

Hybrid Decomposition Time-Series Forecasting by DirRec Strategy: Electric Load Forecasting Using Machine-Learning

Branislav Vuksanovic*, Davoud Rahimi Ardali
University of Portsmouth, Portsmouth PO1 3DJ, UK.

* Corresponding author. Tel.: +44(0)2392842159; email: branislav.vuksanovic@port.ac.uk
Manuscript submitted September 15, 2018; accepted November 9, 2018.
doi: 10.17706/ijcee.2019.11.1.1-10

Abstract: The rise in availability of huge amounts of historical data and the need for accurate forecasting techniques of future behavior of electricity consumption emphasize the need for efficient techniques able to reliably estimate the stochastic dependency between the past and future observations in the grid. This study introduces a new time-series forecasting technique for performing short/medium-term electricity consumption forecasting with high accuracy, given a limited period of historical measurements available to extract trends present in time-series. Proposed technique can predict the future energy requirements without the need for additional information such as date or time of the measurements. Described hybrid method can overcome the performance drop issue, where there is redundant or missing data in historical measurements or when the historical measurements are noisy by utilizing three machine learning algorithms; Random Forests, Quinlan's M5 and Linear Regression. The operation of the proposed method is tested on different substations located in central London and the prediction performance established by comparing it to AutoRegressive Integrated Moving Average (ARIMA) and Autoregressive Neural Network (NNAR) time-series forecasting methods.

Key words: Time-series forecasting, load forecasting, DirRec strategy, multivariate machine-learning regression, power consumption prediction.

1. Introduction

The electricity industry and energy demand are growing at an extremely fast pace due to the development of human population and technological advances. Related technologies are being developed and implemented every day from smart grids to advanced metering infrastructures. Accurate forecasting and efficient management of the electrical energy are therefore becoming more important than ever.

Power generation and distribution industry has been rapidly growing and adopting smart technologies in the recent years. The power grid networks tend to use more renewable sources of energy which can nowadays be owned either by a main supplier or be placed in the smart industrial complexes themselves. The study [1] explains how the renewable energy sources (RES) differ from the old sources of energy in terms of supply and demand. In traditional grids, the supply will constantly match the load however, when the RES are available in the grid there will be instances when the supply could be significantly higher than demand. In these cases, the RES should be detached from the grid to prevent the collapse of the grid [2]. In April 2013 for example, in Germany 11.2 GW were forecasted from a photovoltaic (PV) sources for the next

day while actual PV generation was 11.2 GW. The grid supplied the additional 8.8GW which resulted in instability of the grid itself [3]. This illustrates the importance of forecasting the load demand in the grid accurately. Forecasting the load in a complex grid is an impossible task without the help of a smart computerized system. Implementing such systems in real time and performing the associated data processing is one of the main features of smart grid technology [4]. This usually requires artificial intelligence-based techniques to find trends and patterns in historical load consumption data.

1.1. Time-Series Forecasting

Time-series forecasting techniques have been influenced, from the 1960s on, by linear statistical methods such as ARIMA [5] models. ARIMA have been used in many different fields, including energy [5], economics [6], health [7] and tourism [8]. Since the time-series data in real world usually have non-linear characteristics [9], using ARIMA is not optimal for most of the real-world problems [10]. More recently, machine learning models have drawn attention and have established themselves as serious contenders to classical statistical models in the forecasting community. Artificial neural networks (ANN) have been studied and implemented in time-series forecasting applications by several researchers [11]-[15]. Data driven approaches such as ANNs and in general machine learning algorithms (MLAs) are suitable for many empirical datasets, in which there is no theoretical guidance to explain the data generation and its rules.

1.2. Machine Learning Approaches to One-Step Ahead Forecasting

The work by Atkeson *et al.* [16] suggests that given there is a historical data record (time-series T), supervised learning can be applied to one-step forecasting issue. Supervised learning in general performs modelling of a set of observations in regard to the relation between a set of inputs and outputs in those observations. In one-step forecasting the model M returns the next step value of the time-series at the time t by modelling the m previous values of the time-series $[T_0, \dots, T_{(t-1)}]$. This is illustrated in Fig. 1.

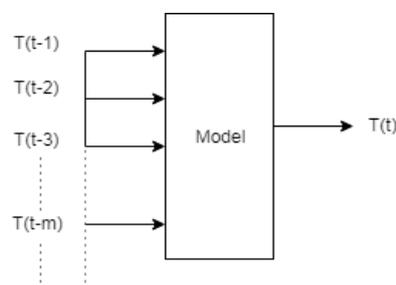


Fig. 1. One-step-ahead model created by supervised learning.

One-step-ahead forecasting requires prediction of the output value of such system given the historical observed values are available, hence, machine-learning is a required tool to perform this task. This paper focuses on local learning [17] techniques when it comes to machine learning. Local learning helps reducing the number of assumptions when modelling the data. It also enables the system to adopt online learning capabilities.

The most important feature of local learning is that it can model non-stationary data. This is beneficial to time-series forecasting since it can model both spatial and temporal dimensions of a time-series. Various machine learning strategies can be adopted to perform one-step-ahead forecasting, for instance utilisation of nearest neighbors and lazy learning techniques have been explored in some recent works [18].

1.3. Machine Learning Approaches to Multi-step-ahead Forecasting

Unlike the one-step-ahead approach, in multi-step forecasting the model predicts the next p values of the time-series $[T_t, \dots, T_{(t+p-1)}]$. There are five main strategies [18] for multi-step-ahead forecasting, Recursive,

Direct, DirRec, MIMO and DIRMO.

Recursive strategy is actually a one-step ahead model which repeats its forecasting recursively for multi-step-ahead forecasting [19]. The weakness of this strategy is that the error rate increases dramatically as the size of the points ahead increase particularly if the first point forecast is not predicted accurately. This strategy is also prone to accumulation errors. Despite these weaknesses, this strategy has been used successfully in real world problems by different MLA's such as artificial neural networks (ANN) [20] and nearest neighbours.

Direct strategy creates p separate models for p points ahead. Since the models are built independently the dependencies between the predictions are not considered by this method. Another drawback of this strategy is that it requires more computational power to create multiple models specially as p increases [21]. This method has been used for different forecasting tasks by ANNs [22] and decision trees [23].

By combining the Direct and Recursive methods, a new strategy called DirRec was introduced [19]. This strategy creates separate models for r ahead points ($r < p$) and enlarges the inputs set by adding the r forecasted points to the training set. It repeats this process recursively to achieve the desired number of points ahead (p). This reduces the functional complexity and computational power requirements of forecast as well as learning the statistical dependencies of every subset in order to use that for more accurate prediction.

All strategies explained above are consisting of a multi-input single-output core technique, however, by adopting multi output techniques it is possible to perform a multi-input multi output (MIMO) strategy, which will consider the full stochastic dependencies between the future points. Despite the benefits of this technique, [24] explains that this strategy can reduce the flexibility of forecasting. Additionally, there are only a few machine learning algorithms that can perform multi-output regression, which is another limitation for this approach.

In order to combine the benefits of MIMO and DirRec, these two strategies can be combined to create DIRMO [24]. Here, multiple partitions of the final p forecast size are created first, followed by MIMO forecasting on each partition. The size of each block in this strategy determines the level of predictor output flexibility and the degree of stochastic dependency.

2. Proposed Hybrid Method

2.1. The Dataset

In this work, a dataset of power consumption measurements on substations located in central London has been analyzed and used to demonstrate the effectiveness of our approach. Full dataset consists of 77 substations with measurements taken every 1 hour over the period from January 2010 to January 2012. The frequency of the measurements taken in each substation is 24 (one per hour).

The measured load data is seasonal with daily patterns; however, no obvious weekly or monthly patterns can be identified in the dataset. There are sudden fluctuations in the data which can be caused by failure of measurement equipment, grid reconfiguration by the network operator or even weather conditions; however, the proposed system does not require additional information about the date, network configuration or weather conditions for forecasting.

2.2. Predictor Variables Construction

The aim of this stage is to convert a one-dimensional time-series (one observation per time instant) containing L measurements into multi-dimensional set of observations by sampling and reshaping the time-series. If D new points (prediction size) for the measured time-series are to be predicted, series needs to be reshaped into $(2 \times D) \times (N \div 2)$ sampling matrix (SM) where obviously, the length of the time

series L must be such that $L = D \times N$. Rows of SM are formed by taking $2 \times D$ successive measurements from the start of the original time-series. In each row of the SM the first D points will act as the predictors and the second D points act as the target variables. The SM matrix formed in this way can now be split into a training and testing sets by considering all rows of the SM but one for training matrix and the last row for the testing set. The target for the testing set will be the D points in the future to be forecasted.

For a sample time-series containing measurements collected over Z days, extracting two days' worth of measurements into columns will result in the sampling matrix indicated in the Fig. 2. Furthermore, partitioning of the sampling matrix into training and testing sets is also indicated in this figure.

Each column on the right-hand side of the SM in Fig. 2, acts as the target variable for the machine learning algorithm while the left-hand side columns will be used as the predictor variables. A regression model will be fitted to the training set based on Direct strategy in order to predict the target variables in the testing set. By doing this the system is able to predict D new points in the future.

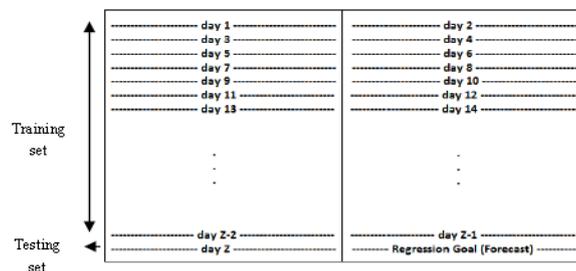


Fig. 2. Splitting time-series data into training and testing sets by creating a sample matrix.

2.3. Time-Series Decomposition and Hybrid Forecasting

Prior to training the system, the historical load time-series is decomposed by locally-weighted scatterplot smoothing (LOWESS) method. The decomposition by LOWESS provides the seasonal component, the trend and the remainder of the time-series which is most likely the noise. Each of the three components is forecasted separately by different machine learning algorithms (MLAs). The selected combination of MLAs forms the order of the system i.e. (Seasonal, Trend, Remainder) or (S, T, R). In this paper 3 MLA's have been selected for the main forecasting system and developed in R (R Core Team, 2014). Random Forests (RF), M5 from Quinlan [25] and linear regression (LR).

The final multi points ahead forecast for the time-series will be achieved by recomposing the time-series back from its three forecasted components. The orders of the system represent the MLAs chosen to forecast for each component of the time-series. LR, RF and M5 have been considered to be orders 1,2 and 3 accordingly.

2.4. Optimal Prediction Size

Term prediction size is used to denote the number of new points to predict. The significance of the prediction size comes from the fact that it is directly correlated to the size of the training and testing sets and in general the size of SM . As the prediction size increases, more points of the data will be entered as predictors and targets to the system. Thus, the MLA needs to find a link between a larger number of predictors and target values. This increases the chances of the MLA over-fitting a model which also considers the link between the noise in the data and the target variables. Moreover, the larger the target variables become fitting a multivariate regression model becomes harder and the results get less accurate. Smaller prediction sizes most likely result in an under-fitting model being created and a larger data set is generally required to properly train the system.

This emphasizes the significance of an optimal prediction size for every dataset before starting forecasting. To solve this a few prediction sizes have been used on one substation’s data to test the accuracy of the system. The mean absolute percentage error (MAPE) is calculated each time by comparing the actual consumption with the prediction. The general formula to calculate MAPE is shown in (1), where A is the actual value and F is the forecasted value:

$$MAPE = \frac{100}{N} \sum_{n=1}^N \left| \frac{A(n) - F(n)}{A(n)} \right| \quad (1)$$

Additionally, the root-mean-square error (RMSE) has been obtained and used as an additional performance measurement tool for each forecast. RMSE can be calculated using (2).

$$RMSE = 100 \sqrt{\frac{\sum_{n=1}^N (A(n) - F(n))^2}{N}} \quad (2)$$

Fig. 3 demonstrates the impact of prediction size on the error rates. As it can be realised from this figure, prediction sizes between 2 and 6 can achieve the lowest error rates however, the total points ahead forecast can be very limited by selecting these prediction sizes.

To find the most suitable prediction size for the time-series, a novel approach has been used by finding the Power Spectral Density (PSD) of the time-series.

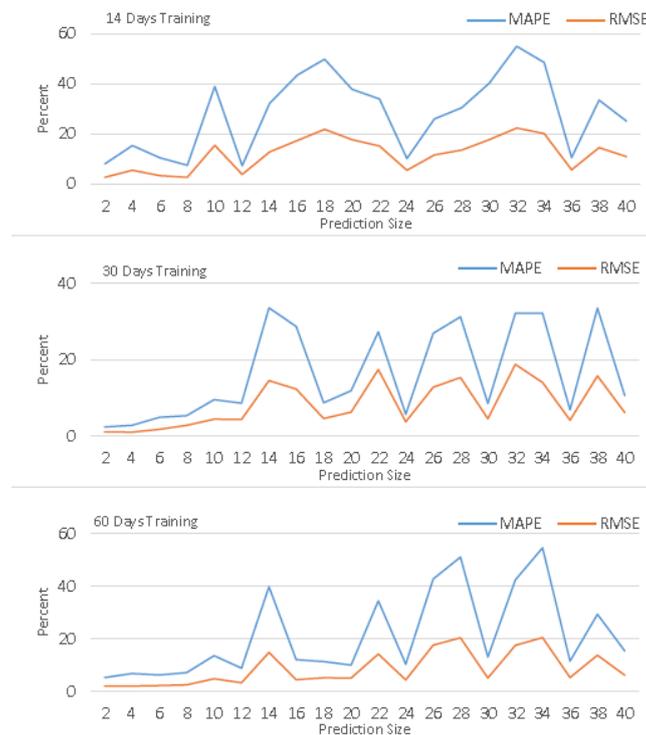


Fig. 3. Impact of the prediction size on MAPE and RMSE for different number of training days.

Following the simple frequency analysis it is established that the prediction sizes that were achieving the lowest forecast error rates in Fig. 3, correspond to dominant frequency and the second harmonic present in the measured data set. The dominant frequency of the time-series and 9 of its most significant harmonics are presented in Table 1.

Since analyzing the PSD of the time-series technique helps with finding the most accurate prediction rate, the optimal prediction size should be determined prior to forecasting for each dataset.

Table 1. Top 10 Dominant Frequencies of Substation 1

Rank	Frequency	Period	Power
1	0.0416666667	24.00000	122445.6796
2	0.0833333333	12.00000	45105.2598
3	0.0006944444	1440.00000	2824.8824
4	0.0013888889	720.00000	1820.9545
5	0.1250000000	8.00000	1669.5960
6	0.2083333333	4.80000	861.6264
7	0.0777777778	12.85714	827.8121
8	0.0361111111	27.69231	771.2810
9	0.0423611111	23.60656	719.2211
10	0.0041666667	240.00000	703.9741

2.5. Transformation into DirRec

To lift the limitation of the maximum number of the forecast points, recursive prediction strategy is added to the system architecture in this section. After finding the most optimal prediction size if the desired forecast points are higher than the prediction size, the system will divide the forecast points into blocks of prediction size length and repeat the forecast for each block by adding the initial predicted values to the original time series data. Moreover, the error rate increases by selecting shorter blocks as previously discussed in section 2.4. The final architecture of the Hybrid DirRec Time-series Forecasting (HDTF) system is provided in Fig. 4.

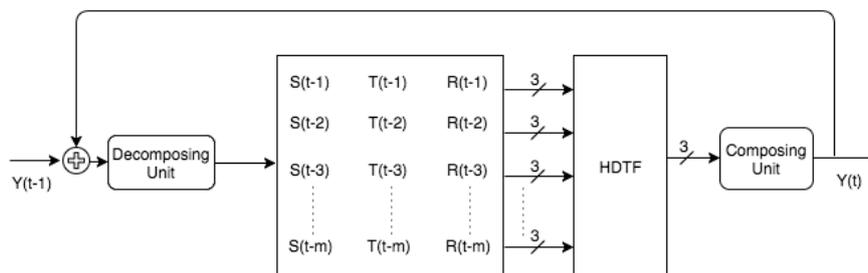


Fig. 4. Complete model for the HDTF system.

The performance of the proposed system has been measured by comparing the results of the three different strategies for 12 points to 96 points ahead forecasting. The comparison results indicate that the DirRec strategy outperforms Direct and Recursive strategies in all forecasting tasks except for one.

The reason for Direct method outperforming DirRec in 24 points ahead is that 24 points is the period corresponding to the dominant frequency for this dataset. On the other hand, in terms of processing speed, Direct strategy achieves the best results by performing the forecast faster specially in situations where the forecast size is larger. This was expected since recursive strategy involves repeating the forecast which makes the overall process slower in DirRec and even more so in Recursive strategy. Recursive strategy is the slowest and the least accurate method as already discussed in Section 1.

3. Results

In order to establish the performance of the HDTF method, two different experiments have been performed on the dataset. First, keeping a short prediction period constant, different amount of training data has been provided for the system. Second, for a fixed length of training data, length of prediction

horizon has been varied.

The experiments have been performed using the proposed method as well as two other standard techniques, Autoregressive Integrated Moving Average (ARIMA) [26] and Autoregressive Artificial Neural Networks (NNAR) which are available in R package forecast [27].

3.1. Short-Term Forecasting

For short term forecasting experiment, the data from 5 different substations are selected to perform the tests designed for 12-hour, 1-day and 2-day power consumption forecasting. The minimal training period for each test starts from 7 days and gradually extends to 31 days with one day interval. MAPE, RMSE and processing time are averaged over all 25 tests and the final results are provided in Tables 2, 3 and 4.

HDTF outperforms NNAR and ARIMA in most cases when predicting one day (24 points) and two days (48 points) ahead in terms of error rate. However, since HDTF's strategy changes to Direct while forecasting 12 hours ahead, there is a slight increase in error rates which results in NNAR performing better in some case (see Table 3). In terms of processing time, ARIMA is the fastest method with a large advantage compared to the machine learning methods. HDTF performs faster than NNAR in 12 hours ahead prediction but it is slower in general compared to the other two scenarios for these three short-term forecasting tasks.

Table 2. Twelve-Hour Prediction by Average of 7 to 31 Days Training

Method	ARIMA			HDTF			NNAR		
Tools	MAPE (%)	RMSE	TIME (Sec)	MAPE (%)	RMSE	TIME (Sec)	MAPE (%)	RMSE	TIME (Sec)
Sub1	7.91	3.57	0.16	7.70	3.56	1.64	8.20	3.73	2.88
Sub2	8.11	7.40	0.17	7.57	7.10	1.71	7.41	6.72	2.75
Sub3	7.74	8.29	0.18	7.74	8.28	1.62	6.18	6.69	2.45
Sub4	9.12	11.27	0.18	8.75	10.78	1.73	10.71	13.42	2.60
Sub5	26.15	12.26	0.16	20.72	11.03	1.69	19.39	9.44	2.68

Table 3. One-Day Prediction by Average of 7 to 31 Days Training

Method	ARIMA			HDTF			NNAR		
Tools	MAPE (%)	RMSE	TIME (Sec)	MAPE (%)	RMSE	TIME (Sec)	MAPE (%)	RMSE	TIME (Sec)
Sub1	7.47	4.03	0.12	7.09	3.80	2.61	8.23	4.48	2.84
Sub2	7.19	7.44	0.16	7.01	7.48	2.64	7.49	7.96	2.61
Sub3	6.94	9.39	0.16	7.87	10.98	2.66	6.86	9.82	2.46
Sub4	9.36	11.04	0.18	8.78	10.46	2.85	10.46	13.12	2.69
Sub5	20.13	9.86	0.15	17.09	8.95	2.75	20.32	9.42	2.71

Table 4. Two-Day Prediction by Average of 7 to 31 Days Training

Method	ARIMA			HDTF			NNAR		
Tools	MAPE (%)	RMSE	TIME (Sec)	MAPE (%)	RMSE	TIME (Sec)	MAPE (%)	RMSE	TIME (Sec)
Sub1	7.68	4.18	0.24	7.20	3.91	5.57	8.97	4.95	3.10
Sub2	7.54	7.97	0.25	7.24	7.80	5.34	8.21	9.10	2.76
Sub3	8.01	11.00	0.23	8.38	11.96	5.24	8.99	13.03	2.55
Sub4	10.97	13.24	0.23	9.61	11.75	5.53	10.99	13.94	2.71
Sub5	22.01	10.99	0.22	18.55	9.85	5.31	25.37	11.87	2.68

3.2. Long-Term Forecasting

In order to measure the performance of the methods over longer periods of time. Three medium to long term forecasting tasks have been performed on substations 1, 2 and 3. The training set for each substation is constant at 30 days. The forecast goal starts from 1 days ahead and increments up to 20 days ahead. The performance of HDTF is then compared to NNAR and ARIMA. The results indicate that HDTF outperforms

the traditional ARIMA and NNAR methods in medium to long-range forecast tasks as well. This is more obvious especially when the noise levels are high (i.e. substation 3) where statistical methods like ARIMA fail. The impact of the forecast days on MAPE can be observed for 3 substations in Fig. 5. This figure shows that HDTF performs very consistently for all substations however the performance of NNAR and ARIMA is data and noise dependent.

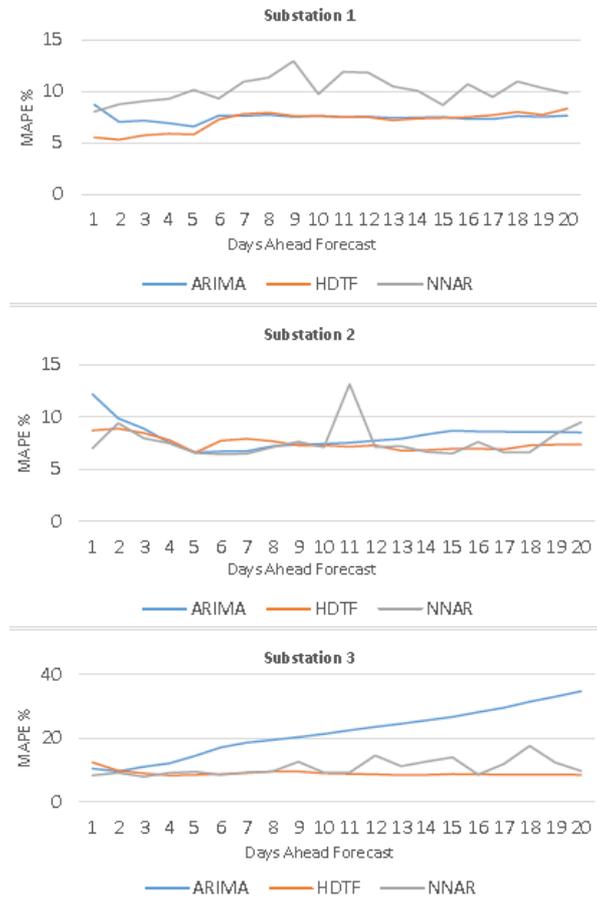


Fig. 5. The error-rate curves of ARIMA, HDTF and NNAR for forecasting 1 to 20 days ahead in three different substations.

4. Conclusions

Predicting the future load consumption is one of the most important tasks in power systems from generation to distribution, especially where the grids are complex.

This paper introduced a new hybrid method for short and medium range time-series forecasting based on DirRec strategy, called HDTF. Additionally, application of this method in demand forecasting in power stations was established without the need for extra information about the date or weather conditions. Finally, the performance of HDTF is compared to two well-known time-series modelling and forecasting techniques, statistical modelling technique, ARIMA and feed forward neural network-based technique with lagged inputs, NNAR.

The comparison results of short-term forecasting show that HDTF outperforms both other methods except for the cases where the forecast period is very short or less than its optimal “prediction size” i.e. one day in this work’s dataset. The reason for this limitation is that forecasting a period less than its optimal “prediction size” will result in the method switching to direct strategy which will result in faster processing times but lower accuracy as evidenced in Table 3.

The medium to long range forecast test results also demonstrate that HDTF not only outperforms the other two techniques, but also overcomes the noise present in the dataset. The reason for the good noise tolerance of the proposed method is that, the orders selected for the datasets include M5 and Random forest regression methods for two components of the complete time-series. Both these MLAs are tree based methods, designed to have great noise tolerance compared to statistical methods such as ARIMA. This was more obvious in the noisier dataset of substation 3 where ARIMA failed.

In comparison with MLA based forecasting systems, HDTF shows consistency in error rates over different forecast periods. This is an advantage compared to NNAR method, which shows sudden increases in error rates as the prediction size increases. All methods show a gradually increase in MAPE trend as the forecast points ahead increase except for substation 3 where ARIMA's MAPE increases faster than the machine learning based methods due to higher noise levels.

Although the comparison results of the HDTF show improved performance compared to other two methods, the impact of the different orders of the system on the short and long ranged forecasting needs to be further investigated. This will be the focus of the continuation of this work.

References

- [1] Teo, T. T., Logenthiran, T., & Woo, W. L. (2015). Forecasting of photovoltaic power using extreme learning machine. *Proceedings of the 2015 IEEE Innov Smart Grid Technol - Asia (ISGT ASIA)* (pp. 1-6).
- [2] Liserre, M., Sauter, T., & Hung, J. Y. (2010). Future energy systems: Integrating renewable energy sources into the smart power grid through industrial electronics. *IEEE Ind. Electron. Mag.*, 4, 18-37.
- [3] Stetz, T., Appen, J., Niedermeyer, F., Scheibner, G., Sikora, R., & Braun, M. (2015). Twilight of the grids: The impact of distributed solar on Germany's energy transition. *IEEE Power Energy Mag.*, 13, 50-61.
- [4] Markovic, D. S., Zivkovic, D., Branovic, I., Popovic, R., & Cvetkovic, D. (2013). Smart power grid and cloud computing. *Renew Sustain Energy Rev.*, 24, 566-577.
- [5] Erdogdu, E. (2007). Electricity demand analysis using cointegration and ARIMA modelling: A case study of Turkey. *Energy Policy*, 35, 1129-1146.
- [6] Khashei, M., Rafiei, F. M., & Bijari, M. (2013). Hybrid fuzzy auto-regressive integrated moving average (FARIMAH) model for forecasting the foreign exchange markets. *Int. J. Comput. Intell. Syst.*, 6, 954-68.
- [7] Yu, H.-K., Kim, N.-Y., Kim, S. S., Chu, C., & Kee, M.-K. (2013). Forecasting the number of human immunodeficiency virus infections in the Korean population using the autoregressive integrated moving average model. *Osong Public Heal Res. Perspect*, 4, 358-62.
- [8] Chu, F.-L. (2008). A fractionally integrated autoregressive moving average approach to forecasting tourism demand. *Tour Manag*, 29, 79-88.
- [9] Lee, Y.-S., & Tong, L.-I. (2012). Forecasting nonlinear time series of energy consumption using a hybrid dynamic model. *J. Apenergy*, 94.
- [10] Khashei, M., Bijari, M., & Raissi-Ardali, G. A. (2008). Improvement of auto-regressive integrated moving average models using fuzzy logic and Artificial Neural Networks (ANNs). *Neurocomputing*, 72, 956-967.
- [11] Bennett, C., Stewart, R. A., & Beal, C. D. (2012). ANN-based residential water end-use demand forecasting model. *Expert Syst. Appl.*, 40, 1014-1023.
- [12] Adebisi, A. A., Adewumi, A. O., & Ayo, C. K. (2014). Comparison of ARIMA and artificial neural networks models for stock price prediction. *J. Appl. Math.*
- [13] Geem, Z. W., & Roper, W. E. (2009). Energy demand estimation of South Korea using artificial neural network. *Energy Policy*, 37, 4049-4054.
- [14] Zhang, G., Eddy-Patuwo, B. Y., & Hu, M. (1998). Forecasting with artificial neural networks: The state of the art. *Int. J. Forecast*, 14, 35-62.

- [15] Zhang, G. P. & Qi, M. (2005). Neural network forecasting for seasonal and trend time series. *Eur. J. Oper. Res.*, 160, 501-514.
- [16] Atkeson, C. G, Moorey, A. W, Schaalz, S., Moore, A. W., & Schaal, S. (1997). Locally weighted learning. *Artif. Intell.*, 11, 11-73.
- [17] Hastie, T, Tibshirani, R., & Friedman J. (2009). *The Elements of Statistical Learning the Elements of Statistical Learning Data Mining, Inference, and Prediction* (2nd ed.).
- [18] Bontempi, G., Taieb, S., & Borgne, Y. L. (2014). Machine learning strategies for time series forecasting. *Bus Intell.*, 62-77.
- [19] Sorjamaa, A., & Lendasse, A. (2006). Time series prediction using DirRec strategy. *Neural Networks*, 143-148.
- [20] Saad, E. W., Prokhorov, D. V., & Wunsch, D. C. (1998). Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks. *IEEE Trans Neural Netw.*, 9, 1456-1470.
- [21] Tong, H. (2010). Threshold models in time series analysis - Some reflections. *J Econom*, 189, 485-491.
- [22] Kline, D. M. (2004). Methods for multi-step time series forecasting neural networks. *Neural Networks Bus. Forecast*, 226-250.
- [23] Tran, V. T., Yang, B. S., & Tan, A. C. C. (2009). Multi-step ahead direct prediction for the machine condition prognosis using regression trees and neuro-fuzzy systems. *Expert Syst. Appl.*, 36, 9378-9387.
- [24] Taieb, S. B., Bontempi, G., Sorjamaa, A., & Lendasse, A. (2009). Long-term prediction of time series by combining direct and MIMO strategies. *Proc. Int. Jt. Conf. Neural Networks, 2009*, 3054-3061.
- [25] Hyndman, R. J., & Khandakar, Y. (2008). Automatic time series forecasting: The forecast package for R. *J. Stat Softw*, 27, C3.
- [26] Quinlan, J. R. (1992). Learning with continuous classes. *Mach. Learn*, 92, 343-348.
- [27] Box, G. E. P., Jenkins, G. M., & Reinsel, G. C. (1994). *Time Series Analysis - Forecasting and Control*. I New Jersey: Prentice Hal.



Branislav Vuksanovic has graduated from the University of Belgrade, Serbia with degree in electrical and power engineering. He holds a MSc degree in measurement and instrumentation from South Bank University, London and a PhD in active noise control from the University of Huddersfield, UK.

Previously, he worked as a project engineer for Croatian Electricity Board in Osijek, Croatia. During his academic career he worked as a research fellow at Sheffield and Birmingham Universities and lectured at University of Derby. He has published papers in the field of active noise control, biomedical signal processing and pattern recognition for intrusion detection and knowledge-based authentication. He is now a senior lecturer at the University of Portsmouth.



Davoud Rahimi Ardali holds the BSc and MSc degrees in electronic engineering. He finished his MSc studies with a distinction and received the best student award from University of Portsmouth, UK. He is currently completing a PhD in the area of machine learning with specific application in mass personalization of skin-care products at the University of Portsmouth. His main research interests, apart from big data and machine learning are image and voice processing, robotics, and biomedical engineering.