Kernel Based Semi-supervised Extreme Learning Machine and the Application in Traffic Congestion Evaluation

Qing Shen, Xiaojuan Ban, Chong Guo and Cong Wang

Abstract Extreme learning machine (ELM) has proven to be an efficient and effective learning paradigm for a wide field. With the method of kernel function instead of the hidden layer, Kernel-ELM overcame the problem of variation caused by randomly assigned weights. In this paper, Kernel based optimization is introduced in semi-supervised extreme learning machine (SSELM) and the improvements of performance are evaluated by the experiment. The result shows that optimized by kernel function, Kernel-SSELM can achieve higher classification accuracy and robustness. In addition, The Kernel-SSELM is used to train the traffic congestion evaluation framework in Urban Transportation Assessment and Forecast System.

Keywords Semi-supervised ELM · Kernel function · Traffic congestion evaluation

Qing Shen, male, 1988, doctoral candidate, main research direction: action analysis, machine learning, E-mail: shenqingcc222333@gmail.com. Xiaojuan Ban, female, 1970, doctor, doctoral student supervisor, main research direction: artificial life, intelligent control, E-mail: banxj@ustb.edu.cn.

Q. Shen (✉) · X. Ban · C. Guo · C. Wang
School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China
e-mail: shenqingcc222333@gmail.com

X. Ban
e-mail: banxj@ustb.edu.cn

C. Guo
e-mail: kotrue2015@gmail.com

C. Wang
e-mail: 1303003723@qq.com

J. Cao et al. (eds.), *Proceedings of ELM-2015 Volume 1*,
Proceedings in Adaptation, Learning and Optimization 6,
DOI 10.1007/978-3-319-28397-5_18

1 Introduction

Primarily, ELM was applied to supervised learning problems in full labeled data. Gao Huang et al. [1] proposed the semi-supervised framework of ELM to extend the capacity to deal with unlabeled data. SSELM greatly extend the application of ELM, for instance, in the field of text classification, information retrieval and fault evaluation as the collection of labeled data is bound to cost a lot of money and time while the unlabeled data is easy to collect and its number is large.

Although ELM improves the training efficiency to a high extent, the random distribution of input layer and the hidden layer parameters cause great variation of classification accuracy under the circumstance of same training data and model parameters which significantly influences the stability of ELM [2]. On the other hand, the number of hidden layer nodes also has a huge impact on the accuracy. In many studies, the number of hidden layer nodes is set to a large number that is usually greater than the number of training samples. However, the experiments show that the more hidden layer nodes is not better. The relationship between optimal accuracy of different datasets and the number of hidden layer nodes is complicated.

The approach replacing ELM hidden layer with kernel function make ELM does not need random hidden layer and input layer because the calculation of hidden input is carried out by kernel function. Kernel-ELM solves the problem resulted from random distribution of input layer and hidden layer parameters in ELM and gain higher relevance to corresponding datasets as well as higher stability [3] with the sacrifice of training speed.

SSELM and ELM have a unified framework. As a result of randomly generated feature mapping, stability problem is existed in the SSELM. This paper introduces the kernel function into the SSELM of Gao Huang et al. [1] and evaluates the improvements in stability and accuracy of SSELM optimized by kernel function.

The rest of the paper is organized as follows. Section 2 reviews the current research progress in the field of semi-supervised learning and kernel function at present in. Section 3 presents the algorithms framework of Kernel-SSELM. The evaluation experiment of efficiency is conducted in Sect. 4. Section 5 elaborates the application of Kernel-SSELM in the Traffic congestion evaluation system based on floating car data. Finally, Sect. 6 draws the conclusion and our future plan.

2 Related Research

Only a few existing research studies ELMs have dealt with the problem of semi-supervised learning. In the earlier days the manifold regularization framework was introduced into the ELMs model to leverage unlabeled data extending ELMs for semi-supervised learning [4, 5]. Li et al. [6] propose a training algorithm that assigns the most reliable predicted value to unlabeled sample in the repeated trainings of ELM for purpose of expanding the labeled sample sets continuously.

The proposed SSELM of Gao Huang et al. [1] takes example by the state-of-the-art semi-supervised learning framework to optimize the cost equation of ELM's processing unlabeled samples. Related to Laplacian support vector machines (LapSVM) and Laplacian regularized least squares (LapRLS), it is involved with the manifold assumption and simplifies the problem into the regularized least square problem.

Ever since the optimization based on kernel function was introduced into the ELM [2], many researchers have made advances in the practical application of theories. The significant solved problems are from two aspects. One aspect aims to choices of specific application's kernel function and optimization [7, 8]. The other aspect aims to the information fusion of ELM [9].

3 Kernel-Based SSELM

Gao Huang et al. [1] introduced manifold assumption into ELM, and proposed the solution of β in SSELM. For a training data set having 1 number of labeled samples and u number of unlabeled samples, the output weights β of a SSELM is:

$$\beta = H^T \left(I + \tilde{C}HH^T + \lambda LHH^T\right)^{-1} \tilde{C}\tilde{Y} \tag{1}$$

The formulate is valid when the number of hidden nodes is more than the number of labeled samples 1. The $\tilde{Y}$ is the training target including the first 1 rows of labeled data equal to Y and the rest equal to 0. λ is user-defined semi-supervised learning rate. $\tilde{C}$ is a $(l + u) \times (l + u)$ diagonal matrix with the first 1 diagonal elements of cost coefficient and the rest equal to 0. $\tilde{C}$ can be calculated as:

$$\mathrm{C}_i = \frac{C_0}{N_{P_i}} \quad i = 1, \ldots, l \tag{2}$$

where C_0 is user-defined cost coefficient, and N_{P_i} represents the sample quantity of the pattern of ith sample. L is Laplacian matrix, which can be calculated as $L = D - W$. $W = \left[w_{i,j}\right]$ is the similarity matrix of all the labeled and unlabeled samples. D is a diagonal matrix with its diagonal elements $D_{ii} = \sum_{j=1}^{n} w_{ij}$.

Huang et al. [2] suggested using a kernel function if the hidden layer feature mapping $\mathrm{h(x)}$ is unknown. The kernel matrix χ for ELM can be written as follows, where $\mathrm{K}(\mathrm{x_i}, \mathrm{y_i})$ is kernel function:

$$\chi_{ELM} = HH^T \quad \chi_{ELM_{i,j}} = h(x_i) \cdot h(y_i) = K(x_i, y_i) \tag{3}$$

Then the output function of Kernel-SSELM can be written as:

$$y = F_{SSELM}(x) = h(x)\beta = \begin{bmatrix} \mathrm{K}(x, x_1) \\ \vdots \\ \mathrm{K}(x, x_n) \end{bmatrix} \left(I + \tilde{C}\chi_{ELM} + \lambda L \chi_{ELM}\right)^{-1} \tilde{C}\tilde{Y} \tag{4}$$

4 Experiment Result

4.1 Experimental Setup

We evaluated the performance of Kernel-SSELM on various semi-supervised tasks. All experiments were implemented using Matlab R2013b on a 3.40 GHz machine with 4 GB of memory.

The experiment was implemented on 4 popular data sets, which have been widely used for evaluating semi-supervised algorithms. In particular, USPST data set is the testing set of USPS, which is a classical handwritten digit recognition data set.

Each data set was randomly divided into 4 equal folds. Each of the folds was used as the testing set once and the rest were used for training (4-fold cross-validation). The random generation process was repeated 3 times, so that there were 12 different experiment groups for each data set. For each group, the training set was split into 3 different folds again as Table 1. In Table 1, L is the labeled data set for training, U is the unlabeled data set, and V represents the validation set.

4.2 Comparisons with Related Algorithms

In the experiment, we compared the Kernel-SSELM and SSELM with the other state-of-the-art semi-supervised learning algorithms such as TSVM, LDS, LapRLS, and LapSVM. The validation set V was used to select the optimal model parameter for every algorithm. In particular, for Kernel-SSELM and SSELM, the cost coefficient C_0 and the semi-supervised rate λ were selected from the exponential

Table 1 Details of the division of the data sets

Dataset	Classes	Dims	L	U	V	T
G50C	2	50	50	314	50	136
G10N	2	10	50	314	50	136
COIL20	20	1024	40	1000	40	360
USPST	10	256	50	1409	50	498

Table 2 Performance comparison between different semi-supervised algorithms

Dataset	Subset	TSVM	LDS	LapRLS	LapSVM	SSELM	Kernel-SSELM
G50C	U	6.43 (2.11)	5.61 (1.46)	6.23 (1.52)	**5.16 (1.45)**	5.92 (2.34)	5.41(1.49)
	T	6.93 (2.37)	5.83 (2.03)	6.84 (2.41)	5.37 (1.56)	6.16 (2.87)	**5.23(1.91)**
G10N	U	13.91 (3.09)	9.79 (2.05)	**9.04 (2.31)**	9.27 (2.63)	9.96 (3.65)	9.17(1.86)
	T	14.36 (3.68)	9.72 (1.9)	**9.48 (2.63)**	9.82 (2.03)	10.44 (3.8)	9.83(2.15)
COIL20	U	26.35 (4.63)	14.68 (4.81)	**10.22 (4.17)**	10.53 (2.47)	11.41 (3.35)	10.62(2.04)
	T	25.87 (4.52)	15.09 (3.79)	11.3 (3.3)	11.59 (2.82)	12.05 (3.57)	**11.2(2.16)**
USPST	U	24.98 (4.89)	15.53 (3.35)	15.38 (4.17)	15.93 (3.56)	14.61 (3.89)	**13.81(2.47)**
	T	26.5 (4.69)	16.8 (3.54)	16.81 (3.28)	16.76 (3.98)	14.76 (3.64)	**13.43(1.95)**

Bold values indicate the best result in the dataset

sequence $\{10^{-6}, 10^{-5}, \ldots, 10^{6}\}$. The number of hidden layer nodes of SSELM was fixed to 1000 for G50C and G10 N, and 2000 for COIL20 and USPST. The Kernel function of the Kernel-SSELM was radial basis function (RBF), and its parameter γ was selected in $\{2^{0}, 2^{1}, \ldots, 2^{10}\}$.

Table 2 shows the error rate (with the standard deviation) of each algorithm. Kernel-SSELM and SSELM can achieve comparable result with the other 4 algorithms. Particularly, for the multi-class problems on the high dimension data such as COIL20 and USPST, Kernel-SSELM gave better performances than the others. Compared with SSELM, Kernel-SSELM yielded higher accuracy and lower deviation on all dataset. It is obvious to find that the algorithm with kernel function could build more stable model in classification task.

Table 3 displays the training efficiency of each algorithm on the 4 experiment datasets. SSELM was the fastest, while Kernel-SSELM was a bit slower but still stayed on the same level. On the binary-problem dataset, Kernel-SSELM and SSELM did not show much advantage to LapRLS, and LapSVM. This result in the they all need to calculate the Laplacian matrix which is a time consuming process

Table 3 Training time of different semi-supervised algorithms

Dataset	TSVM	LDS	LapRLS	LapSVM	SSELM	Kernel-SSELM
G50C	0.539	0.651	0.083	0.089	0.047	0.053
G10N	0.386	0.427	0.046	0.048	0.032	0.036
COIL20	34.32	39.18	11.98	8.367	1.201	1.634
USPST	188.7	205.3	15.27	13.84	2.932	3.524

and dominates the computation cost. However, for multi-class problem, the extreme learning methods showed significant advantage in training efficiency.

In all, from the two tables, we could found that Kernel-SSELM can give higher accuracy and stability in the cost of a little training speed.

5 Application in Traffic Congestion Evaluation

5.1 Traffic Congestion Evaluation

Urban Transportation Assessment and Forecast System analyzes the traffic congestion of transportation network in a city of southwest China and shows the evaluation results of the real-time traffic states on the GIS map using different colors on the foundation of the floating cars' GPS information (Fig. 1).

Seen from Fig. 2, traffic congestion evaluation system based on floating car data is the fundamental part of core function. In previous work, traditional method evaluating the present road congestion through fixed empirical evaluation standard is easy to implement and consumes a little system resources. But it does have the following drawbacks: First, the empirical evaluation frameworks do not take full consideration of the road information and network conditions. Second, it causes a significant gap between the congestion information on the map and users' experience.

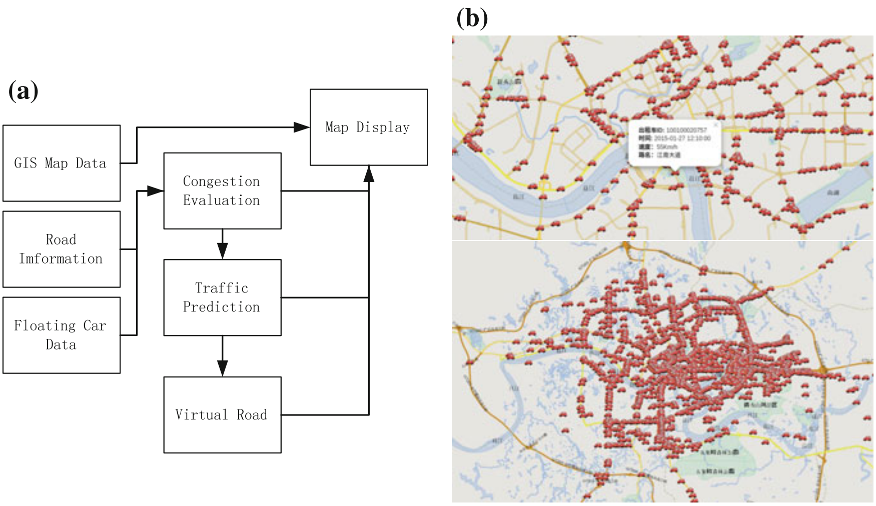

Fig. 1 Urban transportation assessment and forecast system. **a** Structure of urban transportation assessment and forecast system. **b** Floating car distribution on the map

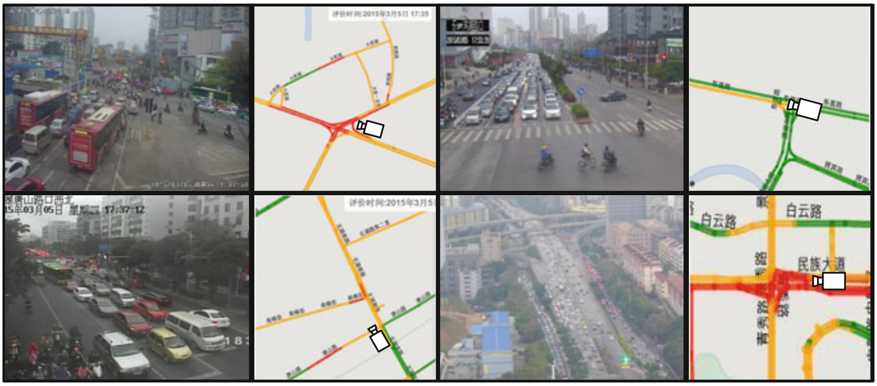

Fig. 2 Real-time traffic evaluation

To overcome the shortcomings above, machine learning methods are introduced into the traffic congestion evaluation system. However, many approaches such as SVM have deficiencies when applied to huge data and semi-supervised task in this traffic congestion evaluation system.

Applying Kernel-SSELM to traffic congestion evaluation system based on floating car data, this paper has the following strengths:

1. Though the congestion value of unlabeled data is uncertain, it represents the different traffic conditions which reflect the distribution information of traffic data. Kernel-SSELM improves the recognition accuracy of evaluation models by involving unlabeled data in the training.
2. Extreme learning machine has high training efficiency and is easy to implement. In the case of large data scales, high training speed ensures that despite traffic conditions changes it can still renew training for several times to choose a better model. At the same time, extreme learning machine is able to be modified into incremental learning easily so that we can make use of the latest information to update the evaluation network in real time.
3. With the neglecting the number of hidden layer nodes, the optimization of kernel function improves the stability of SSELM.

The evaluation system optimized by Kernel- SSELM improves the evaluation accuracy and is more in accord with the evaluation of congestions from local residents. As for the urban administration, the traffic congestion evaluation system plays an assistant role in management and supplies solutions for alleviating urban traffic. As for citizens, they may choose the right way to get around or the optimal driving route via the precise congestion evaluation.

5.2 Congestion Eigenvalue and Congestion Value

Traffic congestion evaluation system takes the road sections as the individual samples. Be specific, a road section demonstrates a portion of a road in a single direction. Its traffic congestion evaluation originates from two sources. The first part of source is the essential information of the road section from the Transportation Department, including Number of lanes, numbers of lanes of the entrance and exit, number of traffic lights and road grades. The second part of source is the real-time speed information of the road section from the floating car data, including average speed, speed distribution, and average stopping time.

The work of labeling training samples is completed by 5 experts from the Transportation Department of the city. Through surveillance cameras experts recorded information and gave evaluation of the traffic congestion at that time. Congestion evaluation is divided into three grades: Smooth, Average and Congested. The final label is in the grade which receives the most votes in 5 experts.

5.3 Evaluation Experiment

The environment of the experiment is the same as Sect. 4. In the experiment, we collect the floating car data from June 15th to June 16th 2015, and the quantity is more than 30,000,000. The data is grouped in interval for 5 min and matched to the corresponding road section. Finally we collect 13,681 samples. The evaluation of experts is based on the video from surveillance cameras about 30 typical road sections in the city. 537 valid samples were finally collected, and the rest 13,144 samples were unlabeled.

For comparison, we tested the SSELM, Kernel-SSELM and the empirical rule in Table 1. The test set had 100 samples randomly selected from the labeled sample, and the random generation process was repeated in 10 times. The cost coefficient C_0 was fixed to 100 and the semi-supervised rate λ was fixed to 0.001. The kernel function of Kernel-SSELM is RBF with the parameter γ fixed to 100. The number of hidden layer nodes of SSELM was set to 5000.

Table 4 shows that the evaluation model trained by Kernel-SSELM had the highest average accuracy at 86.2 %. In addition, Kernel-SSELM only takes 48.2 s for training, which keep the high training efficiency of SSELM.

Table 4 The result of evaluation experiment

	Empirical rule	SSELM	Kernel-SSELM
Average accuracy	68.9 %	82.6 %	86.2 %
Best accuracy	73 %	87.5 %	88 %
Training time	–	41.6	48.2

The trained model was used in the Urban Transportation Assessment and Forecast System. Figure 2 displays the real-time traffic condition. In the map, Green represents smooth traffic, yellow shows average condition, and red means the road is congested. Seen from the image taken by surveillance cameras, the traffic evaluation accurately reflects the road traffic congestion at that time.

6 Conclusion and Future Work

In this paper, a kernel based optimization is proposed to promote the SSELM. Experiments show that Kernel-SSELM can achieve higher accuracy and model stability, because kernel function avoids the problem of setting hidden layer. Compared with the other state-of-the-art semi-supervised learning algorithms, Kernel-SSELM shows significant advantages in training efficiency and multi-classification ability. In the application of traffic congestion evaluation, Kernel-SSELM was used to train the evaluation model on the large-scale data set. Both the experiment and the real-time application show the evaluation system can precisely reflect the traffic condition.

Since the type of kernel function and its parameter also have much influence on the training model, how to choose an optimized kernel function is still an important problem in the particular application of Kernel-SSELM. There is a general that assume a linear combination of a group of base kernels could be the optimal choice. In the future, we plan to research the multi-kernel framework for promoting the Kernel-SSELM in the traffic application.

Acknowledgment This work was supported by National Nature Science Foundation of P. R. China (No. 61272357, 61300074, 61572075).

References

1. Huang, G., Song, S., Gupta, J.N.: Semi-supervised and unsupervised extreme learning machines. IEEE Trans. Syst. Man. Cybern. Part B **44**(12), 2405–2417 (2014)
2. Huang, G.-B., Zhou, H., Ding, X., Zhang, R.: Extreme learning machine for regression and multiclass classification. IEEE Trans. Syst. Man Cybern. Part B **42**(2), 513–529 (2012)
3. Pal, M., Maxwell, A.E., Warner, T.A.: Kernel-based extreme learning machine for remote-sensing image classification. Remote Sens. Lett. **4**(9), 853–862 (2013)
4. Liu, J., Chen, Y., Liu, M., Zhao, Z.: SELM: semi-supervised ELM with application in sparse calibrated location estimation. Neurocomputing **74**(16), 2566–2572 (2011)
5. Li, L., Liu, D., Ouyang, J.: A new regularization classification method based on extreme learning machine in network data. J. Inf. Comput. Sci. **9**(12), 3351–3363 (2012)
6. Li, K., Zhang, J., Xu, H., Luo, S., Li, H.: A semi-supervised extreme learning machine method based on co-training. J. Comput. Inf. Syst. **9**(1), 207–214 (2013)
7. Gu, J., Zhou, S., Yan, X., et al.: Formation mechanism of traffic congestion in view of spatio-temporal agglomeration of residents' daily activities: a case study of Guangzhou. Sci. Geogr. Sinica **32**(8), 921–927 (2012)

8. Liu, X., Yin, J., Wang, L., Liu, L., Liu, J., Hou, C., Zhang, J.: An adaptive approach to learning optimal neighborhood kernels. IEEE Trans. Cybern. **43**(1), 371–384 (2013)
9. Cao, L.L., Huang, W.B., Sun, F.C.: Optimization-based multi kernel extreme learning for multimodal object image classification. International Conference on Multisensor Fusion and Information Integration for Intelligent Systems (MFI) pp. 1–9 (2014)