Incremental ensemble learning for electricity load forecasting

Gabriela Grmanová, Peter Laurinec, Viera Rozinajová, Anna Bou Ezzeddine, Mária Lucká, Peter Lacko, Petra Vrablecová, Pavol Návrat

Faculty of Informatics and Information Technologies, Slovak University of Technology in Bratislava, Ilkovičova 2, 842 16 Bratislava, Slovak Republic, {gabriela.grmanova, peter.laurinec, viera.rozinajova, anna.bou.ezzeddine, maria.lucka, peter.lacko, petra.vrablecova, pavol.navrat}@stuba.sk

Abstract: The efforts of the European Union (EU) in the energy supply domain aim to introduce intelligent grid management across the whole of the EU. The target intelligent grid is planned to contain 80 % of all meters to be smart meters generating data every 15 minutes. Thus, the energy data of EU will grow rapidly in the very near future. Smart meters are successively installed in a phased roll-out, and the first smart meter data samples are ready for different types of analysis in order to understand the data, to make precise predictions and to support intelligent grid control. In this paper, we propose an incremental heterogeneous ensemble model for time series prediction. The model was designed to make predictions for electricity load time series taking into account their inherent characteristics, such as seasonal dependency and concept drift. The proposed ensemble model characteristics – robustness, natural ability to parallelize and the ability to incrementally train the model – make the presented ensemble suitable for processing streams of data in a "big data" environment.

Keywords: big data; time series prediction; incremental learning; ensemble learning

1 Introduction

Generating large amounts of data has become part of our everyday life. In reality, human activities produce data that in recent years have rapidly increased, e.g. as measured through various sensors, regulation systems and due to the rapid development of information technologies [3]. "Big data" significantly changes the nature of data management as it introduces a new model describing the most significant properties of the data -volume, velocity and variety. Volume refers to the vast amounts of data requiring management, and it may not stem from the number of different objects, but

http://doi.org/10.12700/APH.13.2.2016.2.6

from the accumulation of observations about these objects in time or in space. Velocity can be determined by the rate of acquisition of streams of new data, but also by application requirements, where it is necessary to make a very fast prediction, as the result of a particular user's request. This will comprise research of methods and models for big data analysis, whether with low latency, or even in real time.

In our work, we focus on stream data coming from smart metering. The smart meters are able to send measurements of power consumption every 15 minutes thus, providing new possibilities for its modelling and prediction. The most useful aspect of having this vast amount of data is the ability to forecast the power demand more precisely. This is particularly important when viewed with regard to the fact that the possibility to store electricity is very limited. With accurate predictions, the distributor can reliably deliver electricity and fulfil the power authorities' regulations, which protect the distribution network from being at too high or too low a voltage. It also helps to flexibly react to different unexpected situations like large-scale blackouts.

The number of smart meters increases rapidly every day which results in production of large amount of data. Classical methods can fail to process such amount of data in reasonable amount of time; therefore it is necessary to focus on parallel and distributed architectures and design methods and applications that are able to automatically scale up depending on the growing volume of data.

The classical prediction methods of electricity consumption are: regression analysis and time series analysis models. These approaches will not be sufficient in the near future, as the European Union's efforts are aimed at introducing an intelligent network across the whole of the European Union. This fact raises new perspectives in modelling and predicting power demand.

A significant feature of many real-world data streams is *concept drift*, which can be characterized as the arbitrary changes of data characteristics. The occurrence of concept drift in a data stream can make classical predictive techniques less appropriate therefore, new methods must be developed. The typical example of concept drift is a change of workload profile in a system for controlling the load redistribution in computer clusters [48] or a change of user's interests in information filtering and recommendation modelling [14], [26]. In power engineering are concept drifts caused by change of consumers' behaviour during holidays, social events, different weather conditions, or summer leaves in big enterprises – the biggest electricity consumers.

This paper introduces a new approach to electrical load forecasting. It takes into consideration the aspect of concept drift, and is based on the principle of ensemble learning. It is organized as follows: the second chapter is devoted to the characteristics of the problem and the third contains the summary of the related work. In the next chapter we describe our proposed approach (the incremental heterogeneous ensemble model for time series prediction). The experimental evaluation is presented in Chapter 5 and the overall evaluation and discussion is given in Chapter 6.

2 Characteristics of the problem

As mentioned earlier, after the widespread introduction of smart meters, the data provided will satisfy the first characteristic of Big Data based on volume. To analyse these data, it is appropriate to propose parallel methods that can be solved in distributed environments. Because electricity loads can be seen as a stream of incoming data, it is necessary to focus on adaptive methods that are able to learn incrementally.

There are also additional important characteristics that must be taken into account – the presence of concept drifts and strong seasonal dependence. The values of any variable evolving in time, such as the electricity load, often change their behaviour over time. These changes may be sudden or gradual. In the literature, both types of changes are termed *concept drift*. Narasimhamurthy and Kuncheva [35] define the term *concept* as the whole distribution of the problem and represent it by joined distribution of data and model parameters. Then, concept drift may be represented by the change of this distribution [15]. Besides cases of concept drift when the change is permanent, one can often observe changes that are temporary. They are caused by the change of some conditions and after some time, these conditions can again change back. Moreover, seasonal changes may be considered to be concept drift, too.

In electricity load measurement, two types of concept drift can appear. The first one is permanent or temporary change that may be caused by the change of economical or environmental factors. The second type of concept drift is seasonal, caused by seasonal changes of weather and the amount of daylight. Seasonal dependency can be observed as daily, weekly and yearly levels. That is why it is necessary to consider these two possible sources of concept drift in any model proposal.

3 Literature review

In this section we present methods used to compute time series predictions – classical, incremental and ensemble approaches.

3.1 Classical approaches

Classical approaches to time series prediction are represented mainly by regression and time series analysis. Regression approaches model the dependencies of target variables on independent variables. For electricity load prediction, the independent variables can be the day of the week, the hour of the day, the temperature, etc. Plenty of different regression models were presented in the literature, such as a step-wise regression model, a neural network and a decision tree [45].

Because of strong seasonal periodicities in electricity load data, time series models are often used to make predictions. Mainly, Box-Jenkins methodology [5] with AR, MA, ARMA, ARIMA and derived models are applied.

However, the classical approaches are not able to adapt to incoming streams of data and thus, are not suitable for electricity load demand forecasting.

3.2 Incremental learning

Incremental learning algorithms are able to adapt to new emerging data. They process new data in chunks of appropriate size. They can possibly process the data chunks by off-line algorithms.

Polikar et al. [39] defined the four desired properties of an incremental learning algorithm – the ability to learn new information from arriving data, the capability of working independently on historical data, the storage of previously learned knowledge, and the accommodation of new classes of data on their arrival. Minku [32] extends this definition and emphasizes that, in changing environments, where the target variable might change over time, only the useful knowledge will be stored.

Most of the incremental learning algorithms we encountered in the literature are based on machine learning, e.g. incremental support vector machines [51] and extreme learning machines [17]. Recently, the incremental ARIMA algorithm was proposed for time series prediction [34].

Usually, the incremental learning algorithms alone cannot sufficiently treat changes in the target variable. In order to cope with a changing environment, groups of predictors, i.e. ensemble models, are used to achieve better predictions.

3.3 Ensemble learning

Ensemble learning is an approach that uses a set of base models, where each model provides an estimate of a target variable – a real number for a regression task. The estimates are then combined to make a single ensemble estimate. The combination of base estimates is usually made by taking a weighted sum of base estimates. The idea behind it is that the combination of several models has the potential to provide much more accurate estimates than single models. In addition, they have several more advantages over single models, namely the scalability, the natural ability to parallelize and the ability to quickly adapt to concept drift [52]. A great introduction to ensemble learning can be found in [32].

Several empirical studies showed that the accuracy of the ensemble depends on the accuracy of base models and on the diversity among them [11], [12], [28]. The diversity of base models may be accomplished by two different approaches – *homogeneous* and *heterogeneous ensemble learning* [52]. In homogeneous learning, the ensemble is formed by models of the same type that are learned on different subsets of available data. The heterogeneous learning process applies different types of models. The

combination of homogeneous and heterogeneous approaches was also presented in the literature.

The best known methods for homogeneous ensemble learning are bagging [6] and boosting [13]. These approaches have been shown to be very effective in improving the accuracy of base models. To accomplish adaptive ensemble learning for online stream environments, two approaches are known from the literature. The first one – *incremental ensemble learning* – learns the base methods from different chunks of data. The second one – *the ensemble of online/incremental methods* – uses adaptive base methods, that are updated in an online (after each example) or incremental (after a chunk of data is available) manner. *Incremental ensemble learning* employs non-incremental algorithms to provide incremental learning. Wang et al. [46] proposed a general framework for mining data streams using weighted ensemble classifiers. The proposed algorithm adapts to changes in data by assigning weights to classifiers proportional to their accuracy over the most recent data chunk. Another approach was published by Kolter and Maloof [27]. They developed a dynamic weighted majority algorithm, which creates and removes weighted base models dynamically based on changes in performance.

Ensemble of online/incremental methods employ online/incremental base models. These approaches include online versions of well-established approaches such as online bagging and online boosting incorporating online base models [37], [38]. Another approach proposed by Ikonomovska et al. [24] introduces an ensemble of online regression and option trees.

Heterogeneous ensemble learning represents a different way of introducing the diversity of base models into ensemble, with the aim of combining the advantages of base algorithms and to solve problems of concept drift [16], [54], [40]. Different models are trained on the same training dataset; in the case of stream data on the up-to-date data chunk.

From the literature several combinations are also known of heterogeneous and homogeneous learning. Zhang et al. [55] present aggregate ensemble learning, where different types of classifiers are learned from different chunks of data.

The essential part of the ensemble learning approach is the method that is used to combine estimates of base models. For regression problems, this is done by a linear combination of the predictions. The sum of the weights which are used in the combination is 1. The weights are computed by different methods, such as basic or general ensemble methods, linear regression models, gradient descent or by evolutionary or biologically inspired algorithms, e.g. particle swarm optimization or "cuckoo search" [31], [30], [50].

Ensemble learning was also used to predict values of time series. Shen et al. [42] apply an ensemble of clustering methods to cluster 24-hour segments. Based on cluster labels, the segments are converted to sequences. Each testing sequence is matched to the training subsequences, and matching subsequences are averaged to make the prediction for a subsequent segment of the testing sequence. The predictions based on 5 different

clustering methods are combined in the ensemble, where the weights are iteratively updated. Chitra and Uma [7] present an ensemble of RBF-network, k-nearest neighbour and self-organizing maps for a time series prediction. Wichard and Ogorzałek [47] describe the use of an ensemble method for their time series prediction. They use an ensemble of linear and polynomial models, k-nearest neighbour, nearest trajectory models and neural networks, with an RBF-network for "one-step- ahead" prediction.

All of these approaches use ensembles of regression models for generating time series predictions. They do not take explicitly any seasonal dependence into account and do not use time series analysis methods to make predictions.

4 The incremental heterogeneous ensemble model for time series prediction

In this section we propose the *incremental heterogeneous ensemble model for time series prediction*. The ensemble approach was chosen for its ability to adapt quickly to changes in the distribution of a target variable and its potential to be more accurate than a single method. Since we focus on time series with strong seasonal dependence, in ensemble models we take into account different types of seasonal dependencies. Models for yearly seasonal dependence need to be computed based on one year of data. These models can be recomputed once a year. The models coping with daily seasonal dependence need only data from one to several days and can be computed in an incremental manner. The potential of the proposed ensemble is its ability to deal with the scalability problems of big data. Predictions of base models can be computed in parallel or in distributed environment in order to reduce computation time and to scale up to incoming amount of data which makes the proposed ensemble suitable for big streams of data.

The base models used in the ensemble are of two types – regression models and models for time series analysis. Regression models can potentially incorporate additional dependencies, such as temperature. Time series analysis models are suitable to capture seasonal effects.

4.1 Incorporating different types of models

The proposed ensemble model incorporates several types of models for capturing different seasonal dependencies. The models differ in *algorithm*, *size of data chunk* and *training period* (see Figure 1). Different algorithms are assumed in order to increase the diversity of the models. The size of each data chunk is chosen in order to capture particular seasonal variation, e.g. data from the last 4 days for daily seasonal dependence. However, the model that is trained on a data chunk of 4 days' data, can be trained again as soon as the data from the next day (using a 1-day training period) are available. The new data chunk overlaps with the previous one in 3 days.

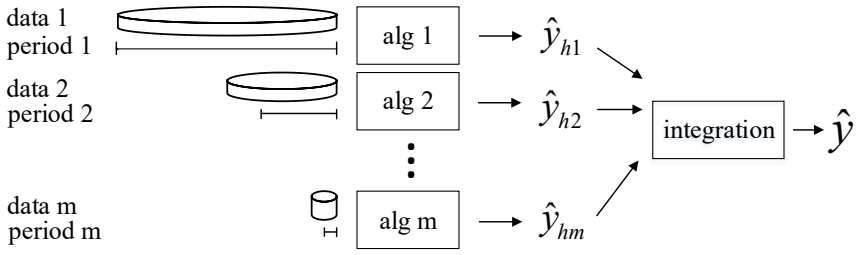

Figure 1
Schematic of ensemble learning

The ensemble model is used to make one-day predictions. Let h be the number of observations that are daily available. At day t, the ensemble makes h predictions by the weighted average of predictions made by m base models. After the observations for the current day are available, the prediction errors are computed. Based on computed errors, the weights are updated and each base model $i=1, \ldots, m$, for which t fits its training period p_i, is retrained on a data chunk of size s_i.

Let $\hat{Y}^t$ be the matrix of predictions of m base methods for the next h observations at day t:

$$\hat{Y}^t = \begin{pmatrix} \hat{y}_{11}^t & \cdots & \hat{y}_{1m}^t \\ \vdots & \ddots & \vdots \\ \hat{y}_{h1}^t & \cdots & \hat{y}_{\mathrm{hm}}^t \end{pmatrix} = (\hat{y}_1^t \quad \ldots \quad \hat{y}_m^t)$$

and $w^t = (w_1^t \quad \ldots \quad w_m^t)^T$ is a vector of weights for m base methods at day t before observations of day t are available. Weights w_j^t are initially set to 1. Weights and particular predictions are combined to make an ensemble prediction $\hat{y}^t = (\hat{y}_1^t \quad \ldots \quad \hat{y}_m^t)^T$. The ensemble prediction for k-th ($k = 1, \ldots, h$) observation is calculated by:

$$\hat{y}_k^t = \frac{\sum_{j=1}^m \hat{y}_{\mathrm{kj}}^t \bar{w}_j^t}{\sum_{j=1}^m \bar{w}_j^t}$$

where $\bar{w}^t$ is a vector consisting of weights rescaled to interval $\langle 1, 10 \rangle$.

After observations of day t are available, the weights vector can be recomputed. From the prediction matrix $\hat{Y}^t$ and the vector of h current observations $y^t = (y_1^t \quad \ldots \quad y_h^t)^T$, the vector $e^t = (e_1^t \quad \ldots \quad e_h^t)^T$ of errors for m methods is computed. The error of each method is given by $e_j^t = \mathrm{median}(|\hat{y}_j^t - y^t|)$. A vector of errors e^t is used to update the weights vector of base models in the ensemble. The weight for j-th method is calculated by:

$w_j^{t+1} = w_j^t \frac{\mathrm{median}(e^t)}{e_j^t}$ The advantages of the presented type of weighting is its robustness and the ability to recover the impact of base methods. The weighting and integration method is robust since it uses the median absolute error and the median of errors. In

contrast to the average, the median is not sensitive to large fluctuations and abnormal prediction errors. A rescaling method, one that does not let the particular weights drop to zero, enables the ensemble to recover the impact of particular base methods in the presence of concept drift.

4.2 Base models

In heterogeneous ensemble models, it is important to integrate the results of diverse base methods. We used 11 different algorithms. The methods are of different complexity, from very simple, e.g. a naïve average long-term model, to complex, such as support vector regression. They assume different seasonal dependencies, from daily to yearly. The presented base methods can be divided into the set of methods based on regression analysis and those based on time series analysis.

4.2.1 Regression algorithms

Multiple linear regression (MLR) attempts to model the relationship between two or more explanatory variables and a response variable by fitting a linear equation to observed data. Rather than modelling the mean response as a straight line, as it is in simple regression, the model is expressed as a function of several explanatory variables.

Support Vector Machines are an excellent tool for classification, novelty detection, and regression (SVR). It is one of the most often used models for electricity load forecasting. SVM is a powerful technique used in solving the main learning problems. We have used the method based on epsilon-regression based on the radial basis Gaussian kernel, and also tested it in combination with the wavelet transform ($\varepsilon = 0.08$ for deterministic part and 0.05 for fluctuation part) [53].

4.2.2 Time series algorithms

The autoregressive model (AR) expresses the current value of electricity load as a linear combination of previous electricity load values and a random white noise [33]. The current value of the modelled function is expressed as a function of its previous n values on which it is regressed.

Feed-forward neural networks (NNE) are biologically inspired universal approximation routines [22]. They were successfully used for solving prediction problems [43]. R package *forecast* [23] contains the *nnetar* method, which is a feed-forward neural network with a single hidden layer and lagged inputs, for forecasting univariate time series. It provides the capability to train a set of neural networks on lagged values for one-step forecasting. The prediction is an average of those neural networks predictions. Number of neurons in hidden layer was determined as a half of the number of input nodes plus one.

The *Holt-Winters exponential smoothing* (HW) [20], [49] is a prediction method applied to a time series, whereby past observations are not weighted equally, as it is in ARMA models, but the weights decrease exponentially with time. So the data that are

closer in time can influence the modelling more strongly. We have considered seasonal changes with and without any trend in triple exponential smoothing (we have chosen the smoothing parameters $\alpha = 0.15$, $\beta = 0$, $\gamma = 0.95$), and combined this model with the wavelet transform (shrinkage method was chosen soft thresholding and threshold estimation was universal). The original load data series were decomposed into two parts - deterministic and fluctuation components, and then the regression of both parts was calculated separately. The resulting series were obtained with suitable wavelet coefficient thresholds and the application of the wavelet reconstruction method.

Seasonal decomposition of time series by loess (STL) is a method [8] that decomposes a seasonal time series into three parts: trend, seasonal and remaining. The seasonal component is found by *loess (local regression)* smoothing the seasonal sub-series, whereby smoothing can be effectively replaced by taking the mean. The seasonal values are removed, and the remainder is smoothed to find the trend. The overall level is removed from the seasonal component and added to the trend component. This process is iterated a few times. The remaining component represents the residuals from the seasonal plus trend fit.

STL decomposition works similarly to wavelet transform. For the resulting three time series (seasonal, trend and remainder) the result is used separately for prediction with Holt-Winters exponential smoothing and ARIMA model.

The ARIMA model has been introduced by Box and Jenkins [5] and is one of the most popular approaches in forecasting [21]. It is composed of three parts: autoregressive (AR), moving average (MA), and the differencing processes. In the case of non-stationary processes, it is important to transform the series into a stationary one and that is usually done by differentiation of the original series.

Seasonal naïve method-Random walk (SNaive) is only appropriate for time series data. All forecasts are simply set to be the value of the last observation. It means that the forecasts of all future values are set to be equal to the last observed value. A similar method is also useful for highly seasonal data, where each forecast value is set to be equal to the last observed value from the same season of the year (e.g., the same month of the previous year).

Double seasonal exponential smoothing (TBATS) forecasting is based on a new state space modelling framework [10], incorporating Box-Cox transformations, Fourier series with time varying coefficients and ARMA error correction. It was introduced for forecasting complex seasonal time series that cannot be handled using existing forecasting models. These types of complex time series include time series with multiple seasonal periods, high frequency seasonality, non-integer seasonality and other effects. The modelling is an alternative to existing exponential smoothing models, and has many advantages.

Naïve average long-term method is based on the assumption that non-seasonal patterns [36] and trends can be extrapolated by means of a moving-average or smoothing model. It is supposed, that the time series is locally stationary and has a slowly varying mean. The moving (local) average is taken for the estimation of the current value of the mean

and used as the forecast for the near future. The simple moving average model predicts the next value as a mean of several values. This is a compromise between the mean model and the random-walk-without-drift-model.

Naïve In median long-term method is an alternative to the previous method. The use of a moving average is not able to react in the case of rapid shocks or other abnormalities. In such cases a better choice is to take a simple moving median over the last n time series' items. A moving average is statistically optimal for recovering the underlying trend of the time series when the fluctuations about the trend are normally distributed. It can be shown that if the fluctuations are Laplace distributed, then the moving median is statistically optimal [2].

5 Experimental evaluation

In this section we describe how data is used for the evaluation of the ensemble method; we provide details of the experiments and then we present the results.

5.1 Data

An experimental sample of data comes from smart meters installed in Slovakia that perform measurements every 15 minutes. Currently, the smart meters operate in around 20,000 consumers' premises. Based on legislation, this number will soon be higher and the amount of incoming data will significantly increase. The data has the potential to become "big" and "fast", because of its incremental and stream character. The consumers are small and medium enterprises. The data are anonymized and only postal codes are available. We created 10 samples by grouping customers according to regions. By doing this we simulate electricity load values at secondary distribution substations. We summed the quarter-hourly data to predict the load of the regions. Table 1 describes the data samples. The studied data samples show collected values from July 1st, 2013 to February 15th, 2015 (596 days, see Figure 2). The sudden changes in load were observed during holidays (e.g., summer leave, and Christmas).

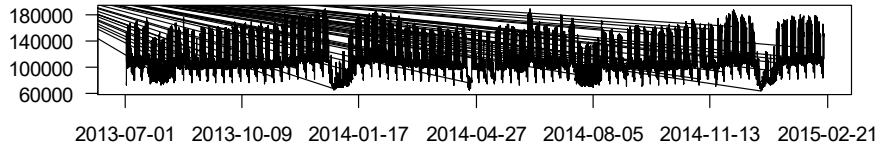

Figure 2
Electricity consumption for Bratislava region over period of 596 days (in kW per 15 minutes)

Table 1
Description of ten data samples and their electricity loads (in kW per 15 minutes)

postal code	region	no of delivery points	average	average per delivery point
04	Košice	722	35,501.854	49.172
05	Poprad	471	17,135.133	36.380
07	Trebišov	382	11,571.184	30.291
08	Prešov	580	18,671.795	32.193
8	Bratislava	1314	119,691.911	91.090
90	Záhorie	773	41,402.715	53.561
92	Piešťany	706	74,340.781	105.296
93	Dunajská Streda	594	28,196.959	47.470
95	Partizánske	584	34,298.912	58.731
99	Veľký Krtíš	114	2,124.887	18.639

The load during the typical week (see Figure 3) consists of the four segments – Mon, Tue—Fri, Sat and Sun. To minimize the noise in the data and to improve our predictors we considered only the Tue—Fri segment, i.e. the days with similar behaviour.

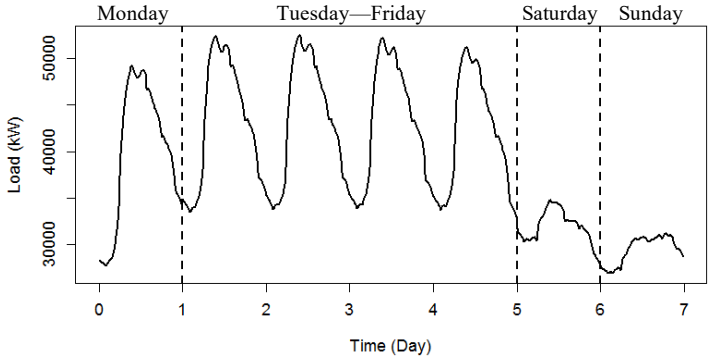

Figure 3
Average weekly electricity load (without holidays).

5.2 Measures of prediction accuracy

To measure prediction accuracy we utilize three measures. *Mean absolute error* (MAE) and *root mean squared error* (RMSE) are commonly used measures of prediction error in time series analysis. The main difference between RMSE and MAE is that RMSE amplifies large errors. *Symmetric mean absolute percentage error* (sMAPE) is an accuracy measure based on relative (percentage) errors that enables us to compare percentage errors for any time series with different absolute values:

$$\text{sMAPE} = \frac{1}{n}\sum_{t=1}^{n}\frac{|\hat{y}_t - y_t|}{(\hat{y}_t + y_t)}$$

5.3 Experiments

To design the experiments, the best data chunk sizes for particular models were found experimentally. The most precise predictions for regression methods (MLR and SVR) were for days. Time series analysis models (AR, HW, STL+EXP, STL+ARIMA) coping with daily seasonality and NNE performed the best with data chunk size equal to 10 days. Based on its nature, SNaive needed only a 1-day long data chunk. Long-term double seasonal exponential smoothing (TBATS), incorporating two seasonal dependencies with 1 and -day periods, used data chunks the size of 41days – 1/3 of days of the test set. Naïve average and median log-term models use 1-year data chunks. In fact, in our experiments we had only 116 days in the test set for which observations from the previous year were available. Thus, 116-days long data chunk was used.

The training period for methods working with short-term seasonal dependency was 1 day. Models coping with yearly seasonal dependency have a 1 year period and since we had less than 2 years of data available, they were trained only once and were not further retrained.

Since there were only available data for training (both previous 10 days and previous 1 year observations) for the period July 1st, 2014 – February 15th, 2015, these were used as a test set. Namely: only non-holiday Tue-Fri days were assumed. The test set consisted of 116 days each having 96 observations. Initially, models were trained on respective chunks from a training set with equal weights in the ensemble. Then, the models were incrementally retrained according to their periods, while subsequently adding new data from the test set and ignoring the old ones.

The experiments were provided in an R environment. We used methods from a standard *stats* package and from *forecast* [23] (STL+EXP, STL+ARIMA, NNE, SNaive and TBATS), *wmtsa* [9] (wavelets) and kernlab [25] (SVR) packages.

5.4 Results

Figures 4 and 5 illustrate the incremental training process for a single region. It presents predicted and measured electricity loads (Figure 4), history of weights (Figure 5 top) and errors (Figure 5 bottom). An interesting observation of concept drift can be seen at t=10 and t=22, when errors sharply rise. In the history of weights, sharp changes can be seen, too. The concept drift was caused by the summer leave in bigger enterprises, which consume most of the electricity.

Tables 3 and 4 contain average errors of predictions and their standard deviations measured by sMAPE for every region and every base method plus the ensemble method. Tables show that there is no superior base method, which gives justification for the ensemble method, where errors are, in all tested cases, smaller.

We used the Wilcoxon rank sum test [19] to evaluate the *incremental heterogeneous ensemble model for time series prediction* against the best base method. The Wilcoxon rank sum test tests the statistical hypothesis whether errors of the ensemble are significantly lower than errors of the best base method used in the ensemble. The test

used is a nonparametric alternative to the two-sample t-test. We used this nonparametric test because errors of predictions are not normally distributed (tested with Shapiro-Wilk test [41] and Q-Q plot). The Base method with the highest weight value at the end of the testing process is considered to be the best base method in the ensemble. A Statistical test on significance level α= 0.05 showed that in 9 out of 10 regions the ensemble method was significantly better than the best base methods in that region (see Table 5). The p-value exceeds the significance level for all but one region with errors measured by MAE, RMSE and sMAPE. Only for the Trebišov region, evaluated by RMSE measure, was the ensemble evaluated as smaller, but not significantly so.

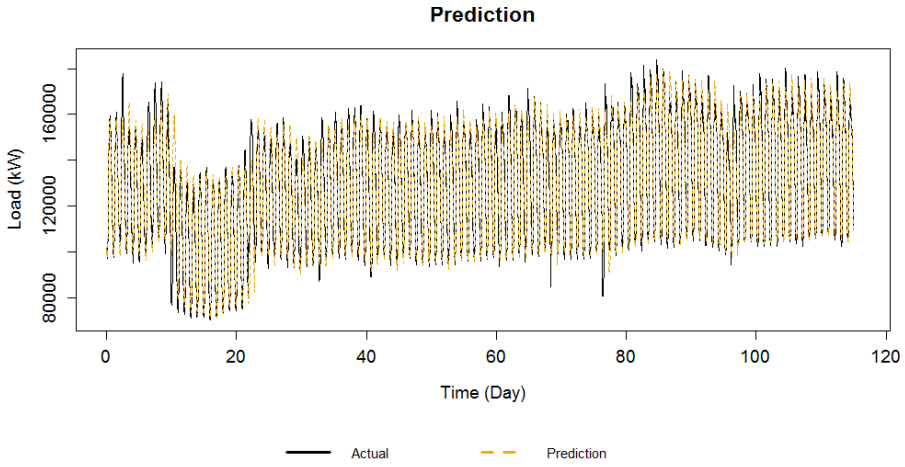

Figure 4

Results of prediction for Bratislava region. Concept drift at times t=10 and t=22 was caused by the summer leave in bigger enterprises, which consume the most of the electricity.

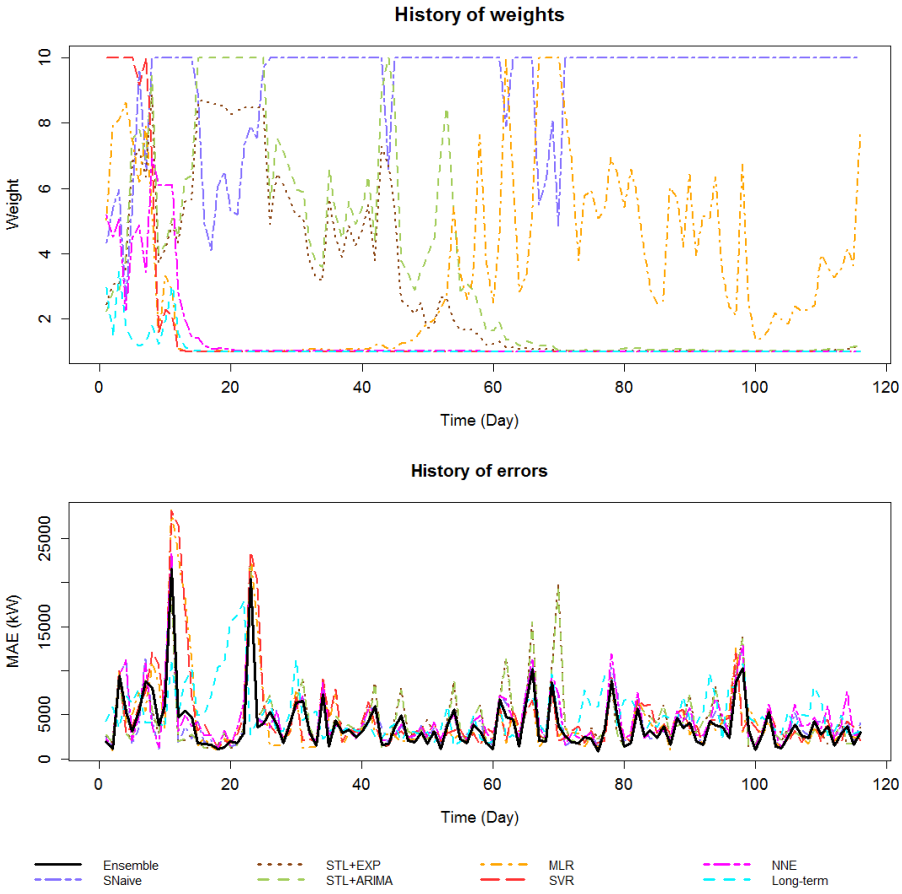

Figure 5

History of ensemble weights and prediction errors for Bratislava region. The legend belongs to both plots. The results of ensemble and following models are shown: seasonal naïve method-random walk (SNaive), seasonal decomposition of time series by loess plus Holt-Winters exponential smoothing (STL+EXP), seasonal decomposition of time series by loess plus ARIMA (STL+ARIMA), multiple linear regression (MLR), support vector regression (SVR), feed-forward neural networks (NNE) and naïve average long-term method (Long-term).

Table 3
Average and standard deviation sMAPE (%) of methods, part 1.
The best base method and the ensemble are highlighted.

Method	Bratislava	Záhorie	Košice	Piešťany	Dunajská Streda
AR	2.328 ± 1.45	2.484 ± 1.75	2.195 ± 1.30	1.876 ± 1.67	2.485 ± 1.58
HW	1.901 ± 1.46	2.744 ± 2.35	2.145 ± 1.59	1.892 ± 2.06	2.342 ± 1.83
STL+EXP	1.663 ± 1.54	2.537 ± 2.71	1.905 ± 1.62	1.759 ± 2.09	2.159 ± 1.86
STL+ARIMA	1.653 ± 1.53	2.433 ± 2.66	**1.833** ± 1.60	1.666 ± 2.17	**2.111** ± 1.79
NNE	1.695 ± 1.36	2.136 ± 1.88	1.985 ± 1.24	1.582 ± 1.73	2.325 ± 1.58
SNaive	**1.561** ± 1.36	2.090 ± 1.92	1.912 ± 1.27	**1.554** ± 1.72	2.299 ± 1.59
MLR	1.652 ± 1.85	1.994 ± 1.79	1.845 ± 1.34	1.696 ± 1.72	2.166 ± 1.69
SVR	1.773 ± 1.92	**1.948** ± 1.80	1.902 ± 1.37	1.656 ± 1.63	2.278 ± 1.76
TBATS	2.581 ± 2.43	2.724 ± 2.73	2.157 ± 1.56	2.791 ± 2.46	5.091 ± 4.84
Float mean	1.939 ± 1.30	1.789 ± 1.34	1.806 ± 1.17	1.730 ± 1.47	2.377 ± 1.56
Float med	2.627 ± 1.46	1.912 ± 1.42	1.907 ± 1.20	1.807 ± 1.52	2.441 ± 1.55
Ensemble	**1.417** ± 1.26	**1.796** ± 1.64	**1.643** ± 1.30	**1.446** ± 1.65	**1.899** ± 1.50

Table 4
Average and standard deviation sMAPE (%) of methods, part 2.
The best base method and the ensemble are highlighted.

Method	Partizánske	Prešov	Poprad	Trebišov	Veľký Krtíš
AR	2.351 ± 1.21	2.512 ± 0.74	3.005 ± 2.03	2.221 ± 1.07	5.453 ± 2.46
HW	2.837 ± 2.10	2.145 ± 1.21	2.927 ± 2.22	2.316 ± 1.56	6.983 ± 4.00
STL+EXP	2.584 ± 2.63	1.969 ± 1.28	2.729 ± 2.50	2.158 ± 1.63	5.962 ± 3.78
STL+ARIMA	2.367 ± 2.40	1.853 ± 1.15	2.487 ± 2.31	2.054 ± 1.52	**5.712** ± 3.49
NNE	2.004 ± 1.43	2.077 ± 0.91	2.402 ± 1.75	2.079 ± 1.20	6.709 ± 3.30
SNaive	1.941 ± 1.49	1.870 ± 0.95	**2.076** ± 1.74	2.006 ± 1.23	6.781 ± 3.36
MLR	1.766 ± 1.33	**1.568** ± 0.73	2.301 ± 2.53	**1.745** ± 0.93	5.658 ± 2.47
SVR	**1.806** ± 1.35	1.742 ± 0.75	2.408 ± 2.68	1.823 ± 0.92	5.523 ± 2.72
TBATS	5.017 ± 2.70	2.009 ± 0.90	3.544 ± 4.27	2.641 ± 1.86	5.621 ± 3.36
Float mean	1.723 ± 1.16	1.978 ± 0.75	2.565 ± 1.27	2.250 ± 1.82	6.769 ± 2.48
Float med	1.794 ± 1.18	2.060 ± 0.79	3.263 ± 1.73	2.296 ± 1.68	6.600 ± 3.17
Ensemble	**1.704** ± 1.43	**1.483** ± 0.73	**1.973** ± 1.74	**1.656** ± 1.05	**5.224** ± 2.67

Table 5
P-values for each region. The best base method compared to the ensemble method is in parentheses.

Region	MAE	sMAPE	RMSE
Bratislava (SNaive)	$1.4*10^{-6}$	$1.5*10^{-6}$	$1.3*10^{-7}$
Záhorie (SVR)	0.0284	0.0267	0.0021
Košice (STL+ARIMA)	$2.8*10^{-7}$	$8.0*10^{-7}$	$3.8*10^{-8}$
Piešťany (SNaive)	$1.6*10^{-4}$	$1.0*10^{-4}$	$1.6*10^{-6}$
Dunajská Streda (STL+ARIMA)	0.0001	0.0001	0.0001
Partizánske (SVR)	0.0205	0.0132	0.0015
Prešov (MLR)	0.0046	0.0018	0.0153
Poprad (SNaive)	$2.2*10^{-4}$	$2.0*10^{-4}$	$2.7*10^{-6}$
Trebišov (MLR)	0.0416	0.0324	0.0565
Veľký Krtíš (STL+ARIMA)	0.0388	0.0411	0.0110

We have used sMAPE measure because we tested our methods for single delivery point predictions, too. Single delivery points, in general, have day parts with zero consumption where MAPE evaluation fails. Comparison with other works dealing with power consumption prediction is difficult because in our work we were strongly focused on predictions during concept drifts. This is why direct error evaluation comparison is not possible. Despite it, we present some recent works of load forecasting, and try to compare them to our method.

He et al. [18] used SARIMA models to forecast the electricity demand in China. They forecasted hourly and quarter-hourly demand for next few days ahead. The MAPE error of the models was about 1.5 %. Trained models were validated on real data.

Xiao et al. [50] presented ensemble learning method for a day-ahead consumption prediction. A cuckoo search algorithm was used to find the optimal weights for combining four forecasting models. Models were based on different types of neural networks (namely BPNN, GABPNN, GRNN and RBFNN). Half-hourly load data of February 2006 – 2009 in New South Wales in Australia were used for verification. The forecasts of the ensemble model were significantly better in comparison with the results of the individual models. The average MAPE was approximately 1.3 %.

Taylor and McSharry [44] presented an empirical study of various short-term load forecasting methods, i.e. ARIMA model; periodic AR model; an extension for double seasonality of Holt-Winters exponential smoothing; an alternative exponential smoothing formulation; and a method based on the principal component analysis (PCA) of the daily demand profiles. Selected methods were evaluated on half-hourly and hourly load data from 10 European countries. The evaluation showed that the Holt-Winters smoothing provided the best average daily MAPE (ca 1.5 %).

Presented works reach MAPE around 1.5 %, some of them are using forecasting methods, which are used as base methods in our ensemble model. Forasmuch as our ensemble model delivers better results than single base methods, we can assume that it would deliver better results on presented works' datasets.

Conclusion

In this paper, we propose the *incremental heterogeneous ensemble model for time series prediction*. The model was designed to make predictions for time series with specific properties (strong seasonal dependence and concept drift) in the domain of energy consumption. Its characteristics – robustness, natural ability to parallelize and the ability to incrementally train the model – make the presented ensemble suitable for processing streams of data in a "big data" environment. The achieved results lead us to assume that the presented approach could be a prospective direction in the choice of prediction models for time series with particular characteristics.

In future work, we plan to incorporate dependencies into the model with external factors such as meteorological data and information about holidays in big enterprises in the different regions. Another interesting idea is to investigate possible correlations between different regions. These aspects should also improve the predictions.

Acknowledgement

This work was partially supported by the Research and Development Operational Programme as part of the project "International Centre of Excellence for Research of Intelligent and Secure Information-Communication Technologies and Systems", ITMS 26240120039, co-funded by the ERDF and the Scientific Grant Agency of The Slovak Republic, grant No. VG 1/0752/14.

References

[1] A. S. Alfuhaid and M. A. El-Sayed, "Cascaded artificial neural network for short-term load forecasting," *IEEE Trans. Power Syst.*, vol. 12, no. 4, pp. 1524–1529, 1997.

[2] G. R. Arce, *Nonlinear Signal Processing: A Statistical Approach*. New Jersey, USA: Wiley, 2005.

[3] A. Benczúr, "The evolution of human communication and the information revolution — A mathematical perspective," *Mathematical and Computer Modelling*, vol. 38, no. 7–9, pp. 691–708, 2003.

[4] G. E. P. Box and D. R. Cox, "An analysis of transformations," *J. Roy. Statistical Soc. Series B*, vol. 26, no. 2, pp. 211–252, 1964.

[5] G. E. P. Box and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*. San Francisco, CA: Holden-Day, 1970.

[6] L. Breiman, "Bagging predictors," *Mach. Learning*, vol. 24, no. 2, pp. 123 –140, 1996.

[7] A. Chitra and S. Uma, "An ensemble model of multiple classifiers for time series prediction," *Int. J. Comput. Theory and Eng.*, vol. 2, no. 3, pp. 454–458, 2010.

[8] R. B. Cleveland *et al.*: "Seasonal-trend decomposition procedure based on LOESS," *J. Official Stat.*, vol. 6, pp. 3–73, 1990.

[9] W. Constantine and D. Percival. (2015, February 20). *Package 'wmtsa'* [Online]. Available: http://cran.r-project.org/web/packages/wmtsa/

[10] A. M. De Livera *et al.*, "Forecasting time series with complex seasonal patterns using exponential smoothing," *J. American Statistical Assoc.*, vol. 106, no. 496, pp. 1513–1527, 2011.

[11] T. G. Dietterich, "Machine learning research: Four current directions," *Artificial Intell.*, vol. 18, no. 4, pp. 97–136, 1997.

[12] T. G. Dietterich, "An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization," *Mach. Learning*, vol. 40, no. 2, pp. 139–157, 2000.

[13] Y. Freund and R. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. and System Sciences*, vol. 55, no. 1, pp. 119–139, 1997.

[14] J. Gama, I. *et al.*, "A survey on concept drift adaptation," *ACM Comput. Surv.*, vol. 46, no. 4, pp. 1–37, Mar. 2014.

[15] J. Gama *et al.*, "Learning with drift detection," in *Advances in Artificial Intelligence – SBIA 2004*, LNCS 3171, Springer, pp. 286–295, 2004.

[16] J. Gao *et al*., "On appropriate assumptions to mine data streams: Analysis and practice," in *7th IEEE Int. Conf. Data Mining*, 2007, pp. 143–152.

[17] L. Guo *et al.*, "An incremental extreme learning machine for online sequential learning problems," *Neurocomputing*, vol. 128, pp. 50–58, 2014.

[18] H. He, T. Liu, R. Chen, Y. Xiao, and J. Yang, "High frequency short-term demand forecasting model for distribution power grid based on ARIMA," in *2012 IEEE International Conference on Computer Science and Automation Engineering (CSAE)*, 2012, vol. 3, pp. 293–297.

[19] M. Hollander *et al.*, *Nonparametric Statistical Methods*. Hoboken, NJ: J. Wiley & Sons, 2014.

[20] C. C. Holt, "Forecasting trends and seasonals by exponentially weighted moving averages," *Office of Naval Research Memorandum*, vol. 52, 1957.

[21] W. C. Hong, *Intelligent Energy Demand Forecasting*. London: Springer-Verlag, 2013.

[22] K. Hornik *et al.*, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, no. 5, pp. 359–366, 1989.

[23] R. J. Hyndman *et al.* (2015, February 26). *Package 'forecast'* [Online]. Available: http://cran.r-project.org/web/packages/forecast/forecast.pdf

[24] E. Ikonomovska *et al.*, "Learning model trees from evolving data streams," *Data Mining and Knowledge Discovery*, vol. 23, no. 1, pp. 128–168, 2011.

[25] A. Karatzoglou *et al.*, "kernlab - An S4 Package for Kernel Methods in R," *J. Statistical Software*, vol. 11, no. 9, pp. 1–20, 2004.

[26] R. Klinkenberg and T. Joachims, "Detecting Concept Drift with Support Vector Machines," pp. 487–494, Jun. 2000.

[27] J. Z. Kolter and M. A. Maloof, "Dynamic weighted majority: An ensemble method for drifting concepts," *J. Mach. Learning Research*, vol. 8, pp. 2755–2790, 2007.

[28] L. I. Kuncheva and C. J. Whitaker, "Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy," *Mach. Learning*, vol. 51, no. 2, pp. 181–207, 2003.

[29] N. Liu *et al.*, "Short-term forecasting of temperature driven electricity load using time series and neural network model," *J. Clean Energy Technologies*, vol. 2, no. 4, pp. 327–331, 2014.

[30] J. Mendes-Moreira *et al.*, "Ensemble approaches for regression: A survey," *ACM Computing Surveys*, vol. 45, no. 1, Article 10, 2012.

[31] C. J. Merz, "Classification and regression by combining models," Ph.D. dissertation, University of California, USA, 1998.

[32] L. L. Minku, "Online ensemble learning in the presence of concept drift," Ph.D. dissertation, University of Birmingham, UK, 2011.

[33] I. Moghram and S. Rahman, "Analysis and evaluation of five short-term load forecasting techniques," *IEEE Trans. Power Syst.*, vol. 4, no. 4, pp. 1484–1491, 1989.

[34] L. Moreira-Matias *et al.*, "On predicting the taxi-passenger demand: A real-time approach," in *Progress in Artificial Intelligence*, LNCS 8154, Springer, pp. 54–65, 2013.

[35] A. Narasimhamurthy and L. I. Kuncheva, "A framework for generating data to simulate changing environments," in *25th IASTED Int. Multi-Conf. Artificial Intell. and Applicat.*, Innsbruck, Austria, 2007, pp. 384–389.

[36] R. Nau. (2015, February 28). *Moving average and exponential smoothing models* [Online]. Available: http://people.duke.edu/~rnau/411avg.htm

[37] N. C. Oza, and S. Russell, "Experimental comparisons of online and batch versions of bagging and boosting," in *Proc. 7th ACM SIGKDD Int. Conf. Knowl. Disc. and Data Mining*, San Francisco, CA, USA, 2001, pp. 359–364.

[38] N. C. Oza and S. Russell, "Online bagging and boosting," in *IEEE Int. Conf. Syst., Man and Cybern.*, New Jersey, USA, 2005, pp. 2340– 2345.

[39] R. Polikar *et al.*, "Learn++: An incremental learning algorithm for supervised neural networks," *IEEE Trans. Syst., Man, and Cybern. Part C*, vol. 31, no. 4, pp. 497–508, 2001.

[40] S. Reid, *A Review of Heterogeneous Ensemble Methods*. University of Colorado at Boulder: Department of Computer Science, 2007.

[41] S. S. Shapiro and M. B. Wilk, "An analysis of variance test for normality (complete samples)," *Biometrika*, vol. 52, no. 3–4, pp. 591–611, 1965.

[42] W. Shen *et al.*, "Ensemble model for day-ahead electricity demand time series forecasting," in *Proc. 4th Int. Conf. Future Energy Syst.*, Berkeley, CA, USA, 2013, pp. 51–62.

[43] Md. Shiblee *et al.*, "Time series prediction with multilayer perceptron (MLP): A new generalized error based approach," *Advances in Neuro-Information Processing*, LNCS 5507, Springer, pp 37–44, 2009.

[44] J. W. Taylor and P. E. McSharry, "Short-Term Load Forecasting Methods: An Evaluation Based on European Data," *IEEE Trans. Power Syst.*, vol. 22, no. 4, pp. 2213–2219, Nov. 2007.

[45] G. K. F. Tso and K. K. W. Yau, "Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks," *Energy*, vol. 32, no. 9, pp. 1761–1768, 2007.

[46] H. Wang *et al.*, "Mining concept-drifting data streams using ensemble classifiers," in *Proc. 9th ACM Int. Conf. Knowledge Discovery and Data Mining (KDD'03)*, Washington, DC, USA, 2003, pp. 226–235.

[47] J. Wichard and M. Ogorzałek, "Time series prediction with ensemble models," in *Proc. Int. Joined Conf. Neural Networks*, Budapest, Hungary, 2004, pp. 1625–1630.

[48] G. Widmer and M. Kubat, "Learning in the presence of concept drift and hidden contexts," *Mach. Learn.*, vol. 23, no. 1, pp. 69–101, Apr. 1996.

[49] P. R. Winters, "Forecasting sales by exponentially weighted moving averages," *Management Science*, vol. 6, no. 3, pp. 324–342, 1960.

[50] L. Xiao *et al.*, "A combined model based on data pre-analysis and weight coefficients optimization for electrical load forecasting," *Energy*, vol. 82, pp. 524–549, 2015.

[51] W. Xie *et al.*, "Incremental learning with support vector data description," in *2014 22nd Int. Conf. Pattern Recognition (ICPR)*, 2014, pp. 3904–3909.

[52] W. Zang *et al.*, "Comparative study between incremental and ensemble learning on data streams: Case study," *J. Big Data*, vol. 1, no. 1, 2014.

[53] F. Zhang *et al.*, "Conjunction method of wavelet transform-particle swarm optimization-support vector machine for streamflow forecasting," J. Appl. Math., vol. 2014, article ID 910196, 2014.

[54] P. Zhang *et al.*, "Categorizing and mining concept drifting data streams," in *Proc. 14th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, Las Vegas, NV, USA, 2008, pp. 820–821.

[55] P. Zhang *et al.*, "Robust ensemble learning for mining noisy data streams," *Decision Support Syst.*, vol. 50, no. 2, pp. 469–479, 2011.