This work conducts experiments on four datasets, and compares the proposed algorithm with several alternatives. The introductions for datasets, evaluation metric, and experimental results are presented in the following sections.

A. DATASET

The introductions for the datasets are listed below. All of them are free and public datasets.

- Blood Brain
  This dataset is available in the caret package of R programming language, and it was originally used by Mente and Lombardo [23] to develop models to predict the log of the ratio of the concentration of a compound in the brain and the concentration in blood, in which 134 descriptors were calculated. Notably, 208 non-proprietary literature compounds are included in this package.
- Airfoil Self-Noise
  This is a NASA dataset that can be obtained from UCI machine learning repository, which was obtained from a series of aerodynamic and acoustic tests of two and three-dimensional airfoil blade sections conducted in an anechoic wind tunnel.
- Physicochemical Properties of Protein Tertiary Structure
  The source of the data is obtained from UCI machine learning repository. The dataset is obtained from CASP 5-9, in which 45730 decoys are involved and size varying from 0 to 21 armstrong. We randomly select 1103 samples in the experiments.
- Superconductivty
  The Superconductivty dataset could be downloaded from UCI machine learning repository, and the goal is to predict the critical temperature. This dataset consists of 21263 samples, each of which comprises 81 features. We randomly select 1103 samples in the experiments.

More detailed information for the datasets is listed in Table 1, including the number of features, the number of data samples, and the settings for training set and test set.

B. EVALUATION METRIC

This work focuses on the regression problem, so it is natural to use root mean squared error (RMSE) and mean absolute error (MAE) as the evaluation metrics. The definition for RMSE and MAE are listed in Equation (3) and Equation (4), respectively, where n is the number of data samples in the test set, y_i is the label, and $\hat{y}_i$ is the predicted value.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{n}} \tag{3}$$

$$\text{MAE} = \frac{\sum_{i=1}^{n}|y_i - \hat{y}_i|}{n} \tag{4}$$

C. ARTIFICIAL DATASET

Before conducting experiments on the four real datasets, we generate an artificial dataset based on several criteria to evaluate our proposed method. We follow the setting presented in Figure 1 to develop the dataset, in which the training set comprises 20 labeled data samples and 168 unlabeled data samples, and test set comprises 20 data samples.

We considers three criteria to generate the artificial dataset, in which each data point comprises three predictors, namely, x_1, x_2 and x_3. First, we randomly sample the values for the predictors from very non-uniform distributions as listed in Equation (5).

$$\begin{aligned} x_1 &\sim \mathcal{N}(\mu = 20, \sigma = 5), \\ x_2 &\sim \text{Beta}(\alpha = 1, \beta = 2), \\ x_3 &\sim B(N = 10, P = 0.5), \end{aligned} \tag{5}$$

TABLE 1. Summary of datasets.

Dataset	Features	Data samples	Training set		Test set
			Labeled data	Unlabeled data	
Blood Brain	134	208	20	168	20
Airfoil Self-Noise	5	1503	20	1380	103
Physicochemical	9	45730	20	980	103
Superconductivty	81	21263	20	980	103

TABLE 2. Summary statistics of y for 208 artificial data samples.

mean	std	min	Q1	Q2	Q3	max
2287.67	1197.40	94.68	1611.38	2187.40	2694.35	10030.23

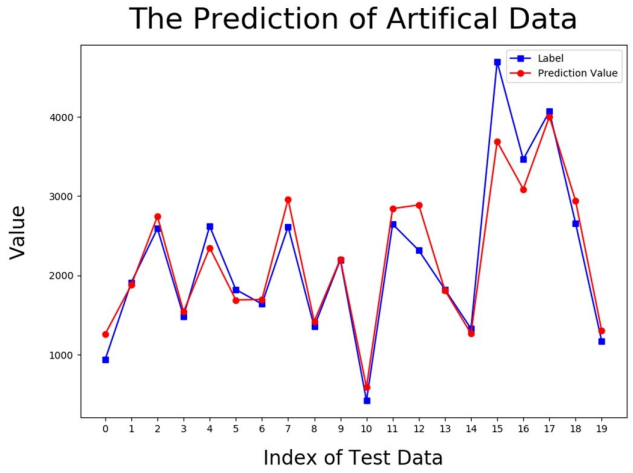


FIGURE 3. Prediction results on an artificial dataset.

where x_1 is drawn from a normal distribution, x_2 is drawn from a beta distribution, and x_3 is drawn from a binomial distribution. Second, we use a quadratic function $y = 5 \times x_1^2 + 15 \times x_2 + x_3 + \mathcal{N}(\mu = 0, \sigma = 1)$ to generate the data. Notably, we add random noises drawing from a normal distribution that has a mean of 0 and a standard deviation of 1 to the data points. Third, we include extreme outliers in the dataset, in which 1% of the data points are defined as the outliers as these data points are generated by adding 7 standard deviations to their original labels, namely, y. Table 2 shows the summary statistics of y for the 208 data samples. It is apparent that the value range of y is huge.

Figure 3 shows the experimental results, in which x-axis represents the index of the test data and y-axis denotes the values. The blue and red points are the labels and prediction values for the 20 data samples in the test set. The experimental results on the artificial dataset point out that the proposed method could capture the trend of the test data points, and normally predict well even though this dataset comprises extreme outliers and noise. The proposed method learns an embedding space that could separate data well, and then performs regression estimation with kNN. The kNN is a non-parametric method, and it only relies on nearest neighbors to make predictions, making it possible to reduce the influence brought by these extreme outliers.

D. COMPARISON METHODS

We use the semi-supervised regression methods provided by the R package called ssr [10][1] as the comparison methods. This package implements the well-known semi-supervised learning approaches, including self-learning and co-training by committee. Notably, the underlying regressors can be the models from the caret (short for Classification And REgression Training) package, or custom functions. For co-training methods, we use two comparison methods, kNN-LM and BayGLM-kNN, in which kNN-LM uses kNN and linear regression as the underlying regressors, while BayGLM-kNN uses Bayesian generalized linear model and kNN. As for self-learning methods, the comparison methods comprise kNN and Bayesian generalized linear model, and these two methods are called self-kNN and self-BayGLM in the experiments. Besides the methods provided by ssr package, two recent methods published in AAAI and NIPS are also included in the experiments as the comparison methods, including SAFER [20] and SSDKL [15].

The SAFER [20] learns a safe prediction from multiple semi-supervised regressors, in which SAFER uses three semi-supervised regressors, one is from the Self-LS method which is semi-supervised extension of the supervised least square method and the other two are from the self-kNN methods which adopt the cosine and the Euclidean distance. SSDKL [15] is a semi-supervised regression model based on minimizing predictive variance in the posterior regularization framework. In their approach, we changed the number of labeled examples and number of test data in order to match the situation of our method.

[1] ssr package: https://github.com/enriquegit/ssr

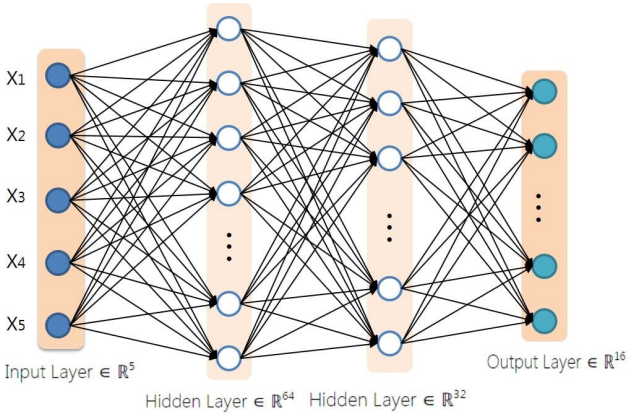


FIGURE 4. Neural network architecture of sub-networks.

E. EXPERIMENTAL SETTINGS

As mentioned above, the two networks in siamese network share the same architecture. We use deep neural networks to construct the sub-networks and each hidden layer uses the rectified linear unit (ReLU) [25] function to perform the non-linear transformation as it is the most commonly used activation function in neural networks. The architecture of the sub-network for the Airfoil Self-Noise dataset is illustrated in Fig. 4, comprising two hidden layers and the number of neurons for the two hidden layers are 64 and 32, respectively. Besides, the dimension for the input layer is the number of features for the input data, while the dimension for the output layer, namely, the embedding space, is 16 or 8 depending on the characteristics of the datasets. Finally, we set $k = 5$ for kNN in the prediction of our proposed algorithm.

F. EXPERIMENTED RESULTS

We conduct experiments on four datasets, and compare the proposed method with several alternatives. The experimental results are presented in Table 3 and Table 4. In the experiments, all comparison methods repeat 5 times, while our proposed method repeats 15 times as the proposed method involves over-sampling and under-sampling, so each time we use five different random seeds to conduct experiments. The mean and standard deviation of the experimental results are presented in Table 3 and Table 4. Besides, we perform t-test on the experimental results as listed in Table 5 and Table 6.

The experimental results indicate that the proposed method achieves the best overall performances on the four datasets. Table 3 shows that the average performances of our proposed method on the four datasets are the best in RMSE, and the t-test results presented in Table 5 point out that the performance differences are significant in many cases. On the other hand, although the proposed method only achieves the best average performance in MAE on one dataset as shown in Table 4, the performance differences between the proposed method and the best ones on the remaining three datasets are insignificant, indicating that the proposed method could achieve almost the same performances as the best ones.

The key idea behind the proposed method is to learn an embedding space with siamese network. It is expected that the data points projecting onto the new space preserve the similarity relationship. Besides, the pairs involve the relationship between labeled and unlabeled data, so the smoothness property among all the data points could be satisfied, giving a base to learn a discriminative space with the setting of semi-supervised learning. This provides a good estimate for the target data point on the embedding space.

The prediction of our proposed model is fast, but the training time of the proposed model is related to the number of pairs generated from the training set. One possible approach to deal with this problem is to use the concept of semi-hard exemplars used in FaceNet [29] to reduce the number of candidates and lead to fast convergence.

VI. DISCUSSION

Besides the experimental results, this work investigates the settings of the proposed method and provides detailed analysis about the proposed method in the following sections.

A. THE IMPACT OF NUMBER OF LABELED DATA ON PERFORMANCE

In semi-supervised learning, the number of labeled data is limited, and it is expected that a model could achieve better performance as more training data are available. This work conducts experiments to investigate the impact of number of labeled data on performance.

We try different settings by using different numbers of training examples to train the embedding of the siamese network. Note that the training data are obtained by random sampling and the number of unlabeled data decrease as more data instances are used as the labeled data. We conduct experiments on three datasets, including Airfoil Self-Noise, Physicochemical Properties of Protein Tertiary Structure, Superconductivty.

Besides, the proposed method uses kNN to estimate the regression of the target data point, so we consider to use different settings of the hyper-parameter k in the experiments. The first setting is the same as the one we used in the experiments, namely, $k = 5$, while the second setting is to use all training examples.The experimental results are presented in Fig. 5.

The experimental results indicate that the model could normally improve performance as more training examples are used for model training. On the other hand, the proposed model could normally yield better performance when $k = 5$. We conjecture that the regression estimation relying on the labeled examples that are informative is a better choice in the setting of semi-supervised learning.

B. OVER-SAMPLING VS. UNDER-SAMPLING

Our proposed method uses two approaches to estimate the similarity between data points, and collect similar pairs. The first approach relies on the relationship between labeled samples, whereas the second approach focuses

TABLE 3. Experimental results (RMSE).

Dataset	kNN-LM	BayGLM-kNN	self-kNN	self-BayGLM	SAFER	SSDKL	Proposed Method
Blood Brain	1.44 ± 0.17	0.97 ± 0.04	1.02 ± 0.13	1.10 ± 0.06	1.04 ± 0.12	0.98 ± 0.12	$\mathbf{0.95 \pm 0.14}$
Airfoil Self-Noise	5.36 ± 0.40	5.25 ± 0.15	6.62 ± 0.60	5.50 ± 0.20	6.24 ± 0.66	6.11 ± 1.06	$\mathbf{5.2 \pm 0.67}$
Physicochemical	5.90 ± 0.18	5.82 ± 0.28	6.65 ± 0.68	7.61 ± 0.54	6.33 ± 0.76	6.2 ± 0.46	$\mathbf{5.78 \pm 0.25}$
Superconductivty	68.1 ± 24.31	25.84 ± 1.23	28.65 ± 2.37	30.2 ± 2.27	24.63 ± 1.43	32.63 ± 1.88	$\mathbf{24.21 \pm 2.35}$

TABLE 4. Experimental results (MAE).

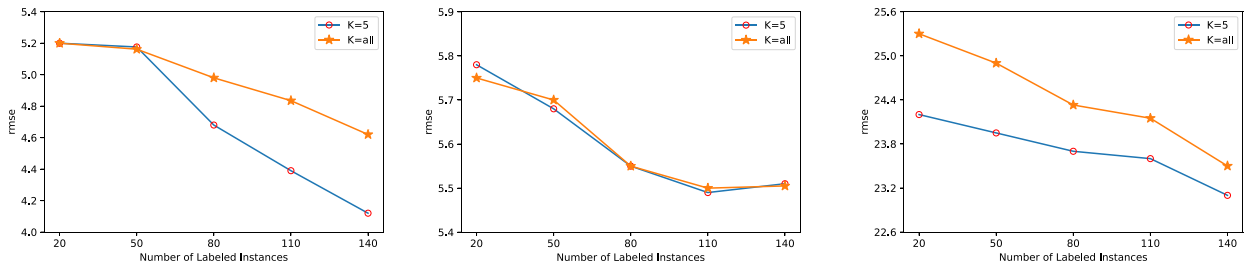
Dataset	kNN-LM	BayGLM-kNN	self-kNN	self-BayGLM	SAFER	SSDKL	Proposed Method
Blood Brain	1.12 ± 0.18	0.70 ± 0.04	0.74 ± 0.09	0.82 ± 0.04	0.79 ± 0.05	0.67 ± 0.08	$\mathbf{0.60 \pm 0.08}$
Airfoil Self-Noise	4.27 ± 0.27	$\mathbf{4.20 \pm 0.16}$	5.30 ± 0.41	4.35 ± 0.18	4.83 ± 0.66	4.62 ± 0.53	4.61 ± 0.61
Physicochemical	$\mathbf{4.88 \pm 0.24}$	4.97 ± 0.40	5.77 ± 0.55	6.03 ± 1.03	5.19 ± 0.22	5.17 ± 0.21	5.06 ± 0.32
Superconductivty	46.27 ± 12.79	19.07 ± 1.06	$\mathbf{18.18 \pm 1.51}$	23.30 ± 2.15	25.17 ± 3.04	28.54 ± 3.81	19.16 ± 1.18

TABLE 5. The p-values of T-test on RMSE metric.

Dataset	kNN-LM	BayGLM-kNN	self-kNN	self-BayGLM	SAFER	SSDKL
Blood Brain	**0.0006**	0.3327	0.1728	**0.003**	0.1009	0.3276
Airfoil Self-Noise	0.2588	0.4025	**0.001**	0.073	**0.009**	0.064
Physicochemical	0.1364	0.4011	**0.022**	**0.028**	0.09	0.055
Superconductivty	**0.0077**	**0.033**	**0.004**	**0.0007**	0.3213	**0.000013**

TABLE 6. The p-values of T-test on MAE metric.

Dataset	kNN-LM	BayGLM-kNN	self-kNN	self-BayGLM	SAFER	SSDKL
Blood Brain	**0.001**	**0.0012**	**0.01278**	**0.000001**	**0.00002**	0.067
Airfoil Self-Noise	0.947	0.9851	**0.0081**	0.9166	0.2668	0.4864
Physicochemical	0.8849	0.6668	**0.0209**	0.0522	0.1675	0.1992
Superconductivty	**0.0044**	0.5614	0.8824	**0.005**	**0.0005**	**0.00023**



(a) Airfoil Self-Noise dataset (b) Physicochemical dataset (c) Superconductivty dataset

FIGURE 5. Performances with different numbers of training samples.

on the cases that involve unlabeled data points. The estimation of the first approach is based on labels, explaining why we perform over-sampling on the pairs generated by this approach. However, performing under-sampling on the pairs collected by the second approach to cope with the problem of imbalance data is worth investigation. As a result, we conduct experiments to explore the difference.

TABLE 7. Experimental results with different sampling methods (RMSE).

Dataset	Airfoil		Physicochemical		Superconductivty	
# of Labeled Samples	Over-Sampling	Under-sampling	Over-Sampling	Under-sampling	Over-Sampling	Under-sampling
20	5.2	6.04	5.78	6.45	24.21	26.53
50	5.15	4.86	5.67	5.88	23.85	24.05
80	4.66	4.64	5.54	5.53	23.55	23.56
110	4.39	4.33	5.48	5.49	23.41	23.46
140	4.09	3.79	5.52	5.52	22.9	22.56

TABLE 8. Experimental results with different percentages of similar pairs (RMSE).

Dataset	5%	10%	15%	20%
Blood Brain	0.95	1.01	1.04	1.06
Airfoil Self-Noise	5.2	5.86	6.00	6.08
Physicochemical	5.78	6.01	6.19	6.27
Superconductivty	24.21	26.16	28.50	28.82

Table 7 shows the experimental results. The experimental results point out that when the number of labeled samples is less than 100, over-sampling normally outperforms under-sampling. In contrast, when the number of labeled samples is more than 100, under-sampling could achieve better results than over-sampling.

When only few labeled samples are available, over-sampling could emphasize the importance of available labeled samples by increasing the number of labeled samples in a random manner. In contrast, when number of labeled samples is sufficient to learn an embedding space, under-sampling on the pairs collected from the second approach could balance the distribution to cope with the problem of imbalance data, while keeping the original similar pairs to avoid overfitting. Another benefit of under-sampling is that it could dramatically reduce the number of similar pairs, making the model training to be more efficient.

C. DIFFERENT PERCENTAGES OF SIMILAR DATA

In our proposed method, we use a hybrid approach to collect similar pairs from the two groups, namely, the group of similarity between labeled data and labeled data as well as the group of labeled data and unlabeled data. We set the threshold to be 5% in the collection process for the two groups. It is apparent that the threshold is a hyper-parameter, so we conduct experiments to analyze the impact of this hyper-parameter on performances.

We conduct experiments with different percentages of similar pairs and evaluate their performances. Table 8 shows the experimental results, indicating that using the top 5% achieves the best performances on the four datasets. Notably, the performances decrease as more similar pairs are collected as the training data. It is expected that the quality of the training data that are collected from the top 5% similar pairs is better than that of the data collected from the top 20%. Moreover, the proposed method requires to perform over-sampling on the collected similar pairs to balance the distribution between similar pairs and dissimilar pairs, so the quality of the similar pairs should be considered and it also explains why the model could benefit from less but informative pairs.

VII. CONCLUSION

This paper proposes a metric-based semi-supervised regression method, in which the goal is to use a small amount of labeled data and enormous unlabeled data to develop an accurate regression model. Central of the proposed method is to rely on a discriminative embedding space, so that the regression estimation on the new space could be accurate. To enable the proposed method to learn a good embedding space, we propose to use siamese network with the similar and dissimilar pairs collected from labeled data and unlabeled data. The experimental results point out that the proposed method outperforms the other alternatives. Besides the regression experiments, this work provides detailed analysis about our proposed method. One of the future works is to combine metric learning and regression in the same network. We expect that the model could benefit from jointly learning two tasks simultaneously. Besides, the selection of pairs is crucial for fast convergence and can speed up model training, so this is our another future work.

REFERENCES

[1] K. P. Bennett and A. Demiriz, "Semi-supervised support vector machines," in *Proc. Adv. Neural Inf. Process. Syst.*, 1999, pp. 368–374.
[2] D. Berthelot, N. Carlini, I. Goodfellow, N. Papernot, A. Oliver, and C. A. Raffel, "MixMatch: A holistic approach to semi-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, H. Wallach, H. Larochelle, A. Beygelzimer, F. D'Alché-Buc, E. Fox, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2019, pp. 5050–5060.
[3] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a 'siamese' time delay neural network," in *Proc. Adv. Neural Inf. Process. Syst.*, 1994, pp. 737–744.
[4] K. Y. Chan, H.-K. Lam, C. K. F. Yiu, and T. S. Dillon, "A flexible fuzzy regression method for addressing nonlinear uncertainty on aesthetic quality assessments," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 8, pp. 2363–2377, Aug. 2017.

[5] O. Chapelle, B. Scholkopf, and A. Zien, "Semi-supervised learning," *IEEE Trans. Neural Netw.*, vol. 20, no. 3, p. 542, Mar. 2009.
[6] F. G. Cozman, I. Cohen, and M. C. Cirelo, "Semi-supervised learning of mixture models," in *Proc. 20th Int. Conf. Mach. Learn. (ICML)*, 2003, pp. 99–106.
[7] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *J. Roy. Stat. Soc., Ser. B (Methodol.)*, vol. 39, no. 1, pp. 1–22, 1977.
[8] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, Aug. 2017, pp. 1126–1135.
[9] G. S. Galloway, V. M. Catterson, C. Love, A. Robb, and T. Fay, "Modeling and interpretation of tidal turbine vibration through weighted least squares regression," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published.
[10] E. Garcia-Ceja, *SSR: Semi-Supervised Regression Methods*. R package, 2019. [Online]. Available: https://CRAN.R-project.org/package=ssr
[11] C. Gong, D. Tao, W. Liu, L. Liu, and J. Yang, "Label propagation via teaching-to-learn and learning-to-teach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 6, pp. 1452–1465, Jun. 2017.
[12] C. Gong, D. Tao, S. J. Maybank, W. Liu, G. Kang, and J. Yang, "Multi-modal curriculum learning for semi-supervised image classification," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3249–3260, Jul. 2016.
[13] R. Hadsell, S. Chopra, and Y. Lecun, "Dimensionality reduction by learning an invariant mapping," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jul. 2006, pp. 1735–1742.
[14] A. Iscen, G. Tolias, Y. Avrithis, and O. Chum, "Label propagation for deep semi-supervised learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5070–5079.
[15] N. Jean, S. M. Xie, and S. Ermon, "Semi-supervised deep kernel learning: Regression with unlabeled data by minimizing predictive variance," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 5322–5333.
[16] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proc. 16th Int. Conf. Mach. Learn.*, vol. 99, 1999, pp. 200–209.
[17] T. Joachims, "Transductive learning via spectral graph partitioning," in *Proc. 20th Int. Conf. Mach. Learn. (ICML)*, 2003, pp. 1–8.
[18] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *Proc. ICML Deep Learn. Workshop*, vol. 2, 2015, pp. 1–8.
[19] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," 2016, *arXiv:1610.02242*. [Online]. Available: https://arxiv.org/abs/1610.02242
[20] Y.-F. Li, H.-W. Zha, and Z.-H. Zhou, "Learning safe prediction for semi-supervised regression," in *Proc. 21st AAAI Conf. Artif. Intell.*, 2017, pp. 1–7.
[21] S. Melacci and M. Belkin, "Laplacian support vector machines trained in the primal," *J. Mach. Learn. Res.*, vol. 12, pp. 1149–1184, Mar. 2011.
[22] M. Melhem, B. Ananou, M. Ouladsine, and J. Pinaton, "Regression methods for predicting the product's quality in the semiconductor manufacturing process," *IFAC-PapersOnLine*, vol. 49, no. 12, pp. 83–88, 2016.
[23] S. Mente and F. Lombardo, "A recursive-partitioning model for bloo–brain barrier permeation," *J. Comput. Aided Mol. Des.*, vol. 19, no. 7, pp. 465–481, Jul. 2005.
[24] T. Miyato, S.-I. Maeda, M. Koyama, and S. Ishii, "Virtual adversarial training: A regularization method for supervised and semi-supervised learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1979–1993, Aug. 2019.
[25] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 807–814.
[26] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using EM," *Mach. Learn.*, vol. 39, nos. 2–3, pp. 103–134, 2000.
[27] R. Raina, Y. Shen, A. Mccallum, and A. Y. Ng, "Classification with hybrid generative/discriminative models," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 545–552.
[28] A. Rasmus, M. Berglund, M. Honkala, H. Valpola, and T. Raiko, "Semi-supervised learning with ladder networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 3546–3554.
[29] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.
[30] N. Shental, A. Bar-Hillel, T. Hertz, and D. Weinshall, "Computing Gaussian mixture models with em using equivalence constraints," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 465–472.
[31] K. Skinner, D. Montgomery, G. Runger, J. Fowler, D. Mccarville, T. Rhoads, and J. Stanley, "Multivariate statistical methods for modeling and analysis of wafer probe test data," *IEEE Trans. Semicond. Manufact.*, vol. 15, no. 4, pp. 523–530, Nov. 2002.
[32] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4077–4087.
[33] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1195–1204.
[34] E. Triantafillou, R. Zemel, and R. Urtasun, "Few-shot learning through an information retrieval lens," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 2255–2265.
[35] W. Wang and Z.-H. Zhou, "A new analysis of co-training," in *Proc. ICML*, vol. 2, 2010, p. 3.
[36] E. P. Xing, M. I. Jordan, S. J. Russell, and A. Y. Ng, "Distance metric learning with application to clustering with side-information," in *Proc. Adv. Neural Inf. Process. Syst.*, 2003, pp. 521–528.
[37] S. Xu, X. An, X. D. Qiao, L. J. Zhu, and L. Li, "Semi-supervised least-squares support vector regression machines," *J. Inf. Comput. Sci.*, vol. 8, no. 6, pp. 885–892, 2011.
[38] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Deep metric learning for person re-identification," in *Proc. 22nd Int. Conf. Pattern Recognit.*, Aug. 2014, pp. 34–39.
[39] J. Zhao, L. Chen, W. Pedrycz, and W. Wang, "A novel semi-supervised sparse Bayesian regression based on variational inference for industrial datasets with incomplete outputs," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published.
[40] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 321–328.
[41] K. Zhou, X. Gui-Rong, Q. Yang, and Y. Yu, "Learning with positive and unlabeled examples using topic-sensitive PLSA," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 1, pp. 46–58, Jan. 2010.
[42] Z.-H. Zhou and M. Li, "Semi-supervised regression with co-training," in *Proc. IJCAI*, vol. 5, 2005, pp. 908–913.
[43] X. Zhu, "Semi-supervised learning literature survey," Dept. Comput. Sci., Univ. Wisconsin-Madison, Madison, Wisconsin, Tech. Rep. 1530, 2005.

CHIEN-LIANG LIU (Member, IEEE) received the M.S. and Ph.D. degrees from the Department of Computer Science, National Chiao Tung University, Taiwan, in 2000 and 2005, respectively. He is currently an Associate Professor with the Department of Industrial Engineering and Management, National Chiao Tung University. His research interests include machine learning, data mining, deep learning, and big data analytics.

QING-HONG CHEN received the M.S. degree from the Department of Industrial Engineering and Management, National Chiao Tung University, Taiwan, in 2019. His research interests include machine learning and data mining.