Implementation of Neural Networks and Feature Selection for Short Term Load Forecast

Goran Veljanovski[1], Pande Popovski[1], Metodija Atanasovski[1] and Mitko Kostov[1]

Abstract – **Forecasting power demand is a significant factor in the planning and secure operation of power systems. Forecasting is based on considering all the factors that affect power demand. Reducing of the forecasting error can be achieved by identifying the factors that affect power demand, understanding their influence on the forecasting and adequate usage in the forecasting model. The selection of the optimal number and combination of features that will lead to a prediction model with less error is performed using the Recursive Feature Elimination with Cross-Validation (RFECV) elimination method. The features are selected according to a statistical importance measure. Then, the Artificial Neural Networks (ANN) are used to build load forecasting models, for a specified hour in future, and then, by combining those models, an one day-ahead model was created. The proposed approach combines the abilities of the neural networks to learn the nonlinear relationship between features and the optimization capability of the RFECV elimination method to find the best model for forecasting.**

Keywords – **Short term forecasting, artificial neural networks, Recursive feature elimination with cross-validation.**

I. INTRODUCTION

Short-term load forecasting (STLF), that includes the forecasting period of several hours to several days, provides important information that can be used by electricity suppliers for a day to day operational planning and optimal generation scheduling by system operators [1]. On the other hand, false predictions may cause power imbalance, for which the electricity utility will be penalized with additional costs. In order to avoid the negative effects of inaccurate predictions, it is inevitably important the usage of highly efficient and accurate load forecasting model.

The process of creating a load forecasting model can be divided into four stages:

- Data acquisition
- Data preparation
- Feature selection
- Model creation

In the first stage, through load profile analysis and literature review, the potential features that may be used in the load forecasting model are selected and gathered. In the second stage, using pre-processing techniques, useful information is extracted from the raw data that was obtained in the first stage. In order to increase the model's efficiency and accuracy, using a selection algorithm, the features with the most significant influence on the load are selected. Finally, using an appropriate algorithm, a model is trained on the selected features and can be used for making predictions on the future load patterns.

In this paper, we study the advantages of using a feature selection algorithm (Recursive Feature Elimination (RFE)) on improving the accuracy of a Feed Forward Neural Network (FFNN). Additionally, a comparison between a FFNN trained for the whole forecasting period and an assembled model consisting of several FFNN's for a particular hour in the forecasting period is given.

The rest of this paper is organized as follows. The second section gives a brief description of the mathematical basis of the Recursive Feature Elimination algorithm and Feed Forward Neural Networks. The following section describes the process of creating a load forecasting model for the Macedonian Power System. The results obtained from testing the model are presented in the fourth section. Finally, in the fifth section, a conclusion of the paper is given.

II. METHODOLOGY

The need of an accurate and effective forecasting model that deals with the non-linear relationship between features and load consumption, led to creations of a wide range of methods for short-load forecasting. FFNN have the ability to map the non-linear relationship between features and load, even with a simple structure, consisting of one hidden layer. The model's efficiency and accuracy can be increased by selecting a subset of features from the input feature set that has the most significant influence on load consumption. This is done by ranging the features according to a statistical importance measures. This process of selecting a subset of features for creating a model is called feature selection. Feature selection leads to reducing the number of features for model creation, which leads to a simpler model with reduced training time and increased efficiency. The reduction of features also increases the model's accuracy by amplifying the effect of the features that have greater influence on load consumption and eliminating the features with less influence.

A. F-test

As a statistical measurement for estimating the significance of the difference between model's trained with different set of

1 Goran Veljanovski, Pande Popovski, Metodija Atanasovski and Mitko Kostov are with the Faculty of Technical Sciences-Bitola, St. Kliment Ohridski University, Republic of Macedonia, E-mail: goran.veljanovski@uklo.edu.mk.

978-1-6654-8500-5/22/$31.00 ©2022 IEEE

features, F-test is utilized. F-test gives an information about the effect of each parameter on the model's performances. Through F-scores and p-values, F-test estimates the feature effects on the model's accuracy. This is done by first calculating the correlation p_i between each feature $X[i]$ and the target array y :

$$p_i = \frac{(X[i] - mean(X[i])) \cdot (y - mean(y))}{std(X[i]) * std(y)}. \tag{1}$$

Afterwards, the F-score of the individual features F_i is calculated as:

$$F_i = \frac{p_i^2}{1 - p_i^2}(n-2) \tag{2}$$

where n is the number of samples. Finally, using the F-score of the individual features, ranking of the feature is performed. The features with higher F-score have more significant influence on the target values, and the less correlated features have lower F values.

B. Recursive Feature Elimination

RFE first trains a forecasting model on a certain set of features and tests the model's performances. Then, the information about the model's accuracy and the ranking of the features according to a certain criterion are used for removing the less important features, leading to a smaller set of features. In the next step, the pruned set of feature is used for training the model. By repeating the process, the optimal number and combination of features leading to a model with less error is obtained. The pseudo code-code for the RFE algorithm is presented on fig.1.

Inputs:
 Training set T
 Set of p features $F = \{f_1 . \ldots, f_p\}$
 Ranking method $M(T, F)$
Outputs:
 Final ranking R
Code:
 Repeat for i in $\{1 : p\}$
  Rank set F using $M(T, F)$
  $f^* \leftarrow$ last ranked feature in F
  $R(p - i + 1) \leftarrow f^*$
  $F \leftarrow F - f^*$

Fig 1. Pseudo-code for the RFE algorithm

C. Feed Forward Neural Networks

FFNN as an algorithm for creating a forecasting model performs non-linear mapping between features and target values by learning from examples. FFNN consists of neurons that are arranged into layers: an input layer, to which the model's input is fed; an output layer, which gives the model's output, and one or more hidden layers. The structure of a FFNN is presented on fig 2. Each neuron from one layer is connected to each neuron of the following layer via weight w_i. The output of each neuron is calculated by summing all the weighted inputs x_i, and passing them into an activation function *f*. In a FFNN consisting of three layers, the output of the network y_i is calculated as:

$$y_i = \sum_{j=0}^{h} \left[w_j f\left(\sum_{i=0}^{d} w_{ij} x_i \right) \right], \tag{3}$$

where d is the number of samples in the input vector, h the number of neurons in the hidden layer and w_{ij} the weight between the neurons.

The FFNN learns the non-linear relationship between the input features and the targets by adjusting the weights through minimizing a certain cost function E_D :

$$E_D = \frac{1}{2N} \sum_{i=0}^{N} \{y_i - t_i\}^2 , \tag{4}$$

where t_i is the target value and N number of input – output pairs.

Using the back propagation algorithm, the weights are updated in order to minimize the cost function for all training examples.

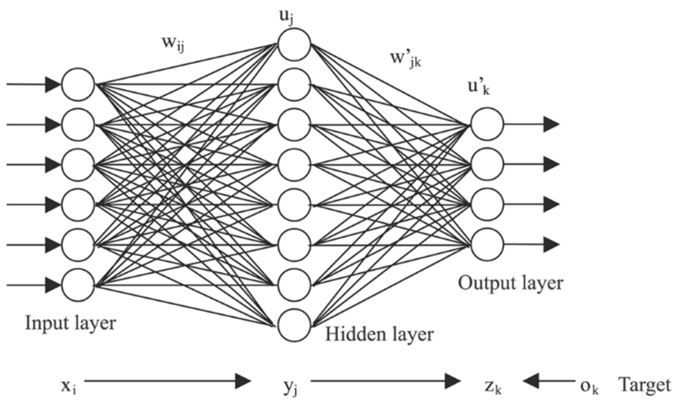

Fig 2. Structure of a FFNN

III. Case Study

In order to demonstrate the advantages of feature selection, a STLF model was created for load forecasting for the Macedonian Power System.

TABLE I
LIST OF POTENTIAL FEATURES

Feature	Abbreviation
Load consumption of the previous hours	LoadLag1, LoadLag2,…, LoadLag168
Maximum consumption of the last 24 hours	Max24
Minimum consumption of the last 24 hours	Min24
Average consumption of the last 24 hours	Avg24
Temperature of the previous hours	TmpLag1, TmpLag2,…, TmpLag168
Predicted temperature	Tmp1, Tmp2,…, Tmp24
Radiation of the previous hours	RadLag1, RadLag2,…, RadLag168
Predicted radiation	Rad1, Rad2,…, Rad24
Humidity of the previous hours	HumLag1, HumLag2,…, HumLag168
Predicted humidity	Hum1, Hum2,…, Hum24
Hour of day	hr
Day of week	dw
Day of year	dy

D. Data acquisition

There are many factors that affect the load consumption and their influence depends on the type of consumers and region under consideration [1] and [4]. By analyzing the load curves for the Macedonian Power System, the features that are mostly used as potential features are selected and obtained for the model creation [1-3]. These features presented in Table 1 include: load data, statistical data and meteorological data for the time period from 2014 to 2019.

E. Data preparation

A set of load data contains hourly load consumption from 1 January 2014 to 31 December 2019. On the other hand, meteorological data is available for every half an hour. In order to match the date, data reduction is performed. Through data reduction, small gaps in the data are filled using linear regression, and skipped if more data are missing. Statistical analysis is performed on load and meteorological data in order to obtain statistical data. The obtained calendar data is transformed to appropriate numerical values using data transformation. Afterwards, all the data is normalized between the values of 0 and 1. Finally, the data is divided in training set (containing 80%), validation set (containing 10%) and testing set (containing the remaining 10%).

F. Feature selection

Forecasting the load curve for the following day for the Macedonian Power System is performed by creating different hourly models for the forecasting period. The selection of the optimal number of models for the next day forecast, and the optimal combination and number of features for every model is performed using Recursive Feature Elimination with Cross-Validation (RFECV). A p-value of 0.1 and regression trees algorithm were chosen for obtaining the F-score of the features. The results from the RFECV suggest that the optimal number of hourly models for the 24-hour load forecasting is four, the first three models for forecasting the load consumption of the first three hours of the forecasting period, and the fourth model for forecasting the load of the remaining 21 hours. Each of these models uses different number of features that are listed in Table II.

TABLE II
SELECTED FEATURES BY RFECV

Forecasting period	Optimal features
First hour	LoadLag24,TempLag23,LoadLag4, LoadLag2,TempLag1,LoadLag1
Second hour	LoadLag24,LoadLag5,LoadLag2, TempLag1,LoadLag1
Third hour	LoadLag24,LoadLag6,LoadLag5, LoadLag3,LoadLag2,LoadLag1
Remaining hours	Tmp1, LoadLag24

G. Model creation

According to the results obtained from RFECV, four models that use an appropriate set of features are created using FFNN and assembled for forecasting the load curve of the following day. The accuracy of FFNN increases with increasing of their complexity, but a satisfactory

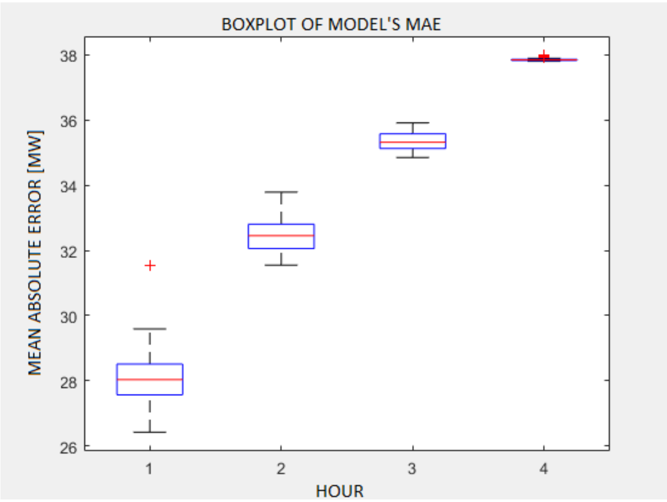

Fig 3. Boxplot of model's MAE

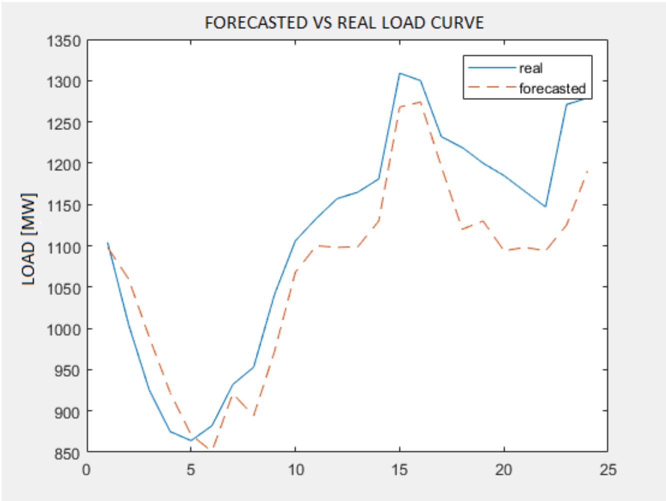

Fig 4. Comparison of forecasted and real load curve for a specific day

compromise is achieved, between model efficiency and accuracy using two hidden layers [3]. For this reason, each model is created with two hidden layers with 50 hidden neurons in each layer. The sigmoid activation function is chosen for the hidden layer, while linear activation function was chosen for the output layer. The models are trained on the training data set.

IV. Results

In order to avoid the uncertainties in the models performances imposed by the sensitivity of the initial values by the weights and biases, every model is trained 100 times on the same training data set and the performances of the models are statistically analyzed. The obtained results from the statistical analysis are presented via boxplot in Fig 3. As it can be concluded from Table II and Fig 3, the first model, which uses features that are closer to the start of the forecasting period, has the best performances. On the other hand, forecasting the load consumption after the fourth hour of the forecasting period is done based on the load consumption of the previous day and the forecasted temperature for the forecasting hour.

After training the models, they are tested on the test data set consisting the period from 1 January 2019 to 31 December 2019. The performances of the forecasting model are evaluated using Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE). The statistical analysis resulted with MAE = 37.88 MW with dispersion of 0.045 MW and RMSE = 48.96 MW with dispersion of 0.069 MW.

Fig 4 compares the forecasted load curve of a particular day and the real load curve. In this case, the Mean Absolute Error is 17.88 MW with a dispersion of 0.048 MW.

V. Conclusion

Comparing the results obtained from testing the proposed model for STLF with the models in [2] and [3], which are trained and tested on the same data, but use only one model for forecasting the load of the following 24 hours, shows an improvement in model accuracy of 8.65% and 9.84%, respectively.

These results confirm the advantages of using a feature selection algorithm, in particular, Recursive Feature Elimination with Cross-Validation,for determining the optimal number of sub-models for creating a next day load forecasting model, and also improve the efficiency and accuracy of the model by reducing the number of features for training the model, thus amplifying the influence of the more important features. RFECV together with the suggested approach of combining hourly models into one single model for next day forecasting, increase the accuracy of forecasting the load curve for the following day.

Acknowledgement

This research is supported by the EU H2020 project TRINITY (Grant Agreement no. 863874) This paper reflects only the author's views and neither the Agency nor the Commission are responsible for any use that may be made of the information contained therein.

References

[1] A. M. Pirbazari, A. Chakravorty and C. Rong, "Evaluating Feature Selection Methods for Short-Term Load Forecasting," 2019 IEEE International Conference on Big Data and Smart Computing (BigComp), 2019, pp. 1-8, doi: 10.1109/BIGCOMP.2019.8679188.

[2] A. Dedinec,S. Filiposka, A. Dedinec, L. Kocarev, Deep belief network based electricity load forecasting: An analysis of Macedonian case,Energy,Volume 115, Part 3, 2016,Pages 1688-1700, ISSN 0360-5442

[3] G. Veljanovski, M. Atanasovski, M. Kostov and P. Popovski, "Application of Neural Networks for Short Term Load Forecasting in Power System of North Macedonia," 2020 55th (ICEST),2020,pp.99-101,doi: 10.1109/ICEST49890.2020.9232.

[4] X. Fei and W. Zhigang, "Analysis of correlation between meteorological factors and short-term load forecasting based on machine learning," 2018 International Conference on Power System Technology (POWERCON), 2018, pp. 4449-4454, doi: 10.1109/POWERCON.2018.8601585.