Ensemble Deep Learning for Regression and Time Series Forecasting

Conference Paper · December 2014

DOI: 10.1109/CIEL.2014.7015739

5 authors, including:

Xueheng Qiu
Nanyang Technological University
16 PUBLICATIONS **464** CITATIONS
SEE PROFILE

Le Zhang
Advanced Digital Sciences Center
64 PUBLICATIONS **1,960** CITATIONS
SEE PROFILE

Ponnuthurai N. Suganthan
Nanyang Technological University
515 PUBLICATIONS **37,077** CITATIONS
SEE PROFILE

Gehan Amaratunga
University of Cambridge
771 PUBLICATIONS **27,511** CITATIONS
SEE PROFILE

Ensemble Deep Learning for Regression and Time Series Forecasting

Xueheng Qiu, Le Zhang, Ye Ren and P. N. Suganthan
School of Electrical and Electronic Engineering
Nanyang Technological Univeristy, Singapore
{*qiux0004, lzhang027, re0003ye, epnsugan*}@ntu.edu.sg

Gehan Amaratunga
Department of Engineering
University of Cambridge, UK
gaja1@hermes.cam.ac.uk

***Abstract*—In this paper, for the first time, an ensemble of deep learning belief networks (DBN) is proposed for regression and time series forecasting. Another novel contribution is to aggregate the outputs from various DBNs by a support vector regression (SVR) model. We show the advantage of the proposed method on three electricity load demand datasets, one artificial time series dataset and three regression datasets over other benchmark methods.**

***Keywords*—Deep learning, Ensemble method, Time series forecasting, Regression, Load demand forecasting, Neural Networks, Support Vector Regression.**

TABLE I: Nomenclature

ANN	artificial neural network
FNN	feedforward neural network
RNN	recurrent neural network
CNN	convolutional neural network
ENN	ensemble feedforward neural network
SVM	support vector machine
SVR	support vector regression
DBN	deep belief network
RBM	restricted Boltzmann machine
SAE	stacked autoencoder
MAPE	mean absolute percentage error
RMSE	root mean square error
ARMA	auto-regressive moving average
ARIMA	auto-regressive integrated moving average

I. Introduction

Along with the fast development of electricity power market, electricity power industries are getting into free competitive area. Therefore, short-term load demand prediction is becoming important in such power systems. However, electricity load forecasting is challenging. There are many influencing issues such as climate factors and social activities which cause the data to be highly nonlinear [1], [2].

Load demand forecasting is a time series forecasting problem. Many statistics based linear time series forecasting models have been proposed since 1940s such as auto-regressive moving average (ARMA) [3]. Recently, artificial neural networks (ANNs) and other machine learning algorithms have been successfully applied to classification, regression and time series forecasting [4], [5]. Besides ANNs, support vector regression (SVR) has shown its advantages. SVR is known as a strong predictor as it is more likely to achieve the globally optimal solution compared to a weak predictor such as ANN which is frequently trapped in a local minimum. In [4], a multivariate peak load forecasting model was reported, where a least-square support vector machine (LSSVM) is employed. In addition, online learning was also implemented to offset the trend component in the time series. Al-Jamimi has developed two different high-performance fault prediction models based on Support Vector Machines (SVMs) and Probabilistic Neural Networks (PNNs) in [5]. Public NASA datasets from PROMISE repository have been used to evaluate the forecasting performance. This work has shown that PNN model provided better results for larger datasets.

Artificial Neural Network saw extensive use in the literature, but then fell out of fashion with the rise of Kernel based methods such as SVR as mentioned above. In 2006, Geoffrey Hinton et al. [6] rekindled interest in neural networks by showing substantially better performance by a ”deep” neural network. Since then, deep learning has become popular in machine learning field. Takashi Kuremoto has proposed a time series forecasting predictor model using DBN with multiple restricted Boltzmann machines [7]. The CATS benchmark data has been used in the form of 5 blocks with 20 missing and 980 known in each block. The model was then optimized by particle swarm optimization (PSO) algorithm. This work has shown DBN's superiority over conventional MLP neural network model and statistical model ARIMA. Busseti *et al.* also conducted simulations to compare deep learning methods with traditional shallow neural networks [8]. The work successfully showed the advantages of deep learning architectures to the problems of electricity load demand forecasting.

Ensemble methods, which work on a higher level to improve the performance of “unstable” predictors [9] such as decision tree and neural networks, have been successfully employed for solving pattern classification, regression, time series forecasting and fault prediction problems. Chatterjee developed an ensemble method for reliability forecasting of a mining machine [10]. This algorithm was based on least square support vector machine (LS-SVM) with hyper parameters optimized by a Genetic Algorithm (GA). The output of this model was generalized from a combination of multiple SVM predicted results. Turbocharger benchmark data sets were used to evaluate the performance. The outcome successfully showed the advantages of this ensemble method in fault prediction and reliability forecasting applications.

Regression analysis, which focuses on the relationship between an output variable and one or more input variables, is an active research field . The term regression was introduced by Francis Galton to describe a biological phenomenon which was also known as regression toward the mean [11]. In 1805

978-1-4799-4512-2/14/$31.00 ©2014 IEEE

and 1809, Legendre and Gauss published the earliest regression based on the method of least squares. There are numerous methods for performing regression. Moreover, Breiman has successfully showed that bagging could reduce variance of regression predictors [12]. Other researchers also demonstrated the advantages of ensemble method for regression from different viewpoints such as strength-correlation [13] or bias-variance [14].

To the best of our knowledge, there is no ensemble deep learning method for regression and time series forecasting in the literature. The proposed ensemble deep learning method consists of deep belief network (DBN) and support vector regression (SVR). More specifically, the SVR aggregates the outputs of different DBNs. The advantage of the proposed method is demonstrated on several benchmark datasets.

The remaining of this paper is organized as follows: Section II describes the commonly used methods. Section III presents the proposed ensemble deep learning method. Section IV summarizes comparison methodology and results. Finally in Section V, the conclusions and future work are stated.

II. Forecasting Models

A. Support Vector Regression

The Support Vector Machine (SVM) is a machine learning algorithm proposed by Cortes and Vapnik [15] based on statistical learning theory. Structural risk minimization is the basic concept of this method. A version of SVM for regression was proposed in [16]. Support vector regression (SVR) has been widely applied in time series prediction as well as power load demand forecasting and fault prediction [2].

Suppose a time series data set is given as follows for a power system

$$D = \{(X_i, y_i)\}, 1 \leqslant i \leqslant N \tag{1}$$

where X_i is the input vector at time i with m elements and y_i is the corresponding output data. The regression function can be defined as

$$f(X_i) = W^T \phi(X_i) + b \tag{2}$$

where W is the weight vector, b is the bias, and $\phi(X)$ maps the input vector X to a higher dimensional feature space. W and b can be obtained by solving the following optimization problem:

$$\text{Min}\, \frac{1}{2} \|W\|^2 + C \sum_{i=1}^{N} (\varepsilon_i + \varepsilon_i^*) \tag{3}$$

Subject to:

$$\begin{aligned} & y_i - W^T(\varphi(x)) - b \leq \xi + \varepsilon_i \\ & W^T(\varphi(x)) + b - y_i \leq \xi + \varepsilon_i^* \\ & \varepsilon_i, \varepsilon_i^* \geq 0 \end{aligned} \tag{4}$$

where C is a predefined positive trade-off parameter between model simplicity and generalization ability, ξ_i and ξ_i^* are the slack variables measuring the cost of the errors.

For nonlinear input data set, kernel functions can be used to map from original space onto a higher dimensional feature space in which a linear regression model can be built. Thus, the final SVR function is obtained as

$$y_i = f(X_i) = \sum_{i=1}^{N} ((\alpha_i - \alpha_i^*) K(X_i, X_j)) + b \tag{5}$$

where α_i and α_i^* are the Lagrange multipliers. The most frequently used kernel function is the Gaussian radial function (RBF) with a width of σ

$$K(X_i, X_j) = exp(-\|X_i - X_j\|^2 / (2\sigma^2)) \tag{6}$$

B. Artificial Neural Network

An ANN is a machine learning model inspired by the central nervous system. Fig. 1 is an illustration of a three-layer ANN. The first layer is the input layer which has the same number of neurons as the size of the input vector. The second layer is the hidden layer which has neurons with non-linear activation function. The third layer is the output layer which aggregates the outputs from the hidden layer neurons by a weighted summation. In addition, the connection between each neuron is weighted by adaptive weights which represent the strengths of the input to and output of neurons.

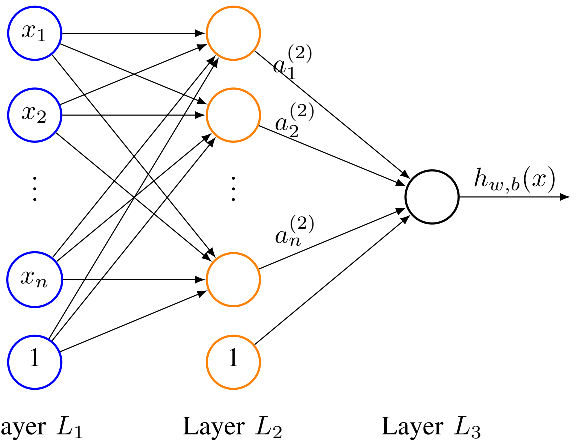

Fig. 1: Schematic of a Neural Network Model

The output from ANN is:

$$y_i = f(\sum_{i=1}^{n} w_i x_i + b_i) \tag{7}$$

where x_i is the input to the neuron, w_i is the weight of network, b_i is the bias, $f()$ is a nonlinear function and y_i is the output.

There are two types of ANNs which can be used in load demand forecasting: feedforward back-propagation neural networks (FNN) and recurrent neural networks (RNN).

1) feedforward Back Propagation Neural Network: The basic concept of FNN is quite simple: the network is supplied with both a set of input data to be learned and the desired output response for each data sample. If the networks output does not match the required target response, the weights are adjusted in an adaptive manner so that the error is minimized.

Therefore, future responses of the network are more likely to be correct. Back-propagation training is mathematically designed to minimize the mean square error (MSE) over all training data samples.

2) Recurrent Neural Network: There exist additional neurons in the input layer, which are called state or context neurons, to accept feedback connections. The role of context neurons in RNN is to get inputs from the upper layer, and after processing send their outputs to the hidden layer together with other inputs. Jordan RNN and Elman RNN are two of the most frequently used RNN models, which are also known as "simple recurrent networks" [17].

C. Deep Learning Algorithms

Deep learning algorithms are machine learning methods based on distributed representations. Deep learning attempts to learn high-level features in data by using structures composed of multiple non-linear transformations. The frequently used models are DBN, CNN and SAE.

1) Deep Belief Network: A DBN is a type of deep neural network which is composed of multiple layers of hidden units. There is no inter-connection between units in each layer [6]. Fig. 2 shows the schematic diagram of a deep belief network. DBN can be used to extract discriminant features in an unsupervised manner. Then a supervised learner such as softmax or SVM/SVR can be added on top of DBN.

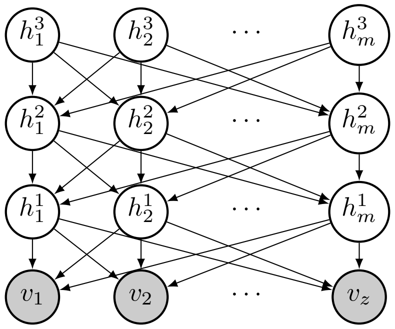

Fig. 2: Flowchart of a three-layer Deep Belief Network (DBN)

An RBM is a neural network which can learn the probability distribution over the input dataset. Fig. 3 shows the network structure of an RBM. v_z is the visible layer unit; h_m is the hidden layer unit; $W_{m \times n}$ represents the weights connecting hidden and visible units and b_z, c_m are the offsets of the visible and hidden layers, respectively.

2) Stacked Autoencoder: Architecturally, an autoencoder is simply a feedforward, non-recurrent neural network composed of an input layer, an output layer and one or more hidden layers [18]. The main difference between autoencoder and multilayer perceptron is that an autoencoder is trained to reconstruct its own input as its output. By setting the hidden layers narrower than the input layer, the network can learn useful features in the input data. Thus, it is an unsupervised machine learning method.

A stacked autoencoder is actually a neural network consisting of multiple layers of sparse autoencoders in which the outputs of each layer is wired to the inputs of the successive layer. The encoding step of each layer is:

$$\begin{aligned} a^{(l)} &= f(z^{(l)}) \\ z^{(l+1)} &= W^{(l,1)} a^{(l)} + b^{(l,1)} \end{aligned} \tag{8}$$

While the decoding step is given by

$$\begin{aligned} a^{(n+l)} &= f(z^{(n+l)}) \\ z^{(n+l+1)} &= W^{(n-l,2)} a^{(n+l)} + b^{(n-l,2)} \end{aligned} \tag{9}$$

where $a^{(l)}$ is the activation for the node in layer l, $z^{(l)}$ is the total weighted sum of inputs for the unit in layer l (for the first layer, z is the input data), $W^{(l,k)}$ is the weight values and $b^{(l,k)}$ is the bias.

A supervised neural network can be trained to predict at the output layer of an SAE based on the features learned by the SAE.

3) Convolutional Neural Network: A convolutional neural network (CNN) is also a type of feedforward neural network which is composed of alternating convolutional and subsampling layer [19]. Convolutional networks are designed to use minimal amounts of pre-processing, which is the main difference compared to other deep architectures. In the convolutional layer, a set of filter bank is employed to convolve with the input. This kind of "shared weights" mechanism can significantly reduce the parameter of the network, thus leading to better generalization ability.

D. Ensemble Method

An ensemble learning method is a machine learning process to obtain better prediction performance by strategically combining multiple learning algorithms [10]. There are three primary advantages brought by ensemble methods [20]. The first one is called statistical reason which is related to lack of sufficient data to properly represent the data distribution. Without sufficient data, many hypotheses which give the same training accuracy may be chosen as the learning algorithm. Ensemble methods can thus reduce the risk of selecting the wrong model by aggregating all these candidate models. The second is computational reason. Many learning algorithms, such as decision tree and neural network, work by performing some form of local search. These methods can frequently result in locally optimal solutions. Ensemble methods show their advantages in this scenario by running many local search from different starting points. The last reason is representational. In most cases, the true function f cannot be represented by any single hypothesis H. However, the function can be better approximated by a weighted sum of several hypotheses. The similar idea can be demonstrated by Fourier transformation which is widely used in signal processing.

III. Proposed Ensemble Deep Learning Method

For regression and time series forecasting, the prediction results can be different when the number of epochs of back propagation training is changed. Therefore, we can combine all the outputs generated by FNNs trained with different number of epochs. By analyzing the relationships between these outputs and target output values, it is possible to assign each output a corresponding weight value to compute the overall

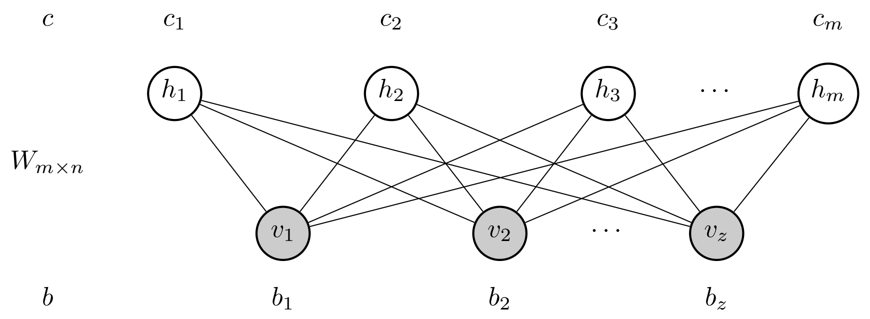

Fig. 3: Schematic Diagram of a Restricted Boltzmann Machine (RBM)

predicted output value. In this work, we choose an ensemble of deep learning algorithm composed of DBNs trained using different number of epochs and an SVR with inputs as the outputs of the DBNs and output as the final prediction. The detailed procedure is shown as follows:

1) Train a DBN by using the input data matrix $\mathbf{X}$.
2) By setting the back propagation epochs from 100 to 2000 with step size equal to 100, we are able to get 20 prediction outputs y_1 to y_{20} [21]. The DBN is re-initialized 20 times.
3) Put all the outputs into a matrix $\mathbf{X}_{new}$, which is used to train an SVR with the expected prediction values $\mathbf{Y}$.
4) Finally, we propose to get more accurate forecasting results.

Fig. 4 shows the overall schematic of this ensemble method.

IV. Results and Comparison

There are four time series datasets (Mackey-Glass dataset and three electricity load demand datasets) and three regression datasets (2D planes, Friedman Artificial Domain and California Housing datasets) used in the comparison.

The Mackey-Glass dataset is a time series generated by the Mackey-Glass equation to model the blood cell regulation. This dataset is widely used in the literature as a benchmark for prediction models. In the experiment, 9000 data points were used with first 6000 data points for training and the remaining 3000 data for testing.

The electricity load demand data sets from Australian Energy Market Operator (AEMO) were also used for the comparison [22]. Especially, the data sets of year 2013 from New South Wales (NSW), South Australia (SA) and Tasmania (TAS) were chosen to train and test the proposed method. The first nine months was used to train the model, and the last three months was used as the testing set. Thus, there are totally 13100 examples for training and 4370 examples for testing.

2D planes dataset is an artificial data set generated by equations introduced in [23] [24]. Friedman Artificial Domain dataset was first generated in [25] and also described in [12] [24]. For both of these datasets, 10000 data points were used with the first 7500 for training and the remaining 2500 for testing.

California Housing dataset is generated by collecting information on related variables in California from the 1990 Census [26] [24]. The final data contains 20640 observations with 9 inputs, while the output is the median house value. In this work, 15480 data points were used for training and the remaining 5160 data points were used for testing.

A. Methodology

For the time series load demand datasets, we put the demand data of last 24 hours as the input vector $\mathbf{X}_i$, while the corresponding y_i should be the demand value one step later. All the training and testing values are linearly scaled to [0, 1]. The transfer formula is

$$y_i' = \frac{y_{max} - y_i}{y_{max} - y_{min}} \tag{10}$$

To implement the experiment, we use the LIBSVM [27] to simulate the SVR model. For neural network, we use the deep learning toolbox in Matlab. Thus, the DBN and ensemble implementations are both developed from this toolbox [28].

To examine the accuracy of the prediction model, two evaluation measures are used in this study: Root Mean Square Error (RMSE) and Mean Absolute Percentage Error (MAPE). They are defined as:

$$\begin{aligned} RMSE &= \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i' - y_i)^2} \\ MAPE &= \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i' - y_i}{y_i}\right| \end{aligned} \tag{11}$$

where y_i' is the predicted value of corresponding y_i.

B. Results

From prediction results for Mackey-Glass data set in Table II, the effectiveness of all of these prediction methods can be appreciated.

The comparison results for time series load demand forecasting are shown in Tables III, IV and V. For SVR, we choose the RBF kernel function with parameters chosen by a grid search. The range of C is [2^{-4},2^{4}], and the range of σ is [10^{-3},10^{-1}]. For FNN, the size of the neural network is [48 96 1], which is a one hidden layer model. The sizes of

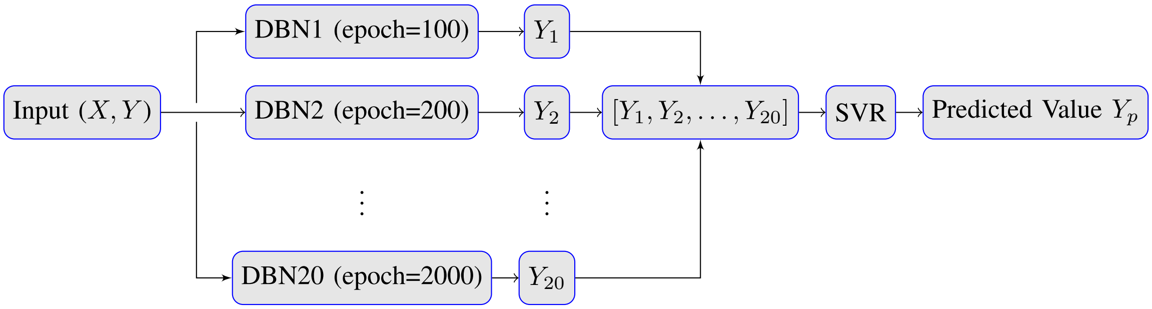

Fig. 4: Schematic Diagram of the proposed Ensemble Deep Learning Network

TABLE II: Prediction results for Mackey-Glass Time Series

Mackey-Glass	SVR	FNN	DBN	ENN	Proposed
CV RMSE	0.0025	0.002	0.0018	0.0012	**0.0015**
RMSE	0.0024	0.002	0.0018	0.0226	**0.0015**
CV MAPE	1.25%	2.33%	2.06%	0.87%	**0.43%**
MAPE	1.03%	2.45%	2.17%	1.13%	**0.43%**

RBM in DBN is [20 20]. For the proposed ensemble learning method, 20 DBNs and the SVR have the same parameters as listed above. ENN is the ensemble version of the 20 FNNs (trained using epochs ranging from 100 to 2000) and combined using an SVR. The last column named AEMO shows the average forecasting error in year 2013 given on the AEMO website [22].

By analyzing the forecasting outputs in Tables III to V, we can find that these methods also perform well for short-term time series load demand forecasting. Moreover, The ensemble learning methods have more accurate outputs than single structure algorithms. However, SVR has a slightly better forecasting performance than artificial neural networks for these data sets. This phenomenon is probably caused by the reason that there is only one hidden layer in the neural networks used here [29]. The prediction results of DBN are better than FNN, which shows the advantage of deep learning methods. Most outstandingly, the proposed ensemble deep learning method composed of DBN and SVR has yielded both the best training reconstruction results and the most accurate prediction outputs.

TABLE III: Prediction results for load demand of New South Wales

NSW	SVR	FNN	DBN	ENN	Proposed	AEMO
CV RMSE	75.5476	99.4513	90.4974	79.1079	**59.8561**	/
RMSE	74.3053	95.8105	90.2061	78.6394	**72.2545**	/
CV MAPE	2.25%	3.00%	2.73%	2.34%	**1.79%**	2.00%
MAPE	2.83%	4.12%	3.50%	2.96%	**2.71%**	/

Similar to time series part, the comparison results for regression are shown in Tables VI, VII and VIII. For 2D plane and Friedman datasets, the size of the neural network is [10 20 1], while the size for California housing is [9 18 1]. The rest of parameters are the same as before. To make the numbers more comparable, the RMSE values for regression forecasting were calculated using scaled data.

TABLE IV: Prediction results for load demand of South Australia

SA	SVR	FNN	DBN	ENN	Proposed	AEMO
CV RMSE	40.6467	36.8863	33.1023	32.3606	**26.6159**	/
RMSE	44.6742	38.8585	35.9375	34.9473	**30.5989**	/
CV MAPE	3.64%	4.38%	3.74%	3.63%	**3.35%**	4.53%
MAPE	5.30%	6.22%	5.70%	5.32%	**4.98%**	/

TABLE V: Prediction results for load demand of Tasmania

TAS	SVR	FNN	DBN	ENN	Proposed	AEMO
CV RMSE	18.7509	18.9368	19.0076	19.9086	**18.3066**	/
RMSE	20.1068	19.7952	19.9187	19.9034	**19.7580**	/
CV MAPE	3.00%	3.06%	3.05%	3.06%	**3.01%**	3.17%
MAPE	3.43%	3.41%	3.41%	3.41%	**3.38%**	/

From prediction results for regression, we can have similar conclusions as in time series forecasting. Especially, for California Housing result in Table VIII, the advantage of ensemble deep learning method is outstanding. Therefore, compared with the performance on simple equation generated datasets, the ensemble deep learning method demonstrates much stronger ability on real complicated regression problems.

TABLE VI: Prediction results for 2D planes dataset

CART	SVR	FNN	DBN	ENN	Proposed
CV RMSE	0.0399	0.0425	0.0397	0.0402	**0.0321**
RMSE	0.0406	0.0420	0.0412	0.0428	**0.0403**
CV MAPE	7.52%	8.61%	7.50%	7.56%	**6.11%**
MAPE	7.49%	8.16%	7.59%	7.61%	**7.49%**

TABLE VII: Prediction results for Friedman Artificial Domain dataset

FAD	SVR	FNN	DBN	ENN	Proposed
CV RMSE	0.0304	0.0350	0.0310	0.0314	**0.0300**
RMSE	0.0339	0.0349	0.0320	0.0315	**0.0313**
CV MAPE	5.59%	6.75%	5.87%	5.85%	**5.51%**
MAPE	6.38%	6.82%	6.14%	5.96%	**5.93%**

V. Conclusion

In this paper, for the first time, we proposed an ensemble deep learning method by combining DBN and SVR. The proposed method has been evaluated with Mackey-Glass time

TABLE VIII: Prediction results for California Housing

CH	SVR	FNN	DBN	ENN	Proposed
CV RMSE	0.1389	0.1209	0.0926	0.1111	**0.0834**
RMSE	0.1637	0.1803	0.1773	0.1615	**0.1508**
CV MAPE	29.33%	27.06%	21.00%	22.46%	**17.17%**
MAPE	28.36%	33.31%	32.26%	29.44%	**27.33%**

series dataset, three electricity load demand datasets and three regression datasets. The proposed method has been compared with four benchmark methods: SVR, feedforward NN, DBN and ensemble feedforward NN. Based on RMSE and MASE, the proposed ensemble deep learning method has outperformed the four benchmark methods for both time series and regression datasets. In addition, the proposed method has the potential ability to deal with massive and more complicated datasets.

In modern society, accurate electricity load demand forecasting is an important guide for effective implementations of energy policies. Therefore, for future work, ensemble methods composed of different deep learning algorithms will be implemented and compared with various types of regression and time series datasets from the power industry. Moreover, more advanced optimization algorithms for parameter selection will be developed to further enhance the prediction process.

Acknowledgment

The authors would like to thank the Cambridge Centre for Carbon Reduction in Chemical Technology (C4T) project for funding and scholarship. The authors would also like to thank the Clean Energy Program Office (CEPO) for scholarship.

References

[1] W.-C. Hong, Y. Dong, L.-Y. Chen, and S.-Y. Wei, "Seasonal support vector regression with chaotic genetic algorithm in electric load forecasting," in *Proceedings of the 2012 Sixth International Conference on Genetic and Evolutionary Computing*. IEEE Computer Society, 2012, pp. 124–127.

[2] J. C. Sousa, H. M. Jorge, and L. P. Neves, "Short-term load forecasting based on support vector regression and load profiling," *International Journal of Energy Research*, vol. 38, no. 3, pp. 350–362, 2014.

[3] G. E. Box, G. M. Jenkins, and G. C. Reinsel, *Time series analysis: forecasting and control*. John Wiley & Sons, 2013.

[4] Z. Aung, M. Toukhy, J. R. Williams, A. Sanchez, and S. Herrero, "Towards accurate electricity load forecasting in smart grids," in *The Fourth International Conference on Advances in Databases, Knowledge, and Data Applications (DBKDA2012)*, 2012, pp. 51–57.

[5] H. A. Al-Jamimi and L. Ghouti, "Efficient prediction of software fault proneness modules using support vector machines and probabilistic neural networks," in *5th Malaysian Conference in Software Engineering (MySEC2011)*, 2011, pp. 251–255.

[6] G. Hinton, S. Osindero, and Y.-W. Teh, "A fast learning algorithm for deep belief nets," *Neural computation*, vol. 18, no. 7, pp. 1527–1554, 2006.

[7] T. Kuremoto, S. Kimura, K. Kobayashi, and M. Obayashi, "Time series forecasting using restricted boltzmann machine," in *Emerging Intelligent Computing Technology and Applications*. Springer, 2012, pp. 17–22.

[8] E. Busseti, I. Osband, and S. Wong, "Deep learning for time series modeling," Technical report, Stanford University, Tech. Rep., 2012.

[9] L. Breiman, "Bias, variance, and arcing classifiers," 1996.

[10] S. Chatterjee, A. Dash, and S. Bandopadhyay, "Ensemble support vector machine algorithm for reliability estimation of a mining machine," *Quality and Reliability Engineering International*, 2014.

[11] F. Galton, "Kinship and correlation (reprinted 1989)," in *Statistical Science*, 1989, vol. 4, no. 2, pp. 80–86.

[12] L. Breiman, "Bagging predictors," in *Machine Learning*. Kluwer Academic Publishers, 1996, vol. 24, no. 3, pp. 123–140.

[13] ——, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[14] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," *Machine learning*, vol. 63, no. 1, pp. 3–42, 2006.

[15] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[16] H. Drucker, C. J. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support vector regression machines," *Advances in neural information processing systems*, vol. 9, pp. 155–161, 1997.

[17] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, and J. Schmidhuber, "A novel connectionist system for unconstrained handwriting recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 855–868, 2009.

[18] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion," *The Journal of Machine Learning Research*, vol. 11, pp. 3371–3408, 2010.

[19] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[20] T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple classifier systems*. Springer, 2000, pp. 1–15.

[21] J. Xie, B. Xu, and Z. Chuang, "Horizontal and vertical ensemble with deep representation for classification," *arXiv preprint arXiv:1306.2759*, 2013.

[22] (2013, Dec.) Australian energy market operator. [Online]. Available: http://www.aemo.com.au/

[23] L. Breiman, J. H. Freidman, R. A. Olshen, and C. J. Stone, "Classification and regression trees." Wadsworth, 1984.

[24] L. Torgo. (2014) Regression datasets. [Online]. Available: http://www.dcc.fc.up.pt/~ltorgo/Regression/DataSets.html

[25] J. FRIEDMAN, "Multivariate adaptative regression splines," in *Annals of Statistics*, 1991, vol. 19, no. 1, pp. 1–141.

[26] R. K. Pace and R. Barry, "Sparse spatial autoregressions," in *Statistics and Probability Letters*. StatLib repository, 1997, vol. 33, pp. 291–297.

[27] C.-C. Chang and C.-J. Lin, "Libsvm: a library for support vector machines," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 2, no. 3, p. 27, 2011.

[28] R. B. Palm, "Prediction as a candidate for learning deep hierarchical models of data," Master's thesis, 2012.

[29] E. Romero and D. Toppo, "Comparing support vector machines and feedforward neural networks with similar hidden-layer weights," *IEEE Transactions on Neural Networks*, vol. 18, no. 3, pp. 959–963, 2007.