Load Forecasting Using Support Vector Machines: A Study on EUNITE Competition 2001

Bo-Juen Chen, Ming-Wei Chang, and Chih-Jen Lin

Department of Computer Science and
Information Engineering
National Taiwan University
Taipei 106, Taiwan (cjlin@csie.ntu.edu.tw)

Abstract—Load forecasting is usually made by constructing models on relative information, such as climate and previous load demand data. In 2001 EUNITE network organized a competition aiming at mid-term load forecasting (predicting daily maximum load of the next 31 days). During the competition we proposed a support vector machine (SVM) model, which was the winning entry, to solve the problem. In this paper, we discuss in detail how SVM, a new learning technique, is successfully applied to load forecasting. In addition, motivated by the competition results and the approaches by other participants, more experiments and deeper analyses are conducted and presented here. Some important conclusions from the results are that temperature (or other types of climate information) might not be useful in such a mid-term load forecasting problem and that the introduction of time-series concept may improve the forecasting.

Index Terms— Load forecasting, Regression, Support vector machines, Time series.

I. INTRODUCTION

Electricity load forecasting has always been an important issue in the power industry. Load forecasting is usually made by constructing models on relative information, such as climate and previous load demand data. Such forecast is usually aimed at short-term prediction (*e.g.* [10], [18], [12] and references therein), like one-day ahead prediction, since longer period prediction (mid-term or long term) may not be reliant due to error propagation. Mid-term and long-term prediction on load demand, however, may still be very useful in some situations.

In 2001, EUNITE* network organized a competition on the similar problem. The goal is to predict daily load demand of a month. Given information includes the past two-year load demand data, the previous four-year daily temperature and the local holiday events. Dealing with such a mid-term forecast problem, reliant prediction and error propagation are both competitors' concern. During the competition, we proposed a model, which was the winning entry, to solve the problem. Though the competition has been closed, we find this topic interesting and useful. Moreover, we would like to figure out the performance of mid-term load forecasting with such limited information. In this paper, therefore, we present our approach and discuss more on this problem. The main technique used in our solution is support vector machine (SVM) [8], a new machine learning method. To the best of our knowledge, this is the first work to successfully apply SVM on load forecasting. Some important conclusions from our experiments are that, temperature should not be considered and the time-series modeling scheme

is better. For the prediction of a 30-day period during which the temperature does not vary much, trying to predict the temperature and incorporate it into the model is not useful.

This paper is organized as follows. In Section II, we describe the goal of the competition task and the data provided. In addition, the analysis of data is also presented. Section III demonstrates the techniques we employed. Experiments and results of different models are showed in Section IV. Finally, the conclusion of our research and the comparison with other competitors are in Section V.

II. DATA AND TASK DESCRIPTION

A. Competition Task Description

The organizer of the EUNITE load competition provides competitors the following data:

- Electricity load demand recorded every half hour, from 1997 to 1998.
- Average daily temperature, from 1995 to 1998.
- Dates of holidays, from 1997 to 1999.

The task of competitors is to supply the prediction of maximum daily values of electrical loads for January 1999. Evaluation of submissions would mainly depend on the error metric of the results:

$$\text{MAPE} = 100 \frac{\sum_{i=1}^{n} \left| \frac{L_i - \hat{L}_i}{L_i} \right|}{n}, n = 31, \quad (1)$$

where L_i and $\hat{L}_i$ are the real and the predicted value of maximum daily electrical load on the ith day of the year 1999 respectively, and n is the number of days in January 1999. The goal of the competition is to forecast electrical load with minimum MAPE. [†]

B. Data Analysis

Before delving further into the solution we proposed, some observations about the data are examined first. Like many other literatures working on load forecasting, some relations between load demand and other information, such as climate or local events, are also investigated. The following observations are somehow specific to the competition, yet they can be applied to general load forecasting.

1) Properties of Load Demand: Load demand data given are half-hour recorded. Figure 1 gives a simple description of the maximum daily load demand from 1997 to 1998. By simple analysis, one can easily observe some properties of the load demand. First, the demand has some seasonal patterns: high demand for electricity in the winter while low demand in the summer. This pattern implies the relation between electricity usage and weather conditions in different seasons.

Moreover, if scrutinizing the data further, another load pattern could also be observed: a load periodicity exists in every

[†]Originally in the competition there are two error metrics used. One is MAPE, and the other one, finally not really used, is the "maximal error":

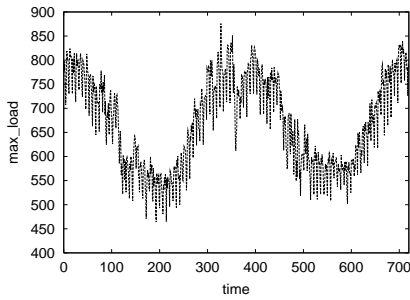$$\text{Maximal error} = \max |L_i - \hat{L}_i|, i = 1, \dots, 31$$

Fig. 1.   Maximum Daily Load

week. Load demand in weekend is usually lower than that of weekdays (Monday through Friday). In addition, electricity demand on Saturday is a little higher than that on Sunday.

Further detailed examination of load data, such as daily patterns, can also be found since the dataset contains more details (half-hour recorded). However, since the aim of the competition is maximum values of daily load demand, this paper would, without loss of generosity, focus on the maximum values only.

2) Climate Influence: In load forecasting, climate conditions have always played an important role. Previous works on short-term load forecasting [10], [18], [12] also indicate the relation between climate and load demand. Climate conditions considered may include temperature, humidity, illumination, and some special events like typhoon or sleet occurrences. These considerations may be regarded on different levels due to different localities. However, while forecasting load demand, more climate information usually give predictions more confidence.

Taking our data for example, as we mentioned earlier, the load data have some seasonal variation, which indicates a great influence of climate conditions. A negative correlation between load demand and daily temperature can be easily observed in Figure 2. The correlation coefficient between the maximum load and the temperature is -0.868. In our dataset, it is clear to see that because of the heating use, higher temperature causes lower demands. Yet, unfortunately the only climate information provided in the competition is daily temperature. Such information limitation also affects solutions to such a problem.

There is another interesting observation: the temperature at December 31st, 1998 is the lowest from 1997 to 1998. This might imply the high uncertainty of the temperature and load of the incoming January 1999, and thus increase the difficulty of the load prediction.
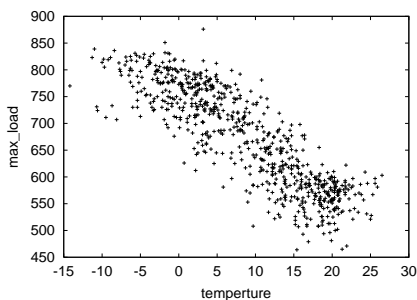

Fig. 2.   Correlation between the maximum load and the temperature

3) Holiday Effects: Local events, including holidays and festivities, also affect the load demand. These events may lead to higher demand for extra usage of electricity, or otherwise. Influences of these events are usually local and highly depend on the customs of the area. From the two-year load data, it is easy to find out that the load usually lowers down on holidays. With further scrutiny, the load also depends on what holiday it is. On some major holidays such as Christmas or New Year, the demand for electricity may be affected more compared with other holidays.
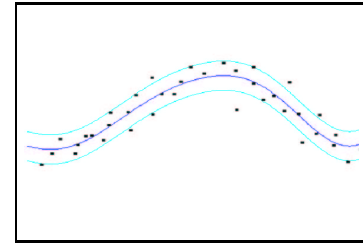
III. METHODS

Fig. 3.   Support Vector Regression

Support vector machine (SVM) is a new and promising technique for data classification and regression [23]. In this section we briefly introduce support vector regression (SVR) which can be used for time series prediction. Given training data (x_1, y_1), ..., (x_l, y_l), where x_i are input vectors and y_i are the associated output value of x_i, the support vector regression solves an optimization problem:

$$\min_{w,b,\xi,\xi^*} \quad \frac{1}{2}w^T w + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \qquad (2)$$

$$\text{subject to} \quad y_i - (w^T\phi(x_i) + b) \le \epsilon + \xi_i,$$
$$(w^T\phi(x_i) + b) - y_i \le \epsilon + \xi_i^*,$$
$$\xi_i, \xi_i^* \ge 0, i = 1, \ldots, l,$$

where x_i is mapped to a higher dimensional space by the function ϕ, ξ_i is the upper training error (ξ_i^* is the lower) subject to the ϵ-insensitive tube $|y - (w^T\phi(x) + b)| \le \epsilon$. The parameters which control the regression quality are the cost of error C, the width of the tube ϵ, and the mapping function ϕ.

The constraints of (2) imply that we would like to put most data x_i in the tube $|y - (w^T\phi(x) + b)| \le \epsilon$. This can be clearly seen from Figure 3. If x_i is not in the tube, there is an error ξ_i or ξ_i^* which we would like to minimize in the objective function. SVR avoids underfitting and overfitting the training data by minimizing the training error $C\sum_{i=1}^{l}(\xi_i + \xi_i^*)$ as well as the regularization term $\frac{1}{2}w^T w$. For traditional least-square regression ϵ is always zero and data are not mapped into higher dimensional spaces. Hence SVR is a more general and flexible treatment on regression problems. For experiments in this paper, we use the software LIBSVM [6], which is a library for support vector machines.

IV. Experiments

Though the competition has been closed, the problem still presents an interesting issue in load forecasting. In order to study mid-term load forecasting more, we conducted some experiments on the competition problem. In this section, these experiments and results would be described.

A. Data Preparation

In Section III, we have described the SVM technique. We next need to prepare datasets to build SVM models. While preparing the datasets, we need to encode useful information into the data entries (*i.e.* x_i in (2)). Also, different data encodings affect the selection of modeling schemes. Here we will discuss these issues in detail.

1) Feature selection: Each component of the training data is called a feature (attribute). Here, we consider what kind of information should be included. Assuming that y_i is the load of the ith day, in general we incorporate information at the same day or earlier as features of x_i. There are a few choices for the feature:

Basic information: calendar attributes. In Section II-B, we discuss the weekly periodicity of the load demand. Also, as we pointed out earlier, the load demand on holidays is lower than that on non-holidays. Therefore, encoding these information (weekdays and holidays) in the training entries might be useful to model the problem. Actually, among literatures of load forecasting, many works (*e.g.* [12] and [21]) have used the calendar information (time, dates or holidays) to model the load forecasting problem.

Temperature. Another possible feature is the temperature data. This is quite a straightforward choice, since load demand and temperature have a causal relation in between. In most short-term load forecasting (STLF) works, meteorological information which includes temperature, wind speed, sky cover and etc., has also been used to predict the load demand. However, to include the temperature in the training entries, there is one difficulty: in this competition, the real temperature data of January 1999 are not provided. In other words, for such mid-term load forecasting, temperature several weeks away are generally not available by weather forecasting. If we want to encode the temperature in our training entries, we will also need to predict or estimate the temperature of January 1999. Yet, temperature forecast is not easy, especially with such limited data. The use of temperature, therefore, would be a dilemma.

Time series style or not. Besides the weekdays, holidays and temperature, there is another information we consider to encode as the attributes: the past load demand. That is to introduce the concept of time-series into our models. To be more precise, if y_i is the target value for prediction, the vector x_i includes several previous target values $y_{i-1}, \ldots, y_{i-\Delta}$ as attributes. In the training phase all y_i are known but for future prediction, $y_{i-1}, \ldots, y_{i-\Delta}$ can be values from previous predictions. For example, after obtaining an approximate load of January 1, 1999, if $\Delta = 7$, it is used with loads of December 26-31, 1998 for predicting that of January 2. We continue this way until finding an approximate load of January 31. An earlier example using SVR for time series prediction is [17]. As

we know, the past load demand could affect and imply the future load demand. Therefore, considering to include such an information in the models might help forecast the load demand. In fact, in some load forecasting works, time series models have been explored ([1], [18]).

2) Data segmentation: Besides the features choices, Section II-B also shows the seasonal pattern for load demand. Previous works [10], [18], [21] on load forecasting also propose models built on data of different seasons. This inspires us to do some analyses for the data segmentation.

Usually people model time-series data by using the formulation,

$$y_t = f(x_t), \forall t = \Delta, \ldots, l,$$

where $x_t = (y_{t-1}, y_{t-2}, \ldots y_{t-\Delta})$, and Δ is the embedding dimension. However, this formulation is not suitable for non-stationary time series, because the characteristic of the time series may change with time. For such time series which alternate in time, we can consider a mixture model where

$$y_t = f_{i(t)}(x_t), \forall t = \Delta, \ldots, l.$$

Note that the formulation allows different characteristic functions in different time. Given the series $y_t, t = 1, \ldots, l$, the task is to identify all $i(t)$. We call this unsupervised segmentation where earlier work can be found in, for example, [19]. In other words, the method breaks the series into different segments where points in the same segment can be modeled by the same f_i. Recently, [7] states a similar framework using SVR with different parameters tuning. At any time point t, these methods consider different weights representing the probability that y_t belongs to corresponding functions. The sum of weight at any given time point t is always fixed to one. The weights are iteratively updated until one weight is close to one but others are close to zero. That means eventually y_t is associated to one particular time series.
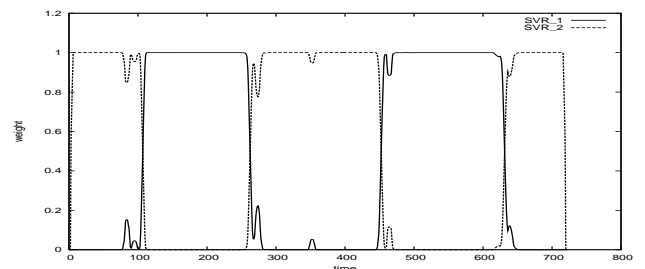

Fig. 4. Unsupervise segmentation for EUNITE data

Now, we can consider the loads as a time series. First we linearly scale all load data to $[0, 1]$. Then we can get time series style data by incorporating load of last seven days and weekday information to attributes. After that we follow the framework of [7] to analyze the data. We consider two possible time series so at each time point there are two weights. The experimental result is in Figure 4. The x-axis indicates days from January 1997 to December 1998 and the y-axis indicates the weights of two time series. Interestingly, "winter" and "summer" data are automatic separated without any seasonal information. The figure shows that the loads in the summer and in the winter have different characteristics.

Unsupervised data segmentation has been very useful for time series prediction (*e.g.* [19]). If the training data are associated with different time series, it is better to consider only data segments related to the same series of the last segment. Now the objective of the competition is to predict the load demand of January 1999, so we consider to use only the winter segment for training. Here, we choose January to March, and October to December to be our "winter" period, as the analytical result indicates. That is, the "winter" dataset would contain half of the data in 1997 and 1998. Also, we further extract data of January and February to form another possible training dataset. This dataset is much smaller than the "winter" one, and it would focus more on the load pattern in the period of our target concern.

3) Data Representation: After selecting useful information and proper data segments for encoding, we can prepare several combinations of training datasets. In these datasets, we encode a training entry (*i.e.* x_i in (2)) for the particular ith day, as follows:

(calendar, temperature (optional), past load (optional))

Here we use seven binaries to encode calendar information which includes weekdays, weekends and holidays, where six are for weekdays and weekends, and the other one for holidays. The six binaries stand for Monday to Saturday respectively and Sunday is represented as all six attributes are set to zero. Also, one numerical attribute is used for normalized temperature data, if the temperature is encoded. As for the past load, if encoded, we use seven numerics for the past seven daily maximum loads. The reason for using "seven" instead of other numbers is the complexity of model selection. We will elaborate more on this later. And finally, for such an entry, its target value (y_i in (2)) is assigned to the maximum load demand of the ith day.

Yet, for the models built with temperature information, the lack of temperature of January 1999 will pose a problem in the predicting phase. As we mentioned before, this may lead us to the prediction or estimation of the temperature in 1999 and that will be a more difficult problem, especially with only daily temperature of the past four years (1995-1998). A straightforward idea is to use the average of the past temperature data for the estimation. That is, the temperature of each day in January 1999 is estimated by averaging the past January daily temperature data (from 1995 to 1998). Competitors like [9], [5] and [13] used the averaged temperature in their models. Also [14] uses temperature of the other cities close to Slovakia for the estimation. In their report, the temperature for January 1999 was calculated through a linear combination of that of the other three cities. This is a kind of cheating as we are supposed to do the prediction at December 31, 1998 so temperature information of January, 1999 at any place is not available. For experiments here, these two estimations for temperature will be employed in the predicting phase, provided the temperature is encoded in our training datasets.

There is one remark about the testing data of January 1999. In this period, there are two holidays, January 1 and 6. However, in the data entries prepared for the prediction, we remove the holiday flag of these two entries. In other words, we treat them as non-holidays. The reason is that our models cannot learn well about the load demand on holidays. The number

TABLE IV.1

MAPE USING DIFFERENT DATA PREPARATION

(seg., TS style)	W/out T	W/ avg. T	W/ 3c T	W/ real T
(Winter, yes)	1.95%	3.14%	2.71%	2.70%
(Jan-Feb, yes)	2.54%	3.21%	2.93%	2.96%
(Winter, no)	2.86%	3.48%	2.97%	3.10%

of holidays in the training data are too few to provide enough information. Moreover, for the time-series-based approach, inaccurate prediction at one day could affect the succeeding forecasting.

B. Implementation and Results

With different schemes of model construction, a series of experiments are conducted. MAPEs of different model schemes are the main concern of our comparison.

Upon the datasets we prepare, SVM models are built for load forecasting. When training an SVM model, there are some parameters to choose. They would influence the performance of an SVM model. Therefore, in order to get a "good" model, these parameters need to be selected properly. Some important ones are

1) cost of error C,
2) the width of the ϵ-insensitive tube,
3) the mapping function ϕ, and
4) load of how many previous days included for one training data.

In our experiments, as we mentioned earlier, for each training data we simply include the maximum load of the previous seven days. In addition, we consider only the Radial Basis Function (RBF) function, which is one of the most commonly used mapping functions. The RBF function has the property that $\phi(x_i)^T \phi(x_j) = e^{-\gamma \|x_i - x_j\|^2}$. Note that γ is a parameter associated with the RBF function which has to be tuned. Also, we fix $\epsilon = 0.5$ which is the default of LIBSVM [6]. Thus, parameters left are C and γ. Choosing parameters can be time consuming so in practice we decide some of them by using knowledge or simply guessing. Then, the search space is reduced.

As searching for the proper parameters, we need to access the performance of models. With their performance, then the suitable parameters are chosen. To do this, usually we divide the training data into two sets. One of them is used to train a model while the other, called the validation set, is used for evaluating the model. According to their performance on the validation set, we try to infer the proper values of C and γ. Here, due to the different characteristics of the data encoding schemes, we employ two procedures for the validation.

For time-series-based approaches, we respectively extract the data entries of January 1997 and 1998 to form the validation set and evaluate the models on them. The performance is decided by averaging the errors of these two validations. As for the non-time-series models, we simply conduct 10-fold cross validation to infer the parameters. That is, we randomly divide the training sets into 10 sets. Using each set as a validation set, we then train

a model on the rest. The performance of a model would be the average of the 10 validating predictions.

With this procedure, proper C and γ are selected to build a model for the future prediction. We then evaluate its performance by forecasting the load demand for January 1999.

Now we are ready to present experimental results. Table IV.1 shows the prediction errors generated by different data encodings and segmentations. In the table, the first column shows the data segments used and if the past load demand is encoded. Then the next four columns indicate the predictions with or without the temperature (T): "avg. T" for average temperature, "3c T" for the estimation derived from the three other cities' data, and "real T" for the real temperature of the January, 1999.

1) Time-series Models with Winter Data: In Table IV.1, it can be observed that using the "winter" data along with the past loads (time-series information), the model built without temperature outperforms all others. In fact, the winning entry in the competition is generated by such a model [‡]. The MAPE using the temperature of three other cities is smaller than that using the average temperature. Moreover, after the competition is closed, we get the real temperature data of January 1999. We employ the temperature-incorporated models on datasets with the real temperature. The result is shown in the last column of Table IV.1. Its performance, compared with those two using different temperature estimations, is not better. That means, even assuming the real temperature is known, the forecasting result is still not satisfactory.

2) Time-series Models with Jan-Feb Data: With the training set containing only data of January and February, the model built without temperature also performs better than all others built with temperature. This is the same as using the "winter" data segment. Again, the prediction error with temperature estimation coming from the other three cities' data is smaller than that with the other estimation or even that with the real temperature.

3) Non-time-series Models with Winter Data: Besides time-series-based approaches, we build models without taking the past load demand as attributes. The training set of these models is limited to the "winter" segment. In Table IV.1, we show the test errors of the forecast with or without temperature information. Just like the aforementioned experimental results, the MAPE of prediction on the dataset without temperature is better than that with. Actually as one can see in Figure 8, the forecast generated by the model using only calendar attributes is the same weekly. That is because besides the calendar attributes, the dataset does not provide any other information for the model. Using the temperature estimation derived from linear combination of the three cities' temperature data is again better than using the real temperature or the average estimation.

4) Remarks: From the results, an interesting observation can be made: models built without the temperature generally perform better than those built with. This renders the issue about the usage of temperature information. As we mentioned earlier, temperature information is important for load forecasting. However, models constructed with such information may

be really sensitive to the temperature and thus the estimation for the temperature in 1999 would surely affect the performance of the models.

In Figure 5, the real temperature data and the two estimations are shown. Figures 6-8 plot the predicted values as well as the real load demand of January 1999. Observing these figures, we can find that the higher the temperature is, indeed, the lower the demand is. That is, the models we build do catch the causal relation between the load demand and the temperature. Nevertheless, using real temperature cannot predict the load demand more precisely than using the estimation coming from the data of the three other cities. This result somehow implies the inappropriateness of incorporation of the temperature. In fact, as we compute the correlation coefficient between the maximum load and the temperature for each month instead of the whole two year, we find they are variant (ranging from -0.64 to 0.32). This also indicates the fuzzy correlation between the load and the temperature in a shorter period.
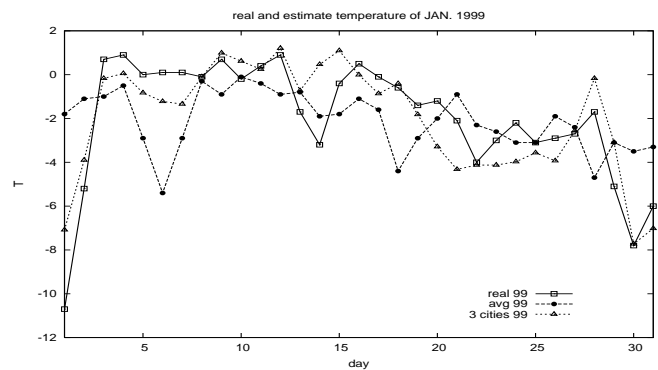

Fig. 5.   Estimates and real temperature in Jan. 1999

Models built on data segments of January and February perform not as well as those built on the "winter" segments. We think the main reason is that, with the limited data given (only two years), such data segments cannot provide enough information for models compared with the "winter" segments that contains more entries. Therefore, though dataset containing data only from January and February may represent the period of the prediction better, due to the limited data, models built on such segments are somehow not as competitive as those built with more information.

For comparison, models without time-series information are also built. The performance of such models is less competitive. This shows that models built without the past loads may not be able to learn the tendency of the load demand. Moreover, if the temperature is used to built the model, it would be the main attribute that affects the predicted values. Due to the limited data and the fuzzy correlation between the temperature and the load demand, such incorporation of temperature information could introduce higher variance into the model and result in unreliable prediction. That is why even using the real temperature, the error is higher than that of using only calendar information.

Recall that the temperature of last day in 1998 is the lowest of the whole dataset, but the load is not extremely high. Moreover, the correlation coefficient between the load and the temperature in December 1998 is 0.092. Therefore, we suspect the behavior

---

[‡] During the competition, our validation procedure is different from what we employ in this paper. Thus, the actual parameters that generate the winning entry is not the same as those we present here. However, the modeling scheme is the same.

of the load demand in January 1999 and in the end of December 1998 is abnormal. But despite the possible abnormality, with the time-series information, the model could at least catch the load tendency, and thus give reasonable predictions following the trend of December 1998. That is why the load demand predicted by time-series-based models is generally lower than that predicted by non-time-series-based ones.
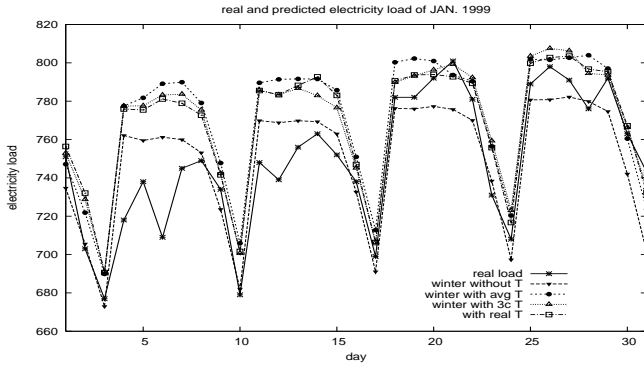

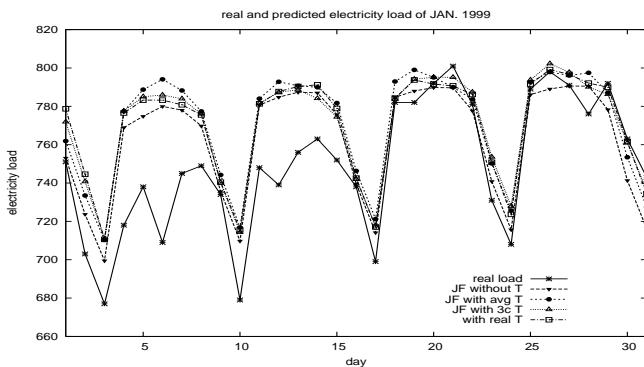Fig. 6. Estimates and real electricity load in Jan. 1999, with "winter" data

Fig. 7. Estimates and real electricity load in Jan. 1999, with January-February data

V. DISCUSSIONS AND CONCLUSIONS

A. Comparisons

In the competition, many groups have proposed different methods to solve the problem. Model schemes also differ among them.

First, the data segments used for training are different among the competitors. In our method, we select half year, *i.e.* the winter segment, as our training base. Some competitors like [9], [15], [20] and [4], also select particular segments rather than the whole two-year data. [9] even uses only January and February as such a choice could focus on the target period. Comparing all these competition entries, we find the selection of data segments may improve performance of models, since the load demand appears to have different distribution in different seasons. On the other hand, including the whole two-year data, though with more information, could generate noise for the trained models.

As for choosing the features for models, the causal relation between temperature and load demand is most used. Many
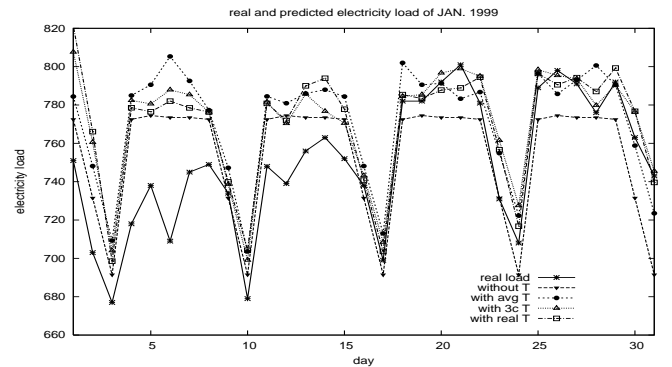

Fig. 8. Estimates and real electricity load in Jan. 1999, with non-time-series model

competitors use the temperature, along with some other information, to build causal models while some also incorporate with time-series analysis (*e.g.* [20], [22] and [3]). For the causal models, most competitors use neural networks ([15], [5], [25] and [13]). Others use adaptive logic network ([9]), fuzzy rule-based models ([24]), self-organizing maps combined with ARIMA models ([11]), clustering ([2]), and etc. However, the use of causal relation between temperature and load demand, although straightforward, might need further consideration. Because, as we pointed out earlier, the temperature of January 1999 is not given during the competition, adopting temperature information in models would need future weather forecast.

In fact, some competitors do try to predict or infer the weather condition of January 1999 ([16], [15] and [14]), yet, with the limited data, forecasting the temperature may even be a more difficult problem. Without good forecast for the temperature, the model might end up with imprecise prediction, even though it may perform well in the training phase,

Holiday is another issue. For most competitors, load forecasting on holidays is not easy to handle. The difficulty is that the historical load data on holidays is not enough, and thus make the behavior of load demand on these special days hard to model. In our works, we do include holiday information in the training phase. However, while predicting the load in 1999, we remove the holiday flags on January 1 and 6. Other competitors treat some holidays as Saturday or Sunday ([15], [9]), build another model for holidays ([5]), or manually adjust the predicted result ([20]).

B. Conclusion

Among these different proposed solutions, we find that choosing appropriate data segments seems to enhance the model performance. Furthermore, models built with temperature information require the estimation of the future temperature, which is very difficult and may lead to inaccurate prediction. In other words, including unprecise information causes higher variance on prediction so a conservative approach using only available correct information is recommended for such mid-term load forecasting. As the experimental results show, the introduction of the time-series attributes also gives models better information to forecast load demand more precisely. Actually, these considerations are what we were upon to propose

the time-series based, winter-data only, without temperature information, model which generated the winning entry.

ACKNOWLEDGMENTS

We appreciate that EUNITE organized such an interesting competition which inspire us to conduct research on load forecasting.

REFERENCES

[1] N. Amjady. Short-term hourly load forecasting using time-series modeling with peak load estimation capability. *IEEE Transactions on Power Systems*, 16(3):498–505, August 2001.

[2] Z. Boger. Electricity load forecast using artificial neural networks clustering. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.

[3] G. Bontempi. Eunite world-wide competition: Research report. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.

[4] W. Brockmann and S. Kuthe. Different models to forecast electricity loads. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.

[5] E. Castillo, B. Guijarro, and A. Alonso. Electricity load forecast using functional networks. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.

[6] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[7] M.-W. Chang, C.-J. Lin, and R. C. Weng. Analysis of switching dynamics with competing support vector machines. In *Proceedings of IJCNN*, pages 2387–2392, 2002.

[8] C. Cortes and V. Vapnik. Support-vector network. *Machine Learning*, 20:273–297, 1995.

[9] D. Esp. Adaptive logic networks for east slovakian electrical load forecasting. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.

[10] H. S. Hippert, C. E. Pedreira, and R. C. Souza. Neural networks for short-term load forecasting: a review and evaluation. *IEEE Transactions on Power Systems*, 16(1):44–55, 2001.

[11] G. A. Ivakhnenko. Inductive self-organising algorithm for maximum electrical load prediction. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.

[12] K. Jabbour, J. F. V. Riveros, D. Landsbergen, and W. Meyer. Alfa: Automated load forecasting assistant. *IEEE Transactions on Power Systems*, 3(3):908–914, August 1988.

[13] I. King and J. Tindle. Storage of half hourly electric metering data and forecasting with artificial neural network technology. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.

[14] W. Kowalczyk. Averaging and data enrichment: two approaches to electricity load forecasting. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.

[15] A. Lewandowski, F. Sandner, and P. Protzel. Prediction of electricity load by modeling the temperature dependencies. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.

[16] A. Lotfi. Application of learning fuzzy inference systems in electricity load forecast. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.

[17] K.-R. Müller, A. Smola, G. Rätsch, B. Schölkopf, J. Kohlmorgen, and V. Vapnik. Predicting time series with support vector machines. In B. Schölkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in Kernel Methods - Support Vector Learning*, pages 243–254, Cambridge, MA, 1999. MIT Press.

[18] P. Murto. Neural network models for short-term load forecasting. Master's thesis, Department of Engineering Physics and Mathematics, Helsinki University of Technology, 1998.

[19] K. Pawelzik, J. Kohlmorgen, and K.-R. Müller. Annealed competition of experts for a segmentation and classification of switching dynamics. *Neural Computation*, 8(2):340–356, 1996.

[20] E. Pelikán. Middle-term electrical load forecasting by time series decomposition. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.

[21] S. Rahman and O. Hazim. A generalized knowledge-based short-term load-forecasting technique. *IEEE Transactions on Power Systems*, 8(2):508–514, May 1993.

[22] F. Rivieccio. SVM for an electricity load forecast problem. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.

[23] V. Vapnik. *Statistical Learning Theory*. Wiley, New York, NY, 1998.

[24] R. C. Weizenegger. Maximum electricity load problem. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.

[25] D. Živčák. Electricity load forecasting using ANN. Report for EUNITE 2001 Competition, 2001. Available at `http://neuron.tuke.sk/competition/`.