Automated Categorization of Consumers Based on Consumption Forecast in Smart Grid

Kálmán Tornai and András Oláh

Pázmány Péter Catholic University, Faculty of Information Technology and Bionics
P. O. Box 278, H-1444 Budapest, Hungary
tornai.kalman@itk.ppke.hu, olah.andras@itk.ppke.hu

***Abstract*—Deploying smart meters in Smart Grid systems entails that large amount of measurement data is acquired. By processing and analyzing the data, relevant information can be obtained about the power consumers. One of the most important tasks is determining the main characteristics of the consumer in order to find the best suitable category. This categorization may be essential i) to optimize and estimate the load of transportation grid; ii) to provide the best rate for the consumer as well as the supplier in case of free market of electricity; iii) to forecast and to plan correctly the required amount of energy from power-plants to minimize the difference between the demand and supply. In this paper, a categorization method based on forecasting consumption time series will be introduced to categorize consumers with different consumption patterns with good performance. The performance of the method was subject of analysis, and the new algorithm is proved to be usable in real applications.**

***Keywords*— Consumer categorization; Clustering methods; Classification methods; Time series forecast; Feedforward Neural Network.**

I. INTRODUCTION

Smart power transmission grids are the future of the electricity distribution and transmission systems. In contrast to the present grids, a smart grid can efficiently and adaptively manage the balance between supply and demand in the network [1]. These capabilities root in the intelligent monitoring and measurement system which is an integrated part of the network.

Using smart meters either at the customer or in the power transmission network entails that a huge amount of measurement data is acquired. In order to obtain intrinsic information about the actual status of the network and the behavior of consumer the data have to be processed. As a result, it is required to have sophisticated and innovative algorithms [2]–[4] because of the complexity and the amount of data. The categorization of consumers in Smart Grid systems is an important task to i) optimize and estimate the load of transportation grid; ii) provide the best rate for the consumer as well as the supplier in case of free market of electricity [5]; iii) forecast and to plan correctly the required amount of energy from power-plants to minimize the difference between the demand and supply [1].

Currently the consumer categorization is done either manually or with supervised methods, as a result it requires time and continuous attention. Recently we have dealt with both the automated classification and clustering of consumers, and we have demonstrated that using nonlinear forecast [6] it is possible to i) classify the consumers to known classes; ii) determine correct clusters without any prior information. Nevertheless, the customer categorization still not fully automated as neither the changes of categorization nor the changes of proper classes can be detected.

In this paper, we introduce a new categorization method for automated categorization of the power consumers, and test results on measured power consumption data as well. The method is capable of distinguishing categories and detecting category changes of power consumers with different consumption patterns. The novelty of the method inheres in capability of categorizing several type of time series with high performance as well as it does not require any supervision. Furthermore, as this method is based on the analysis of error levels of time series forecast, in can be combined with existing forecast methods [7].

II. RELATED WORK

In this section existing methods will be briefly summarized.

A. Classification methods

The classifications methods can be distance based, feature based or model based methods. According to [8], in methods designed for classification of time series can be either i) modified or redesigned methods, which are capable of handling sequential data; or ii) extensions, which transforms the sequential data to fit to the existing methods.

The distance based methods provide decision based on the distance between the data elements, so for all new measurement the distance has to be calculated. The class assignment is made using the minimal distance. The performance of the method is highly influenced by the choice of the measure of the distance [9]. In general the Eucledian distance is applied, however special problem may require refined metrics [10]

Feature based methods transform the sequential data into a feature-set. The class assignment is based on the resemblances of the values of the feature-set. The performance of the method is determined by the method of election of the features [11]. In case of time series the most common transformation is

 ISBN: 978-1-61208-4763

wavelet decomposition. To reduce the number of features, kernel methods are also deployed, such as Support Vector Machines.

In case of model based approaches during a training phase for all classes a model is constructed. The new data is classified upon the best fitting model. The most commonly used statistical models are Gaussian, Poisson, Markovian Model, and Hidden Markovian Model [12].

Artificial neural networks do not require prior knowledge about the data. The main advantage of using these methods is that the input noise has less influence on the performance [13].

Forecast based classification method (Class-MBF) exploits the different statistical properties of the power consumption time series acquired from different classes of consumers [6]. The underlying idea of the Class-MBF has the following two components. When an approximator (with capability of estimating forthcoming values of time series using preceding, known values) is trained on time series of a specific class, then i) values of other time series of the same class can be forecasted with low error rate; and ii) values of time series of other class can be forecasted with significantly higher error. As a result, each of the classes can be described by the process in the background, thus an approximator can be trained for each class using representative time series. The resulting algorithmic flow of the scheme is depicted by Figure 1. After

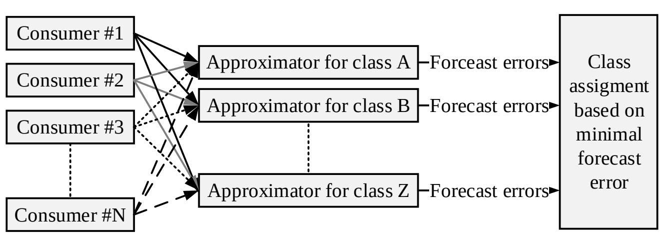

Figure 1. Algorithmic scheme of the classification method

the training phase finished for all approximators, the values of a new sequence with unknown class can be compared to their forecasted values (using each approximator) resulting forecast error sequences. The mean of the forecast error will be used as a decision variable to decide the class where the sequence belongs to. The decision is the result of seeking the approximator which minimizes the forecast error. A training set is used for each approximator to calculate its free parameters with learning algorithms.

B. Clustering methods

The clustering methods can be hierarchical, squared error based, graph theory based, fuzzy clustering, neural networks-based clustering methods [14], [15].

Hierarchical Clustering methods organize data into a hierarchical structure. This approach can be either agglomerative method or divisive method. The first one initializes clusters with one objects and merges clusters based on the distance between them. The latter one proceeds in an opposite way as all the objects belong to one cluster, and the existing clusters are divided into smaller groups [16].

In cases of squared error based methods the sub-optimal partition is found by heuristic algorithms. The k-means algorithm is the most widely used method [17].

Fuzzy Clustering relaxes the constraint which states that a data object belongs only to one cluster. These methods are capable of discovering more sophisticated relations between the clusters and objects [18].

Neural networks-based clustering methods are mainly the self-organizing map (SOM) clusters and adaptive resonance theory networks (ART). The data is represented by artificial neurons and the strengths of connections between the neurons represent the connections between the clusters [19].

The underlying idea of forecast based clustering method is similar to the Class-MBF method. In each iteration of the algorithm, for each existing cluster an approximator is trained using the time series of the cluster. The approximator is used as in the classification method, to estimate the forthcoming value of time series using past values. In each iteration the approximators are evaluated by all time series, as a result, a matrix of error values can be constructed. The values of forecast error matrix can be considered as the distance of a time series from clusters. Then the minimal forecast error value is sought to merge adjacent clusters. If a time series can be forecasted with two or more approximators with low error rate than it is presumed that the two or more clusters can be merged. After the merge is done a new iteration started with the reduced number of clusters. When minimal error rate exceeds any of intra-cluster forecast error rate then the iteration is terminated as the final result is available [20].

III. CONTINUOUS CATEGORIZATION OF POWER CONSUMERS BASED ON FORECAST

In this section the algorithm of the automated method for categorizing the power consumers is introduced. The algorithmic flow of the entire method is summarized by Figure 2, which components are detailed in the following subsections. The used notation is summarized by Table I.

TABLE I. Used notation

Notation	Description
N	Actual number of consumers
$\mathbf{x}^{(i)}, i = 1 \ldots N$	Power consumption time series for consumer i
$n^{(i)}$	Length of $\mathbf{x}^{(i)}$ (number of values in time series i)
M	Actual number of categories
$\mathcal{C}^{(j)}, j = 1 \ldots M$	Category j
$\Gamma^{(j)}()$	Approximator of $\mathcal{C}^{(j)}$
$\tau_{(j)}$	Training set of $\Gamma^{(j)}()$
L	Number of preceding values used by approximator
$\varepsilon_k^{(i)(j)}$	Forecast error value for $x_k^{(i)}$ with $\Gamma^{(j)}()$

A. Approximator

The performance of the method is mainly influenced by the approximator used. Our previous results [6], [20] indicated that the Feedforward Neural Network (FFNN) [21] has the required properties and can be used as approximator. However the clustering method is not restricted to the FFNN approximator, any method (e.g. methods introduced in [7], [22]) which is capable of forecasting power consumption time series can be deployed.

The structure and parameters of the FFNN were determined by experiments. It has been used three hidden layers where the first and second layer implemented with sigmoid non-linear activation function. The third hidden layer and output layer contained artificial neurons with linear function. The Levenberg-Marquardt [23] backpropagation learning function have been applied during training phases.

B. Initial categories

Initially, we suppose that the N consumers are not categorized, thus a clustering has to be executed. Each of the consumers are assigned to its own category as a result $M = N$. For each category the training set is assembled as follows:

$$\tau_{(j)} = \left\{ \left(x_m^{(i)} \dots x_{m+L}^{(i)}; x_{m+L+1}^{(i)} \right) \right\}$$
$$\forall i : \mathbf{x}^{(i)} \in \mathcal{C}^{(j)}, m = 1 \dots \left(n^{(i)} - L - 1 \right) \tag{1}$$

Using the training set, the free parameters ($\mathbf{W}^{(j)}$) of approximators are set by learning algorithms, as a result, an approximator can be used to forecast the forthcoming values based on L preceding values as follows:

$$\hat{x}_k^{(i)(j)} = \Gamma^{(j)} \left(\mathbf{W}^{(j)}, x_{k-1,\dots,k-L}^{(i)} \right). \tag{2}$$

The forecast error value $\varepsilon_k^{(i)(j)}$ is the absolute value of the difference of the estimated and concrete values of the time series i with $\Gamma^{(j)}$. The forecast error of entire time series $\varepsilon^{(i)(j)}$ is calculated as the root mean square error of the individual forecast error values. For every time series and category the forecast error values are also calculated to define the nearness between two categories as follows:

$$e^{(j_1)(j_2)} = \textbf{mean} \left(\varepsilon^{(i),(j_2)}, \forall i : \mathbf{x}^{(i)} \in \mathcal{C}^{(j_1)} \right) \tag{3}$$

where $e^{(j_1)(j_2)}$ denotes the average forecast error of time series of category j_1 with approximator $\Gamma^{(j_2)}()$. As a result, an $\mathbf{E} \in \mathbb{R}^{M \times M}$ matrix of error values can be constructed. Thus two j_1 and j_2 categories can be merged if

$$j_2 = \min_j \left(e^{(j_1)(j)}, j = 1 \dots M, j \neq j_1 \right) \tag{4}$$

and

$$j_1 = \min_j \left(e^{(j_2)(j)}, j = 1 \dots M, j \neq j_2 \right) \tag{5}$$

In order to exclude cases where a time series from a specific category can be forecasted at similar error level with several other approximators, a $\Delta^{(j)}$ threshold value is introduced for each category. The threshold value is calculated as the difference of the mean and deviation of the error values:

$$\Delta^{(j)} = \mu^{(j)} - \sqrt{\frac{1}{K} \sum_{k=1}^{K} \left(e^{(k)(j)} - \mu^{(j)} \right)^2} \tag{6}$$

where $\mu^{(j)}$ is the mean of the error values:

$$\mu^{(j)} = \frac{1}{K} \sum_{k=1}^{K} e^{(k)(j)} \tag{7}$$

We have obtained good performance using (6), however the calculation of threshold value should be fitted to the characteristics of times series of real applications.

After a successful merge the iteration starts over. This recursion is executed while there are categories which can be merged.

When initialization is done the algorithm forks as follows: i) previously not categorized new, upcoming consumers assigned to existing (or new) category; ii) existing category assignments revised periodically in order to follow the changes of consumer's behavior. In next sections the two blocks are discussed.

C. Assigning consumer to existing category

The categorization of time series of a new customer is based on seeking the approximator hich minimizes the forecast error of the values of the time series to be categorized. Formally the new customer with time series $\mathbf{x}^{(\cdot)}$ can be assigned to category j, where the $\varepsilon^{(\cdot)(j)}$ minimal $\forall j = 1 \dots K$. Similarly to (6) a threshold value is applied to consider only significantly low error values during the decision of assigning a consumer to any of existing category. The threshold value is calculated as in (6) with a minor change: instead of values of $\mathbf{E}$ the $\varepsilon^{(\cdot)(j)}$ forecast error values are used.

The case when the minimal $\varepsilon^{(\cdot)(j)}$ value does not significantly differs from the other error values indicates that the new consumer does not belong to any existing category, thus it has to be assigned to a new category $\mathcal{C}^{M+1}$. If a change occurs among the members of a category then the training set of the corresponding approximator is reconstructed and the training is carried out.

D. Category change of a consumer

It is possible that the power consumption behavior of a consumer changes in time. That event should be detected and the assigned category of a consumer should be changed. This can be done with generally two different ways: i) periodically the method is re-initialized; ii) periodically the free parameters of approximators are reset and time series of all categories are re-evaluated. As the execution time of the method is characterized by the number of training phases to be performed, the second method is preferred which has fewer training phases. Furthermore in order to reduce the execution time the training of approximators can be done parallel. Each member of $\mathcal{C}^i$ is evaluated by all $\Gamma^{(j)}()$ and the decision of category assignment is revised using the decision rule introduced previously. After the reassignments are done the approximator of altered (both the old and new) categories have to be adapted by running the learning algorithm. Furthermore, it also may happens that a category turns into empty. In that case the category and its approximator can be disposed.

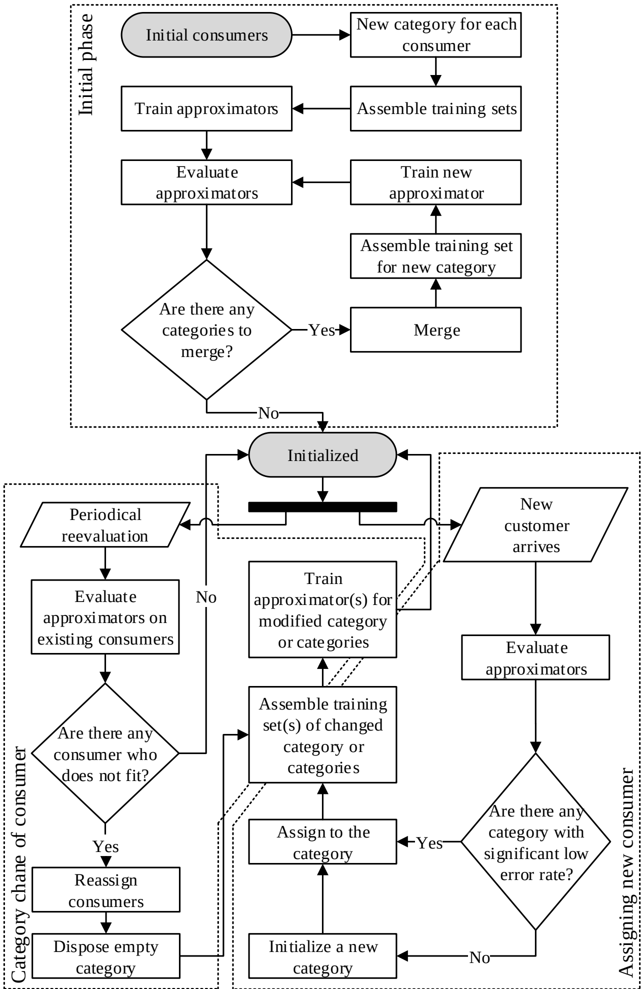

Figure 2. Algorithmic flow of consumer categorization

IV. PERFORMANCE ANALYSIS

In this section the data model, test environment and the performance of the proposed method is introduced. The categorization method were tested by artificially generated time series data and measured power consumption time series as well. The performance of the categorization method is evaluated as follows: for each time series the category which is considered as correct solution is known prior to the test (i.e. the parameters of the model which generated the time series is known, or manual categorization of real data is done). The desired category associations are compared to the results of the new categorization method and the ratio of correct and all decisions is used as performance metric.

A. Data models

Two approaches were used to generate time series: (i) autoregressive (AR) processes for testing and reference purpose only; (ii) a realistic bottom-up model, where the consumption data of consumers were constructed as the sum of statistically independent consumption data of electric appliances.

1) AR model for consumption time series: This model is used to produce time series to investigate the basic capabilities of the method. Exemplary parameters for AR processes (for different categories) are shown by Table II. Using these values, time series with different statistical properties can be artificially generated. The actual parameters for tests were fitted on measured power consumption time series.

TABLE II. Possible parameters for AR processes

	a_1	a_2	a_3	a_4	a_5
Category$_1$	0.3	0.7	0.3	-0.4	-0.2
Category$_2$	0.4	0.2	0.1	0.2	-0.1
Category$_3$	0.9	-0.8	0.5	-0.2	-0.1
Category$_4$	-0.4	0.8	0.3	-0.1	-0.1

2) Markovian model for consumption time series: Using the following model, we have generated close-to-real consumption time series. The consumption of individual appliances is described as a two-state Markovian process. Thus short term time dependencies of the time series are modeled. Formally, the power consumption of a single appliance is generated by a discrete random variable

$$X^{\text{Markovian}} \in \{0, h\}, \tag{8}$$

where the two states (On/Off) are the power off state and operational state where the actual power consumption is h. This model can be easily extended with additional power consumption levels.

The parameters of the Markovian model were fitted on real measurements, which are coming from the Reference Energy Disaggregation Data (REDD) set [24].

The total power consumption of a customer is modeled by the sum of individual Markovian processes. Each category of consumer contains different types of total power consumption data which are constructed from different types of appliances.

Figure 3 depicts the consumption of a single appliance, and also shows the time series generated by summing several independently generated Markovian On/Off processes.

As we have investigated the correctness of the model comparing the autocorrelation values of generated data and measured power consumption time series, we found that the generated values has similar long-term correlations, however the artificial data do not have the characteristics of daily periodicity.

B. Measured consumption data

The real consumption data used for the performance analysis were obtained from a large Central-European electricity distribution company. The measurements were acquired at 150 sites for one year in 2009. Each value in the time series represent aggregated power consumption for one hour. As the actual data is trade secret it has been normalized by the company and personal information was removed as well. The consumption data, for all consumer individually was categorized manually by technicians of the company. The differences of number of customers in categories are minimal. The proposed method was tested on the real data comparing the results of automated categorization to the manual categorization.

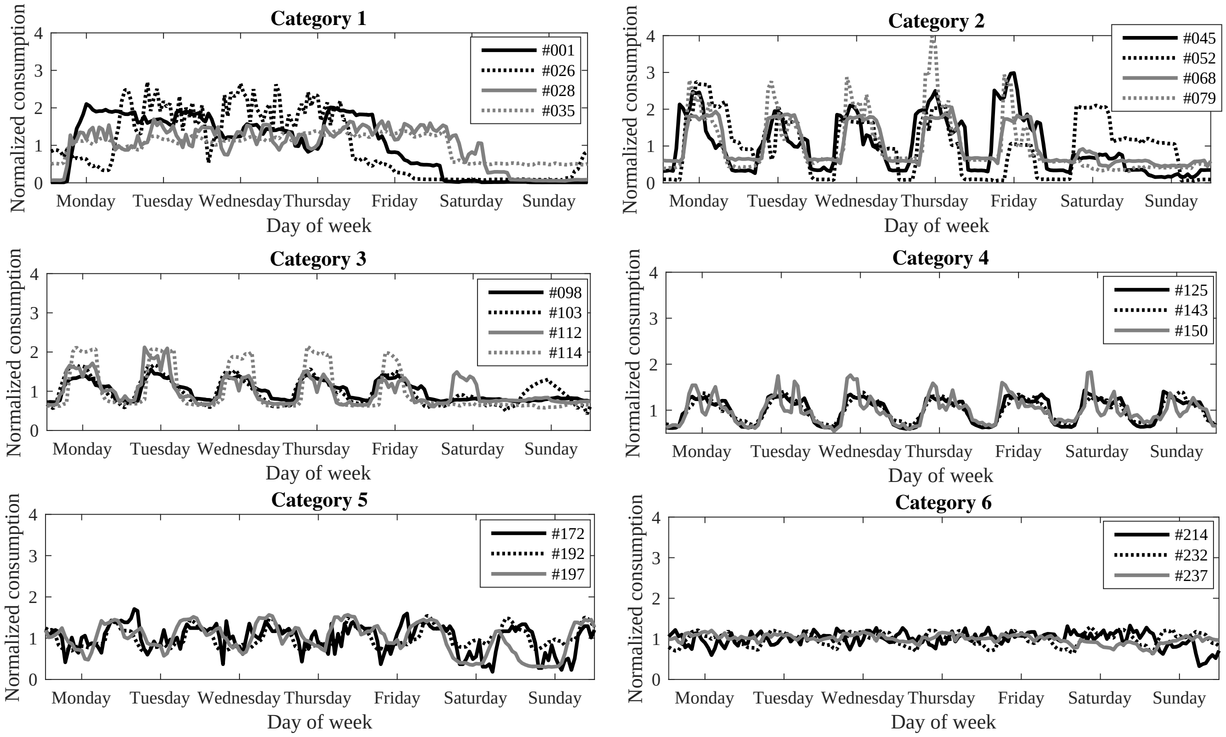

Figure 4. Exemplary power consumption time series from different categories

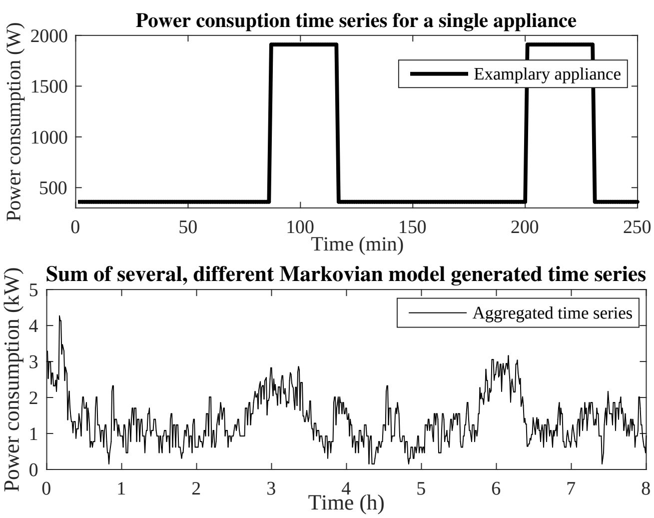

Figure 3. Demonstration of a) single Markovian On/Off model generated time series; and b) aggregated time series

Figure 4 demonstrate some of the different categories, with exemplary consumption time series. The figure also shows the differences of time series in a specific category as well. Category 1 contains power consumption of consumers who operate in weekdays continuously but not on weekends (first category). Consumers of second category have characteristics of repeating daily periods (both weekday only and all day as well), however the third category contains consumers with similar periodicity but the overnight consumption is significantly higher than the other consumers with periodical consumption pattern. Other categories either not have such regularities (e.g. sixth category) or the 'shape' of the daily periodicity differ from time series from other categories (e.g. fourth category). Also the degree of variance of the consumption time series can be the basis of dividing customer into different categories.

As a result, several aspects have to be taken into account, where some of them cannot be described in exact way. Thus the application of automated, self-learning methods are worthy in order to exploit information of hidden properties of time series.

C. Performance results

In this section, we introduce the results on performance analysis carried out on data models described previously. The simulations has been executed in Matlab environment [25]. We have repeated all test for several times to have averaged result in order to eliminate the preferential cases of FFNN.

1) Initial categorization: Figure 5 shows the performance results of initial categorization in case of different number of correct categories and of different data models. (Please note that in case of real consumption data only eight categories of data were available.) The evaluation of resuts is done as follows: the desired categories are matched with the resulted categories based on majority decision. In case of less or more categories, the members of missing or extra categories are considered as wrong category assignments.

For real consumption data the performance of detection capabilities are lower for eight categories than with less

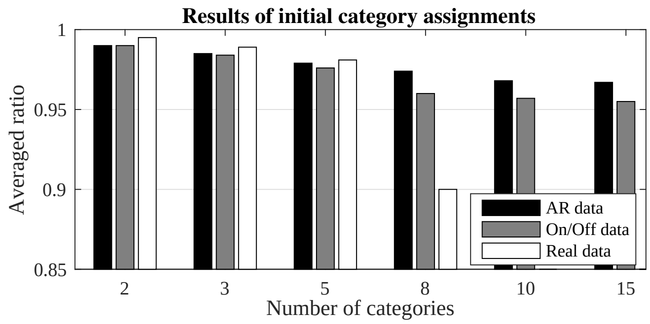

Figure 5. Results of performance test on initial categorization.

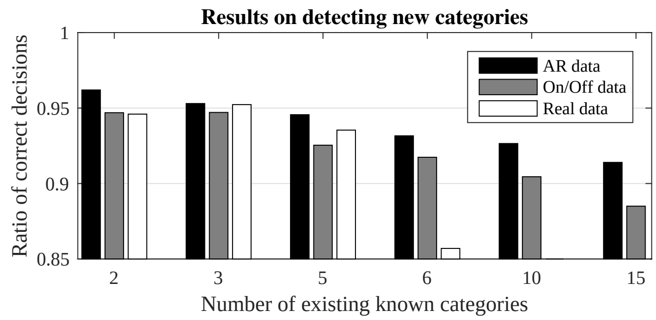

Figure 7. Results of performance test on detecting that for a new consumer a new category may be established

categories. Possible reasons are: i) some of the categories are overlapping; ii) the categorization made by the human experts are not only based on the consumption but some additional meta-information which are unavailable.

2) Category assignments of new consumers: There are two scenarios of category assignment of new consumers: i) the new consumer fits to any of existing categories; ii) new category is required for the new consumer. Both cases were investigated, but separately.

Figure 6 indicates the results, when no new category is required. The category assignment made by the method is considered correct, when the category assignment of new consumer matches to the desired category assignment. As can been seen, the method has high performance in case of each three data models, however in case of real consumption data the same characteristics can be observed as the performance of initial categorization.

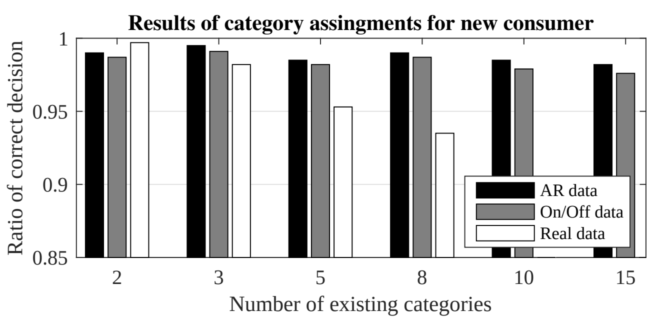

Figure 6. Results of performance test on assigning new consumers to existing categories.

Figure 7 shows the averaged correct decision ratios in case when new category have to be introduced for the new consumer as it does not fit to any of existing category. In case of this test the decision of the method correct, when a new category is introduced for the new consumer. The ratio of correct decision is defined as the proportion of cases when new category is created and number of all test cases. The ratio of correct decisions are generally lower, however it might be improved by introducing more sophisticated or adaptive threshold value (6).

3) Category changes: We have investigated the capabilities of change detection of our method, using hybrid time series which $1 \dots M$ values are from different category from $M + 1 \dots 2M$.

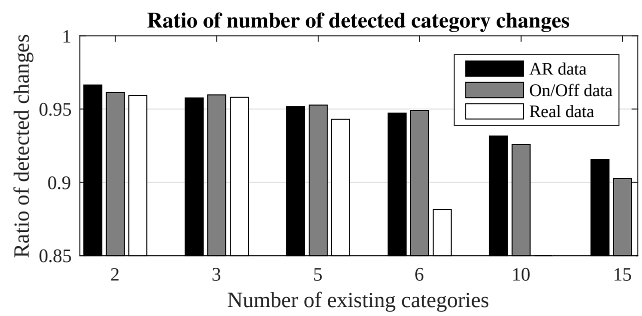

Figure 8. Results of performance test on detecting the consumer category changes

Figure 8 shows the results of correct detection of category change. During tests only the detected category changes were investigated, the false positive decisions were counted neither correct nor wrong decision. Results indicate that the performance is slightly better than in case of categorizing new consumers. This performance also may be improved by using more sophisticated threshold value.

V. CONCLUSION AND FUTURE WORK

In this paper, we introduced a forecast based method to categorize different electricity consumers that have the same first and second order statistics but have different distributions and time dependencies. To forecast the values of time series a nonlinear approximator has been deployed.

The introduced method has been tested on different consumption models including realistic Markovian process based model. Furthermore, we have tested the method on measured power consumption data acquired by a Central European power distribution company. As a final result, it has been shown that feedforward neural network based method is capable of low error rate categorization in real applications on real data.

The performance of the method can be further improved by applying adaptive decision threshold value, or deploying other forecast methods and category decision algorithms as well, such as Radial Basis Function Networks, Deep Learning methods. These possibilities are considered as future work. Also our goal is to extend the method to handle medium term and long term periodicity of the time series.

ACKNOWLEDGMENT

This publication and research has been supported by PPKE KAP 15-084-1.2-ITK Grant. This source of support is gratefully acknowledged.

REFERENCES

[1] F. Rahimi and A. Ipakchi, "Demand response as a market resource under the smart grid paradigm," IEEE Transactions on Smart Grid, vol. 1, no. 1, pp. 82–88, 2010.
[2] M. Last, A. Kandel, and H. Bunke, Data mining in time series databases. World Scientific, 2004.
[3] R. G. Brown, Smoothing, Forecasting and Prediction of Discrete Time Series. Dover Publication Inc., 2004.
[4] S. Rani and G. Sikka, "Recent techniques of clustering of time series data: A survey," International Journal of Computer Applications, vol. 52, no. 15, pp. 1–9, August 2012.
[5] P. M. Noble and T. S. Gruca, "Industrial pricing: Theory and managerial practice," Marketing Science, vol. 18, no. 3, pp. 435–454, 1999.
[6] K. Tornai et al., "Novel consumer classification scheme for smart grids," in Smart Objects, Systems and Technologies (Smart SysTech), 2014 European Conference on, July, 2014, pp. 1–8.
[7] M. Q. Razaa and A. Khosravic, "A review on artificial intelligence based load demand forecasting techniques for smart grid and buildings," Renewable and Sustainable Energy Reviews, vol. 50, pp. 1352—1372, 2015.
[8] T. W. Liao, "Clustering of time series data – a survey," Pattern Recognition, vol. 38, no. 11, pp. 1857–1874, 2005.
[9] Z. Xing, J. Pei, and E. Keogh, "A brief survey on sequence classification," ACM SIGKDD Explorations Newsletter, vol. 12, no. 1, pp. 40—48, 2010.
[10] T. Li, S. Ma, and M. Ogihara, "Wavelet methods in data mining," in Data Mining and Knowledge Discovery Handbook, O. Maimon and L. Rokach, Eds. Springer, 2005, pp. 603–626.
[11] D. Eads, K. Glocer, S. Perkins, and J. Theiler, "Grammar-guided feature extraction for time series classification," in In Proceedings of the 9th Annual Conference on Neural Information Processing Systems, 2005.
[12] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," Proceedings of the IEEE, vol. 77, no. 2, pp. 257–286, 1989.
[13] C. Giles, S. Lawrence, and A. Tsoi, "Noisy time series prediction using recurrent neural networks and grammatical inference," Machine Learning, vol. 44, no. 1, pp. 161–183, 2001.
[14] D. Xu and Y. Tian, "A comprehensive survey of clustering algorithms," Annals of Data Science, vol. 2, no. 2, pp. 165–193, 2015.
[15] S. Rani and G. Sikka, "Recent techniques of clustering of time series data: A survey," International Journal of Computer Applications, vol. 52, no. 15, pp. 1 – 9, 2012.
[16] L. Kaufman and P. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis. Wiley, 1990.
[17] B. Zhang and S. N. Srihari, "Fast k-nearest neighbor classification using cluster-based trees," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 26, no. 4, pp. 525–528, 2004.
[18] H. Izakian, W. Pedrycz, and I. Jamal, "Fuzzy clustering of time series data using dynamic time warping distance," Engineering Applications of Artificial Intelligence, vol. 39, pp. 235 – 244, 2015.
[19] J. Vesanto and E. Alhoniemi, "Clustering of the self-organizing map," IEEE Transactions on Neural Networks, vol. 11, no. 3, pp. 586–600, 2000.
[20] K. Tornai and A. Oláh, "Clustering power consumption data in smart grid," accepted in 1st EAI International Conference on Smart Grid Inspired Future Technologies, 2016.
[21] S. Haykin, Neural Networks, A Comprehensive Foundation, 3rd ed. Pearson, Prentince Hall, 2008.
[22] S. Arora and J. W. Taylor, "Forecasting electricity smart meter data using conditional kernel density estimation," Omega, vol. 59, Part A, pp. 47 – 59, 2016.
[23] J. Nocedal and S. J. Wright, Numerical Optimization, 2nd Edition. Springer, 2006.
[24] J. Z. Kolter and M. J. Johnson, "Redd: A public data set for energy disaggregation research," in SustKDD workshop, San Diego, California, 2011. [Online]. Available: http://redd.csail.mit.edu/kolter-kddsust11.pdf
[25] MATLAB, version 8.6.0 (R2015b). Natick, Massachusetts: The MathWorks Inc., 2015.

 ISBN: 978-1-61208-4763