FACTA UNIVERSITATIS (NIŠ)
SER.: ELEC. ENERG. vol. 23, no. 3, Decemer 2010, 287-298

Mid-Term Load Forecasting Using Recursive Time Series Prediction Strategy With Support Vector Machines

Miloš B. Stojanović, Miloš M. Božić, and Milena M. Stanković

Abstract: Medium term load forecasting, using recursive time - series prediction strategy with Support Vector Machines (SVMs) is presented in this paper. The forecasting is performed for electrical maximum daily load for the period of one month. The data considered for forecasting consist of half hour daily loads and daily average temperatures for period of one year. An analysis of available data was performed and the most adequate set of features for our model are chosen. For evaluation of prediction accuracy we used data obtained from electricity load forecasting competition on the EUNITE network. Some drawn conclusions from the results are that the temperature significantly affects on load demand, but absence of future temperature information can be overcome with time - series concept. Also, it was shown that size and structure of the training set for SVM may significantly affect the accuracy of load forecasting.

Keywords: Load forecasting, support vector machines, time series, regression.

1 Introduction

FOR electric utilities it is important to have accurate load forecasting for different time periods. With the deregulation of the energy industries, load forecasting is even more important especially for dispatcher who can make better decisions and comply with them. Thus, electric utilities reduce occurrences of equipment failures and blackouts.

Forecasting depending on a time period can be generally divided into three types: long term, medium term and short term. Each type has important role on

Manuscript received on August 28, 2010.

M. Stojanović is with Technical College of vocational studies, A. Medvedeva 20, 18000 Niš, Serbia, (e-mail: milos.stojanovic@vtsnis.edu.rs). M. Božić is with Faculty of Electronics Engineering, A. Medvedeva 14, 18000 Niš, Serbia, (e-mail: milos1bozic@yahoo.com). M. Stanković is with Faculty of Electronics Engineering, A. Medvedeva 14, 18000 Niš, Serbia, (e-mail: milena.stankovic@elfak.ni.ac.rs).

economic and reliable operation of electric utilities. Whether to build or upgrade new lines and sub-stations may be shown by long term prediction, which is made for period of one to several years. Medium term load forecasting is related to time period from few days to few weeks or mounts and it is used to meet the load requirements at the peak of summer or winter season. Short term load forecasting is important for operations like real time generation control, security analysis and energy transaction planning. Usually short term load forecasting covers a period from one hour to few days.

Most forecasting techniques have already been tried out on load forecasting with different degree of success. Some of these techniques, which are especially popular in recent years, are: neural networks, fuzzy logic and expert systems. The usage of Artificial Neural Networks (ANNs) [1] has been a widely studied electric load forecasting technique. More recent powerful machine learning techniques for electric load forecasting are SVMs [2], which are one of the significant developments in overcoming shortcomings of ANNs. Unlike ANNs, which try to define complex functions of the input feature space, SVMs perform a nonlinear mapping (by using so-called kernel functions) of the data into a high dimensional space. SVMs then use simple linear functions to create linear decision boundaries in the new space. Thus, a nonlinear problem in the original lower dimensional input space could find its linear solution in the higher dimensional feature space. The problem of choosing architecture for ANN is replaced in SVM by the problem of choosing a suitable kernel. Various modifications of SVMs occur in solutions of electric load forecasting. Also, combination of other artificial intelligence techniques with SVMs can be found in [3, 4]. In this paper, an attempt is being made to predict maximum daily load for period of one month, using different data sets and features. The paper is organized as follows: Section 2 demonstrates the techniques we employed. Section 3 explains our proposed model. Section 4 describes derived experiments. Section 5 presents experimental results and provides an evaluation of our model and section 6 presents the conclusions.

2 Methods

2.1 Time series prediction

Predicting time series is of great importance in many domains of science and engineering, such as finance, electricity, environment and ecology. A time series is a sequence of observations made through time, in the form of vector or scalar [5].

Time series prediction can be considered as a problem of a model formation that establishes a mapping between the input and output values. After such model is formed, it can be used to predict the future values based on the previous and current values. The previous and the current values of the time series are used as inputs for the prediction model:

$$\{y(t+1), y(t+2), \dots, y(t+h)\} = F(y(t), y(t-1), \dots, y(t-m+1)) \qquad (1)$$

where h represents the number of ahead predictions, F is prediction model and m is size of regressor.

Time series prediction can be divided into two categories depending on prediction time period: short term and long term [6]. Short term prediction is related to one "step ahead prediction." The goal of long term prediction is to predict values for several steps ahead. In long term prediction propagation of errors and the deficiency of information occur, which makes the prediction more difficult. For long-term prediction there are two different approaches that can be used: direct and recursive. In following section, an approach for recursive prediction is presented.

2.2 Recursive prediction strategy

Recursive prediction strategy uses the predicted values as known data to predict the next ones [7]. The model can be constructed by making one-step ahead prediction:

$$y(t+1) = F(y(t), y(t-1), \dots, y(t-m+1)). \qquad (2)$$

The regressor of the model is defined as the vector of inputs: $y(t), y(t-1), \dots, y(t-m+1)$, where m is the size of regressor. To predict the next value, the same model is used:

$$y(t+2) = F(y(t+1), y(t), \dots, y(t-m+2)), \qquad (3)$$

and for h^{th} prediction:

$$y(t+h) = F(y(t+h-1), y(t+h-2), \dots, y(t-m+h)). \qquad (4)$$

It is important to notice in 3, that the predicted value of $y(t+1)$ is used instead of the true value, which is unknown. Then, h steps ahead predictions, from $y(t+2)$ to $y(t+h)$ are predicted iteratively. Thus when the regressor length m is larger than h, there are $m-h$ real data in the regressor to predict the h^{th} step. But when h becomes larger then m, all the inputs are the predicted values. Usage of the predicted values as inputs affects on the accuracy of the prediction in cases when h significantly exceeds m.

2.3 SVM

SVMs are developed based on statistical learning theory given by Vapnik [8] in 1995 to resolve the issue of data classification. Two years later, the version of SVM is proposed that can be successfully applied to the data regression problem. This method is called Support Vector Regression (SVR) and it is the most common form of SVMs that is applied in practice [9].

SVMs are based on the principle of structural risk minimization (SRM), which is proved to be more efficient than the empirical risk minimization (ERM), which is used in neural networks. SRM minimizes an upper bound of expected risk as opposed to ERM that minimizes the error on the training data [10].

SVMs implement a learning algorithm that performs learning from examples in order to predict the values on previously unseen data. The goal of SVR is to generate a model which will perform prediction of unknown output values based on the known input parameters. In the learning phase, the formation of the model is performed based on the known training data $(x_1,y_1),(x_2,y_2),\dots,(x_n,y_n)$ where x_i are input vectors, and y_i outputs associated to them. Each input vector consists of numeric features. In the phase of application, the trained model on the basis of new inputs $x_1,x_2,\dots,x_n$ makes prediction of output values $y_1,y_2,\dots,y_n$. SVR is an optimization problem [9], in which is needed to determine the parameters ω and b to minimize:

$$\min_{\omega,b,\xi,\xi^*} \frac{1}{2}\omega^T\omega + C\sum_{i=1}^{n}(\xi_i+\xi_i^*) \tag{5}$$

with conditions:

$$y_i - (\omega^T\phi(x_i)+b) \leq \varepsilon + \xi_i, \tag{6}$$

$$(\omega^T\phi(x_i)+b) - y_i \leq \varepsilon + \xi_i^*, \tag{7}$$

$$\xi_i, \xi_i^* \geq 0, i = 1,2,\dots,n,$$

where x_i is mapped in the multi-dimensional vector space with non linear mapping ϕ. ξ_i is the upper limit of training error and ξ_i^* the lower. The idea of SVR is based on the computation of a linear regression function in a high dimensional feature space where the input data are mapped via a nonlinear function.

The parameters that control the quality of the regression are kernel function, C and ε. C is parameter which determines the ”cost of error”, i.e. determines the tradeoff between the model complexity and the degree to which deviations larger than ε are tolerated [11]. A larger values for C reduces the error on training data but yields a more complex forecasting function that is more likely to overfit on the training data [12]. Parameter ε controls the width of ε insensitive zone, and hence the number of support vectors (vectors that lying on a margin of the tube) and errors that lying outside ε zone [13].

The goal of SVR is to place as many input data inside the tube $|y-(\omega^T\phi(x)+b)| \leq \varepsilon$, which is shown in Fig. 1. If x_i is not inside the tube, an error occurs ξ_i or

ξ_i^*. Loss function assigns errors only to those x_i for which $\xi_i \geq \varepsilon$ or $\xi_i^* \geq \varepsilon$ [12], and it is defined with:

$$|\xi_i| = \begin{cases} 0, & \text{if } |\xi| < \varepsilon \\ |\xi| - \varepsilon, & \text{else} \end{cases} \tag{8}$$

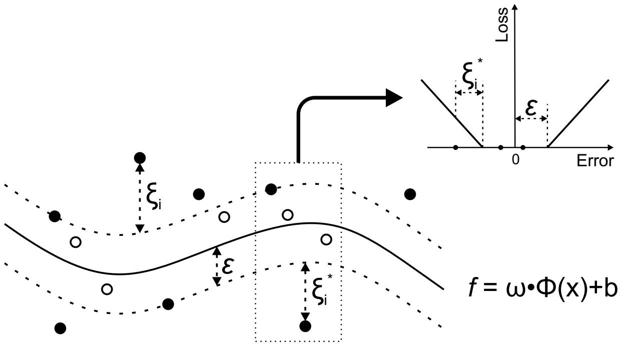

Fig. 1. ε tube of nonlinear SVR.

The problem can be solved using Langrange multipliers [9], and the solution is defined with:

$$f(x) = \sum_{i=1}^{n} (\alpha_i^* - \alpha_i) K(x_i, x) + b \tag{9}$$

where $K(x_i, x)$ represents kernel function, defined as dot product between $\phi(x_i)^T$ and $\phi(x)$. More about SVR can be found in [9], [12].

For the experiments we used a publicly available library LibSVM - A Library for Support Vector Machines [14] which we integrated in our software for recursive time series prediction.

3 Data Analysis

Before selecting features for our model, some observations about the data are examined first. Relations between load demand and other information, such as climate influence or local events are observed and these relations should be taken into account during the formation of model. The data used for experiments were provided by the Eastern Slovakian Electricity Corporation for the EUNITE competition [15].

Properties of Load Demand Load demand data given are half-hour recordings for one year. In Fig 2. maximum daily loads and average daily temperatures for the period of one year are presented.

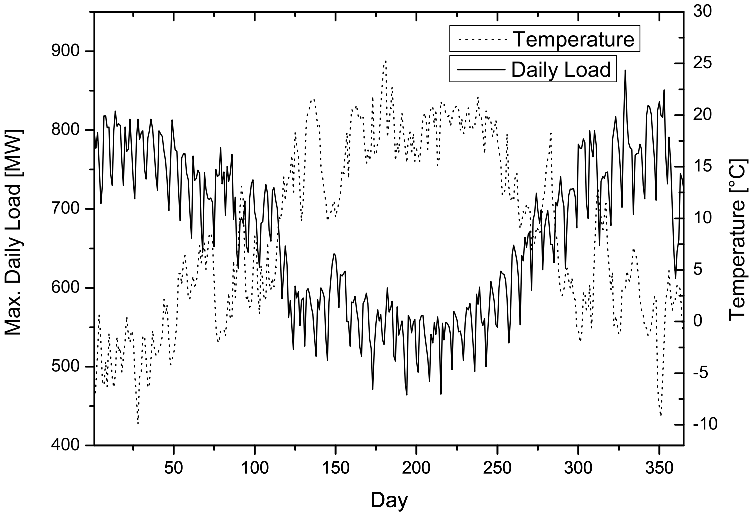

Fig. 2. Load pattern and temperature during period of one year.

Relations between electricity usage and weather conditions in different seasons are observed from data in Fig 2.Load demand in winter period is higher compared to summer period. Also, in winter period of the year variance of load demand is high with many peaks while relatively constant in summer period. Reason for this behavior of load demand is mainly temperature variations.

Additionally, another load pattern could be observed, from Fig 3. We can notice that the load almost periodically exists in every week with pattern where load demand in weekend is usually lower than in weekdays. In Fig 3. weekly load patterns for weeks in January and July are given.

Climate influence Climate conditions influence on load demand pattern analysis may include temperature, wind speed, humidity, pressure and illumination. In this paper the most important climate parameter is considered, temperature. Relation between climate and load demand also was shown in previous works on short term load forecasting [1, 3]. From Fig 2. it can be easily observed negative correlation between maximum daily load and temperature. Reason for that is heating usage in winter period of year.

Holiday influence Holidays and other local events also may affect the load demand [2]. These events are usually local and their influences highly depend on the customs of the area. Major holidays such as Christmas or New Year have more influences on load demand then other holidays.

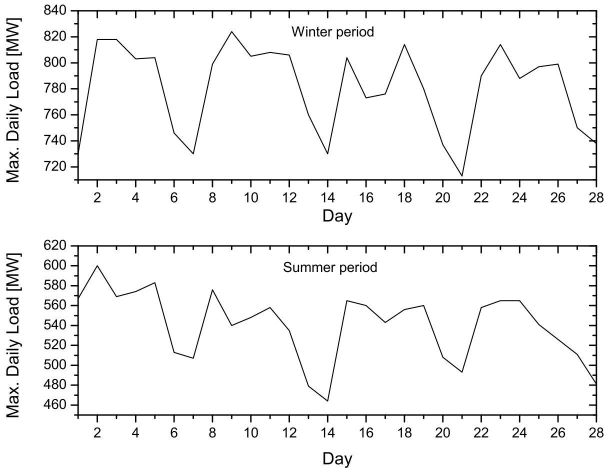

Fig. 3. Weakly load patterns in winter and summer period.

4 Experiments

In this section, we consider what kind of information should be used for features to build appropriate SVM models.

Calendar attributes. We mentioned earlier that load is periodic by week. On the other hand, many works [1, 3] have used the calendar information to model the load forecasting problem. Because of that, we decide to use day of the week as feature in some models and see how it affects on the accuracy of prediction. Additionally, holiday information can be useful to model the problem because load demand on holidays might be lower than that on non-holidays. But, small number of holidays during the year makes the problem for SVM to train the model well.

Temperature. Weather information which includes temperature, wind speed, sky cover, humidity and etc., has also been used in most short - term load forecasting works [3]. But there is one difficulty: for mid-term load forecasting, temperature information for several weeks away is needed. If we want to use temperature information in our model, we will also need to predict the temperature. The usage of temperature, therefore, is not an option when forecasting is performed for period longer than one week. Yet, temperature forecasting is a more complex problem than load forecasting.

Time series. Another information which we considered to encode as the feature in our model is the past load demand. Reason for this is that the past load demand could affect and imply the future load demand. With this approach, concept of time-series is introduced into our models [2]. We used for regressor size $m = 7$, which means that seven past daily load demands are used for features.

The various factors that affect the load forecast are analyzed and appropriate features are chosen. Then we form vectors which will be used as inputs for SVM. Our proposed model is shown in Fig 4. Input vectors for SVM model is composed of the following features:

- Maximum daily load for past seven days $(P_{i-k}), k = 1, \ldots, 7$,
- Average daily temperatures (T),
- Day Of the Week (D).

Data are scaled before training to range [-1,1] [14]. Day of the week is coded with numeric values from 1 to 7, where 1 represents Monday, 2 Tuesday, ..., etc.

All features except time - series are used optionally, i.e. several models are formed with different combinations of features.

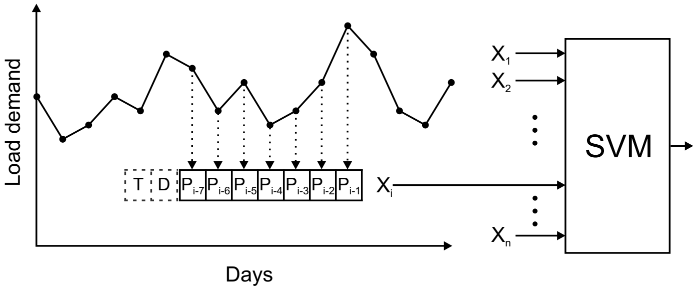

Fig. 4. Proposed architecture.

On the basis of established vectors and the known values of the load in the selected time interval SVM generates model which forecasts the load for a period of one month.

Forecasting performance of the SVM significantly depends from selection of their parameters. In addition to the selected features, in order to train the model, we should determine the kernel function and its parameters, and SVR parameters C and ε.

Radial basis function (RBF) [16] for the kernel is chosen, defined with:

$$K(x_i, x) = e^{-\gamma||x_i - x||^2}, \gamma > 0 \tag{10}$$

The kernel parameter γ defines the width of the kernel to reflect the range of the training data in feature space and therefore the ability of an SVR to adapt to the data [11].

It is necessary to determine the parameters C and γ. It is not known in advance which values of these parameters are best choice for a given problem. The parameters were determined using Grid-Search [14] and Cross-Validation [17].

A simple method to determine SVM parameters is a systematic Grid-Search over the parameter space. Instead of evaluating every possible parameter combination, which would be time consuming, a grid using equidistant steps in the parameter space and limits the search complexity. In our experiments we search couples C and γ exponentially in range $C = 2^{-5} - 2^{20}$ and $\gamma = 2^{-15} - 2^{3}$ with steps 0.5 and 0.1.

For evaluation how model performs prediction with different pairs of parameters C and γ, k fold Cross-Validation procedure is used. The training set is randomly divided into training and testing parts, in relation $1 : k$. Then the learning algorithm is applied to training part with current values of C and γ. Then, evaluation of the prediction quality is performed on testing part. This procedure is repeated k times and a pair C and γ is selected, with which the best prediction is achieved.

In time series prediction problems ε corresponds to the level of noise in a time series [13]. Large values of ε allow an approximation of a time series with high noise as opposed to overfitting the noise. In our approach ε was not included in Grid-Search procedure, so we tune ε ”manually” for each test set.

After training several different SVM models based on different training sets, we used them for recursive prediction of load demand. For experiments described in this work we develop an application that performs recursive time series prediction with SVR. Also, other features can be included in prediction, which are not time series type (e.g. temperature, day of the week, ...). Pseudo code of recursive prediction of load demand for given period of time is given below:

```
i=0
STATIC_VECTOR=[$D_i$,$T_i$]
TIME_VECTOR=[$P_{n-7}, P_{n-6}, \ldots, P_{n-1}$]
WHILE (i$\leq$NUMBER_OF_PREDICTIONS)
BEGIN
INPUT_VECTOR = CONCATENATE(STATIC_VECTOR, TIME_VECTOR)
$P_i$ = SVR(INPUT_VCTOR)
RESULTS[i] = $P_i$
TIME_VECTOR = SHIFTLEFT(TIME_VECTOR)
TIME_VECTOR[6] =$P_i$
i=i+1
STATIC_VECTOR = [$D_i$,$T_i$]
END
VALIDATE (RESULTS)
CALCULATE MAPE
PRINT (RESULTS)
```

5 Experimental Results

In electricity load forecasting, the prediction accuracy is generally evaluated using Mean Absolute Percentage Error (MAPE) [2], [18]. The equation describing this error is:

$$MAPE = 100\frac{1}{n}\sum_{i=1}^{n}\left|\frac{P_i-\hat{P}_i}{P_i}\right|, \tag{11}$$

where P_i and $\hat{P}_i$ are the real and the predicted value of maximum daily electrical load on the i^{th} day and n is the number of predictions.

To evaluate the accuracy of the model, maximum daily load forecasting for one month ahead (January) is done. Training model was committed with several different data encodings and segmentations. Table 1 shows MAPE errors generated by different data encodings and segmentations. The first column represents used data segments while first row shows which features are used. Additional, estimated and real load demand for January, using ”Winter” segment, are shown in Fig. 5.

Table 1. MAPE using different data preparation for one month forecasting.

Segment	Temp&Day&TS	Day&TS	TS
Winter	1.79	2.23	3.94
Dec-Jan-Feb	2.64	3.32	4.62
All	2.52	6.32	5.52

In Table 1, it can be observed that models built with temperature feature, exceed all other. This is expected because load demand is closely related to the temperature. Of course, it should be noticed that for the model testing (forecasting load demand for given period) real temperature is used. It is not a problem when forecasting period is one week. In that matter it can be used predicted temperature because it is relatively close to real temperature. The problem occurs when forecasting period is several weeks or month. Then the temperature is not available because one month ahead temperature prediction is not precise as one week ahead prediction.

Unavailability of precise temperature for one month ahead, has led to usage other features with the aim of reducing MAPE error. Table 1 shows that model built with time-series and day features, using the ”Winter” segment, exceeds all other models without temperature feature. MAPE error is higher in case when models were trained with ”Dec-Jan-Feb” segment, because that data segment cannot provide enough information for models compared with the ”Winter” segment that contains more entries. On the other hand, when the model is trained with ”All” segment, which is formed from data of whole year, MAPE error is significantly higher because this training set adds noise into the model. It is because summer period is present in ”All” segment.

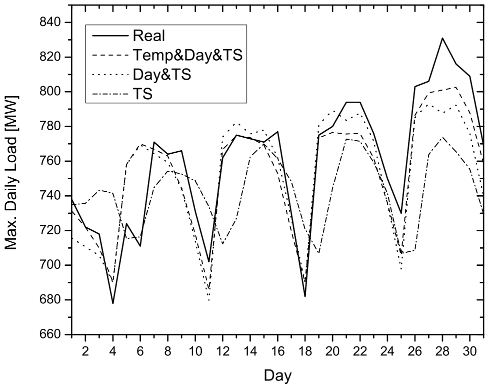

Fig. 5. Estimates and real load demand for Jan.

As previously stated, from Fig 5 it can be observed that the best result is obtained with the model build using time - series and day features. The most deviations from real load are in days 1 and 6. Reason for that is these days are holidays. At first, holiday information has been included in our model. The problem is that there is insufficient data on holidays to train the model well, and it makes predictions even harder on these days. That is why we removed holiday information from our models, so in days of holidays we have error, but in other days error is significantly reduced.

6 Conclusion

Several SVM models are formed using different features in combinations with different data segments for training. We find out that choice of appropriate data segments may improve performance of models, because the load demand has different distribution in different seasons. Furthermore, models build with climate information (temperature) require future climate data for several weeks away. This difficult may lead to inaccurate prediction, and because of that models without temperature feature were build. Rejection temperature feature from models increases models error. However, this difficulty is overcome using time - series strategy and obtained results are close to results with temperature feature. Though, models for one month ahead prediction require further investigation in the direction of selection appropriate features and data segments. Also it is necessary to find a way how to involve holiday information in models to obtain better prediction in days of holidays.

References

[1] D. O. Arroyo, M. K. Skov, and Q. Huynh, “Accurate electricity load forecasting with artificial neural networks,” in *Proceedings of the 2005 International Conference on Computational Intelligence for Modeling, Control and Automation*, Vienna, Austria, Nov. 2005.

[2] B. J. Chen, M. W. Chang, and C. J. Lin, “Load forecasting using support vector machines: A study on EUNITE competition 2001,” Department of Computer Science and Information Engineering, National Taiwan University, Tech. Rep., 2002.

[3] A. Jain and B. Satish, “Clustering based short term load forecasting using support vector machines,” in *PowerTech Conference*, Bucharest, Romania, July 2009.

[4] P. Bunnoon, K. Chalermyanont, and C. Limsakul, “A computing model of artificial intelligent approaches to mid-term load forecasting: a state-of-the-art survey for the researcher,” *International Journal of Engineering Technology*, vol. 2, no. 1, pp. 94–101, Feb. 2010.

[5] J. Hao, “Input selection using mutual information - applications to time series prediction,” Master’s thesis, Helsinki University of tehnology, Department of Computer Science and Engineering, 2005.

[6] A. Lendasse, V. Wertz, G. Simon, and M. Verleysen, “Fast bootstrap applied to LS-SVM for long term prediction of time series,” in *International Joint Conference on Neural Networks*, Budapest, Hungary, July 2004, pp. 705–710.

[7] A. Sorjamaa, J. Hao, N. Reyhani, Y. Ji, and A. Lendasse, “Methodology for long-term prediction of time series,” *Neurocomputing*, vol. 70, pp. 2861–2869, 2007.

[8] V. Vapnik, *Statistical learning theory*. Wiley, 1998.

[9] D. Basak, S. Pal, and D. C. Patranabis, “Support vector regression,” *Neural Information Processing*, vol. 11, no. 10, pp. 203–224, Oct. 2010.

[10] N. Turker and F. Gunes, “A competitive approach to neural divice modeling: Support vector machines,” in *International Conference on Artificial Neural Networks*. Athens, Greece: Springer, Sept. 2006, pp. 974–981.

[11] V. Cherkassky and Y. Ma, “Practical selection of svm parameters and noise estimation for svm regression,” *Neural Networks*, vol. 17, pp. 113–126, 2004.

[12] A. J. Smola and B. Scholkopf, “A tutorial on support vector regression,” *Statistics and Computing*, vol. 14, pp. 199–222, 2004.

[13] S. F. Crone and S. Pietsch, “A naive support vector regression benchmark for the NN3 forecasting competition,” in *International Joint Conference on Neural Networks*, Orlando, Florida, USA, Aug. 2007.

[14] C. C. Chang and C. J. Lin, *LibSVM: a library for support vector machines*, National Science Council of Taiwan, 2005.

[15] M. W. Chang, B. J. Chen, and C. J. Lin, “EUNITE network competition: Electricity load forecasting,” Department of Computer Science and Information Engineering, National Taiwan University, Tech. Rep., 2002. [Online]. Available: http://neuron.tuke.sk/competition/index.php

[16] S. Ruping, “SVM kernels for time series analysis,” in *LLWA 01*, Dortmund, Germany, Oct. 2001, pp. 43–50.

[17] C. W. Hsu, C. C. Chang, and C. J. Lin, *A Practical Guide to Support Vector Classification*, National Taiwan University, Department of Computer Science, 2003.

[18] M. M. Merino and J. Roman, “Electricity load forecasting using self organizing maps,” in *International Conference on Artificial Neural Networks*. Athens, Greece: Springer, Sept. 2006, pp. 709–716.