Data Communication and Analytics for Smart Grid Systems

Arslan Ahmed, Kareem Arab, Zied Bouida, and Mohamed Ibnkahla
Systems and Computer Engineering (SCE) Department,
Carleton University, Ottawa, Canada
{arslanahmed3, kareemarab}@cmail.carleton.ca, {ziedbouida, ibnkahla}@sce.carleton.ca

***Abstract*—With the popularity of smart electrical appliances and home energy management systems, there has been a massive amount of data generated by the power consumption. This data can be beneficial for the utility as it provides the behavior patterns of customers, and thus useful decisions can be made to optimize the load on the grid. In this work, we establish a bidirectional communication system between some homes through the customer agents (CAs), which are installed at home, and the transformer agent (TA) which is installed at the local transformer. Once data is collected at the TA, it is sent to the cloud through LTE. We then use IBM Cloud services to filter and analyze this data to forecast energy consumption and make recommendations to different customers based on their real-time changing behaviors. To this end, we use six different machine learning models predicting the energy consumption: support vector regression (SVR) using linear kernel, SVR using Gaussian kernel, SVR using the polynomial kernel, linear regression, polynomial regression, and feed-forward neural networks (FFNN). To measure the accuracy of these models, we compute three different error metrics, the normalized mean absolute percentage error (NMAPE), the normalized root mean square error (NRMSE), and R^2 also known as the coefficient of determination. Based on these results, we observe that the performance of the forecasting model depends on the dataset properties including the size and variations. For example, while linear and polynomial regressions perform well for small-scale datasets, FFNN gives higher accuracy for large-scale datasets.**

***Index Terms*—Data analytics, energy forecasting, prediction models, smart grid, wireless mesh networks.**

I. Introduction

Internet of Things (IoT) and data analytics are core technologies in smart grid that help the service providers store and process data in real time in the cloud. It also helps users to optimize their energy consumption by negotiating with the utility company. Moreover, statistical techniques for predictive analysis and energy forecasting enable the utility to better understand customers' behavior, keep track and predict the downtime and power failures, act proactively, and negotiate with the end users in spatial and temporal dimensions [1]. However, there are many challenges associated with smart grid systems. Indeed, the big data generated by these systems needs to be stored and managed efficiently to increase the reliability and sustainability of the smart grid infrastructure. Due to limited computational resources, local devices are mostly not useful for this purpose. Another main challenge is how to collect data from homes, send it to the cloud, and establish a bi-directional communication between homes and the cloud. In the remaining part of the introduction, we first present the literature then our contributions to address these challenges.

The authors would like to thank the Ontario Smart Grid Fund and Hydro Ottawa for their support through the GREAT-DR project.

In recent years, a plethora of works has appeared in the literature, addressing different aspects of the communication of smart grids data and the forecasting of the energy consumption [2–5]. We can classify the work related to data analytics in smart grid systems into three categories (i) the first one aims at gaining a deep understanding of the intrinsic properties of energy data including the visualization and filtering (e.g., trends based on temporal consumption or weather) (ii) the second category focuses on forecasting the electricity consumption for a given area and period based on previous readings (iii) and the third set looked at giving recommendations to users based on the available supply of electricity and its usage in the locality by considering the user's consumption behavior. Several authors have studied the problem of customer segmentation in the context of demand response (DR). For instance, a clustering technique using k-means method aiming at determining natural segmentation of customers and identifying their temporal and spatial consumption patterns has been considered in [6]. Similarly, [7] has examined the use of big data analytics to propose a scheme for the selection of right customers for a given level of enrolment in DR program. A cloud-based big data analytics framework in smart grid systems has been considered in [1] by focusing more on data visualization. The accuracy of support vector regression (SVR) models coupled with different clustering methods including k-means, k-medoids-mean, and k-medoids-min on the existing dataset collected over a period of two years at three different locations in Japan has been investigated in [8]. Dong *et al.* have combined convolutional neural network with k-means clustering for short-term load forecasting and have verified their results with different error parameters including the normalized root mean square error (NRMSE) and the normalized mean absolute error (NMAE) [9]. Furthermore, [10] proposes a pattern forecasting ensemble model with iterative prediction procedure and compares the obtained results with five forecasting models using different clustering techniques. The work proposed in [11] discusses the big data management and implementation in smart grids and focuses on the required and available hardware and software tools. In [12], the authors use neural network to compare commercial and residential buildings' data. In [13], Apache Spark has been presented as a unified cluster computing platform for storing and performing big data analytics on smart grids. Recently, Artificial Neural Networks (ANNs) have

been considered in [14] as a popular method for predicting electricity loads thanks to the non-linear and adaptive nature of the prediction model.

In this work, using a two-way communication system between different customer agents and the transformer agent with Raspberry Pi 3 modules, we perform data communication, visualization, filtering, description, and prediction. In this context, we establish a WiFi-based mesh network that we use to send the data from homes to the cloud and to get the response back in terms of recommendations and/or warnings after data is processed. For predictive analysis, we need a big dataset. Since the data collected through our communication system is relatively new, we use an existing dataset from MIT's UMASS laboratory [15], which provides energy consumption values for 114 single-family apartments for the period 2014-2016 and comes with additional weather information including temperature, pressure, and humidity. In our work, we have arbitrary selected 25 apartments for year 2015 from this existing dataset, we have stored this data for each apartment on our CAs, and then used our smart grid communication system to send this information every hour to the cloud through the TA. Therefore, we simulate a real-time data communication providing energy consumption for 25 homes in the cloud.

In light of the above, the main contributions of the proposed work can be summarized as follows:

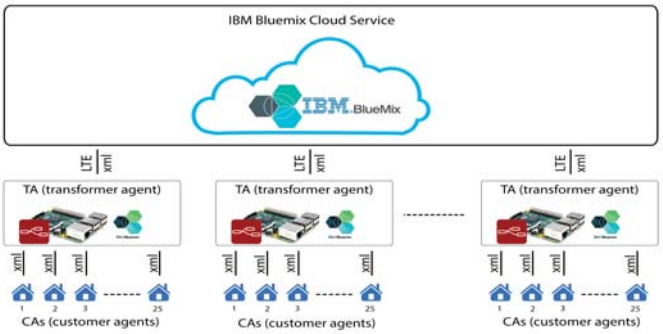
- ⋆ Using our bi-directional communication system between CAs, TA, and the cloud, we establish a real-time data communication process based on a existing real dataset.
- ⋆ Once data is collected from different CAs, we perform data analytics including description and prediction of power consumptions for different apartments.
- ⋆ We use several data prediction methods to predict power usage and compare the accuracy of these techniques.

The remainder of the paper is organized as follows. The details behind the used dataset and the data communication between the CAs, TA, and the cloud are given in Section II. In section III, we define the prediction models and the related metrics to measure their accuracy. In Section IV, we discuss the predicted results and their accuracy. Section V concludes the paper.

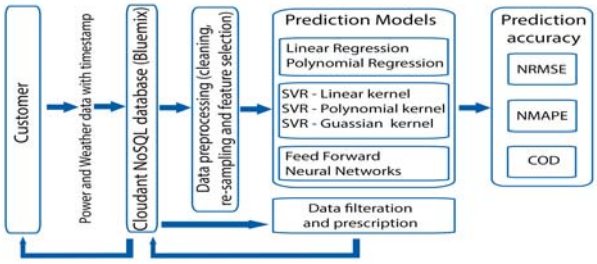
II. Dataset and Data Communication

A. Used Dataset

The dataset used is obtained from UMASS laboratory of MIT [15]. We extract a collection of energy consumption values (measured every 15 minutes in kilowatt hour (kWh)) of 25 single resident apartments collected from January 1 to December 31 2015. This dataset also provides us with weather data including temperature, pressure, humidity, and wind speed for year 2015 sampled every hour. We stored weather and energy data for each apartment on our CAs, and send this after every hour to the cloud so that we see it as a real-time stream of incoming data on the cloud. The system model for data communication from CAs and the TA, and data processing in the cloud is given in Fig. 1.



(a) Data communication from CAs to the cloud via the TA



(b) Data processing in the cloud

Fig. 1. System model for data communication and processing in the cloud

B. Data Communication

In Fig. 1(a), we show the data communication block diagram. The power consumption data is sent every hour from the CA to the cloud, and the recommendations are communicated back to the CA from the cloud via a multi-hop mesh network.

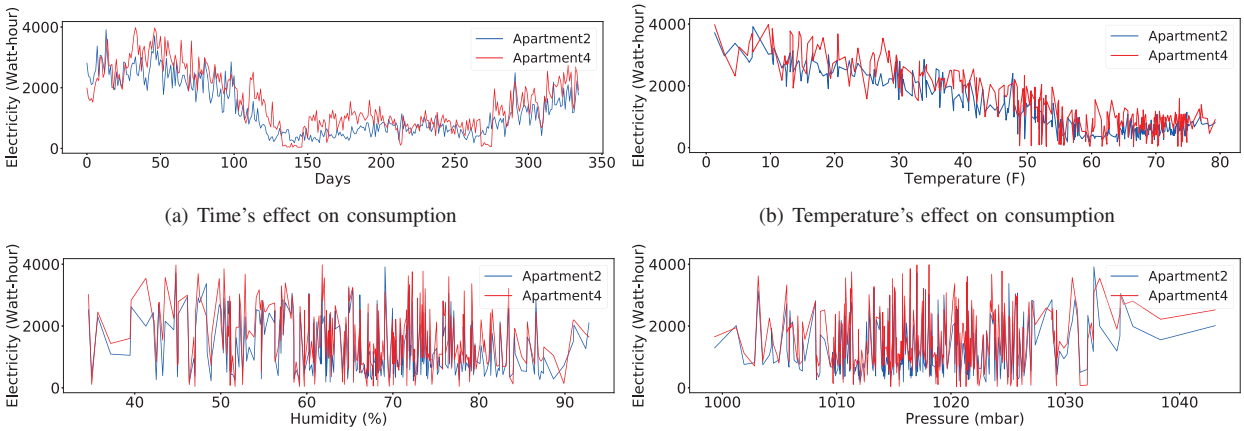
1) Collecting the data from CAs to TA: We use an IP-based wireless mesh network between the CAs and the TA using the Optimized Link State Routing (OLSR) protocol. The messages (energy data) are communicated from the CA (installed at the apartment) to the TA of that neighborhood (installed at the transformer) via multi-hops in xml format in compliance with IEEE 2030.5 profile [16].

2) Collection of data from TAs to cloud: To optimize the energy at the TA, we send all the collected data in real time to the cloud for storage and processing. LTE/cellular networks are used here for this purpose because the TA is located at the pole along with transformer where there is no Wi-Fi.

Once the incoming data arrives to the cloud, we store it on IBM Cloud's Cloudant NoSQL database. We then perform descriptive and predictive analysis on the data using various prediction techniques. The system model for different data analytics steps is shown in the Fig. 1(b) and the details behind data prediction and accuracy are presented in Section III.

C. Data Description and Visualization

The dataset obtained at the cloud from CAs was not fit for processing, i.e. all the apartments did not have the same number of data points, so we pre-processed the data by cleaning and re-sampling. We re-sample the energy at every hour to



(a) Time's effect on consumption

(b) Temperature's effect on consumption

(c) Humidity's effect on consumption

(d) Pressure's effect on consumption

Fig. 2. Energy consumption as a function of different parameters for apartment 2 and 4

make the data points equal to the number of data points in weather data and match with the timings. We then studied the changing behavior of energy consumption of all 25 apartments vs time, temperature, humidity and other parameters including wind speed and pressure. The energy consumption decreases gradually with the increasing temperature, but other weather parameters have negligible effect on it. Based on univariate feature selection, we observed that time and temperature are the only two features that give good results for prediction models. Fig. 2 at the top of the page shows the relationship of energy consumption with different features.

III. Data Analytics

A. Prediction Models

We use several prediction techniques that we explain below with the used notations defined in Table. I.

TABLE I
SYMBOLS AND DESCRIPTIONS

SYMBOL	DESCRIPTION
x	Training/Testing input data
y	Training/Testing output data
$\hat{y}$	Prediction error
Y	Regression model
x_k	Variable dependent on function's order
J	Cost function
$Z(x,y)$	Function output
σ	Adjustable parameter
n	Polynomial order
α, c, m, b_k	Constants
k	Number of apartments equal to 25

1) Linear and Polynomial Regressions:

a) Linear Regression: The linear relationship between variables is mapped with a line equation with a certain slope using the linear relationship $Y = b_0 + b_1\, x_1$. Linear regression is typically used to predict uniform and linear data that does not allow for more complex forecasts.

b) Polynomial Regression: In regression analysis, polynomial regression intakes a predictive path of the nth order as given in the following representation

$$Y = b_0 + b_1\, x_1 + b_2\, x_2^2 + \cdots + b_n\, x_n^n. \tag{1}$$

Similar to linear regression, polynomial regression maps energy readings with slightly additional complexity.

2) Support Vector Regression (SVR):

a) Linear Kernels: The linear kernel, defined in (2), is one of the simplest types of SVR kernels for prediction. It is defined as the dot product of two vectors in a higher dimensional feature space. Linear kernels are extremely useful when dealing with data sets that are linearly separable.

$$Z(x,y) = x^T y + c. \tag{2}$$

b) Polynomial Kernels: Classification problems that are not linearly separable require the use of non-linear kernels. Polynomial Kernels used in SVRs are useful in this case by projecting the features into several higher dimensions, then imposing polynomial classification, and creating a hyperplane to separate the data. These kernels can be given as

$$Z(x,y) = \left(a\, x^T y + c\right)^d. \tag{3}$$

c) RBF or Gaussian Kernels: The RBF kernel, as defined in (4), takes the form of a radial basis or a Gaussian function.

$$Z(x,y) = \exp\left(-\frac{\|x-y\|^2}{2\sigma^2}\right). \tag{4}$$

While RBF kernels are not the best types of classification functions, they are widely implemented thanks to their easily calibrated parameters and reasonable point of reference in terms of comparison between energy consumption predictions in other SVR kernels.

In the above-defined kernels, the predicted value can be obtained by averaging out on the number of samples as follows

$$< Z >= \frac{1}{k}\sum_{n=1}^{k} Z(x,y). \tag{5}$$

3) Feed-Forward Neural Network with Time Series Data: Neural Networks are widely used for many classification and prediction problems in machine learning. Feed-Forward Neural Networks (FFNNs) have been applied to time-series prediction for data sets that contain interval-based data. Applications of FFNNs with time-series range from stock prices to smart city energy consumption as shown in this paper.

This model is based on two input nodes; consuming normalized time and temperature. FFNN uses a sigmoid activation function $f(x) = 1/(1 + e^{-x})$ which is a real-valued differentiable function used also in ANNs to quantify non-linearity in a model. The cost function, $J = \sum 1/2(y-\hat{y})^2$, is then minimized using stochastic gradient decent which avoids unwanted higher-dimensional computational complexity.

B. Data Prediction Scenarios

We forecasted the energy consumption of each apartment using the temperature and time parameters since they have the most influence. Assuming a real-time data, we set the current date to December 1st, 2015 to forecast future data. We use four data prediction frameworks as follows:

- ⋆ **Small-scale prediction:** Use the past two week's data (2015-11-16 to 2015-11-30) to train our models and then use them to forecast the energy consumption for next three days (2015-12-01 to 2015-12-03).
- ⋆ **Medium-scale prediction:** Use past one month's data (2015-11-01 to 2015-11-30) to train our models and then use them to forecast the energy consumption for next week (2015-12-01 to 2015-12-07).
- ⋆ **Large-scale prediction:** Use past four month's data (2015-08-01 to 2015-11-30) to train our models and then use them to forecast the energy consumption for the next month (2015-12-01 to 2015-12-31).
- ⋆ **70-30 ratio prediction:** Use 70% of the total data (2015-01-01 to 2015-12-01) to train our models and then use them to predict the remaining 30% of the dataset.

C. Data Prediction Accuracy

To measure the accuracy of these models, we use three error metrics, the normalized mean absolute percentage error (NMAPE), the normalized root mean square error (NRMSE), and R^2 also known as the coefficient of determination (COD).

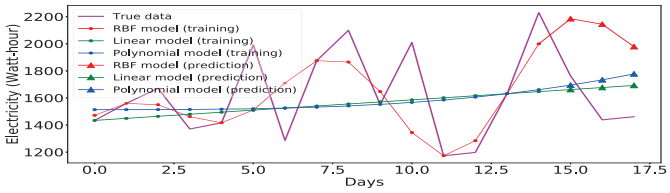
1) NMAPE: This metric measures the accuracy in the form of a percentage, giving an overall comparison between actual and predicted data sets using the absolute values of their differences as given in (6). In this equation, $\hat{y}_i$ is the predicted energy consumption and y_i is the actual energy consumption for a given time at ith sample. The lower the NMAPE, the better the prediction model.

$$\text{NMAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \times 100\%. \tag{6}$$

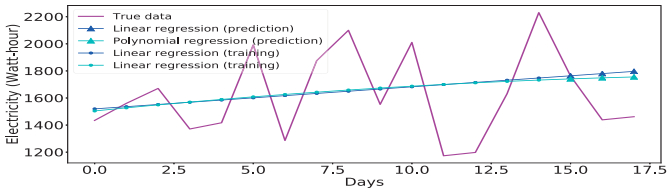
2) NRMSE: The NRMSE metric is the square root of the sample standard deviation of the difference between predicted and observed values (7). Here, y_i is the predicted value of the ith sample and y_i is the actual energy consumption value.

$$\text{NRMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}. \tag{7}$$

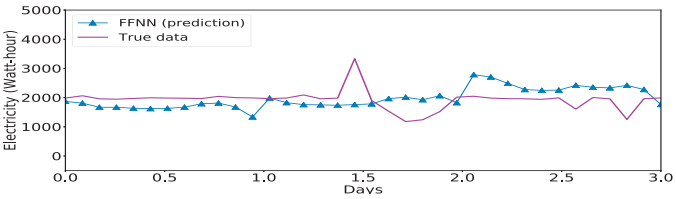
3) R^2 (coefficient of determination): R^2 is a measure of how close the data is to the fitted regression line. If $\hat{y}_i$ is the difference of predicted energy consumption of the ith sample from the actual energy consumption of the ith sample, and y_i is the difference of the mean energy consumption of all true values from the actual energy consumption of the ith sample, then the COD is defined as $\mathrm{R}^2 = \sum {\hat{y}_i}^2 / \sum {y_i}^2$



(a) Small-scale prediction using SVR



(b) Small-scale prediction using linear and polynomial regression



(c) Small-scale Prediction using FFNN

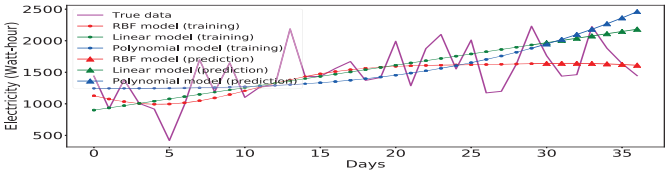
Fig. 3. Small-scale: predicting next three days based on past two weeks

IV. Results

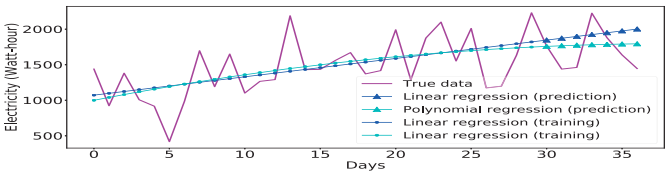
The results for small-scale, medium-scale, large-scale, and 70-30 ratio predictions are shown in Fig. 3, Fig. 4, Fig. 5 and Fig. 6, respectively. For simplicity, we only represent the graphs for energy consumption of Apartment 2. All these results are gathered using time and temperature as predicting features, with all six forecasting models.

Fig. 3 shows the small-scale prediction results. We observe that SVR using linear kernel gives the highest accuracy (87.1%). All other models are also fairly accurate with accuracy ranging between 80%-85%. SVR with Gaussian kernel, however, is only 64% accurate.

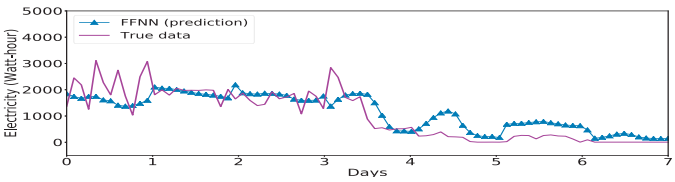
In Fig. 4, we see that polynomial regression of order two gives a good accuracy of 85.3% when used for medium-scale prediction i.e. forecasting energy consumption for the next week, using past two weeks of data. All other models show the



(a) Medium-scale prediction using SVR



(b) Medium-scale prediction using linear and polynomial regression

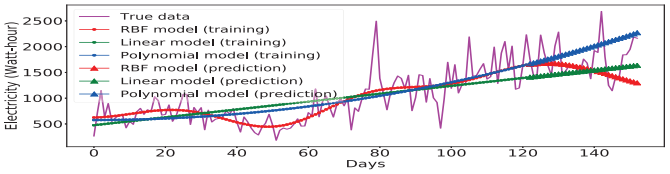


(c) Medium-scale Prediction using FFNN

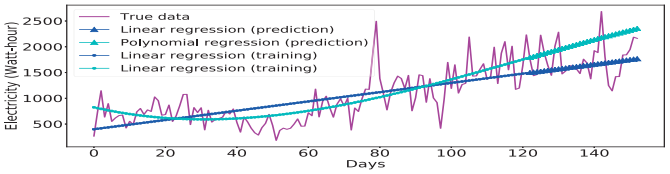
Fig. 4. Medium-scale: predicting next week based on past month

accuracy between 70%-80%. The best model for this scenario is found to be FFNN which gives 86.17% accurate results.

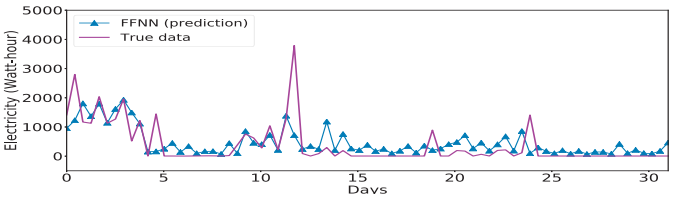
For the large-scale prediction, i.e. forecasting energy consumption for the next month using past four months of data, we see that the FFNN model again surpasses all other models. It gives 92.46% accurate results as shown in Fig. 5, while all other models are between 70%-80% accurate.



(a) Large-scale prediction using SVR

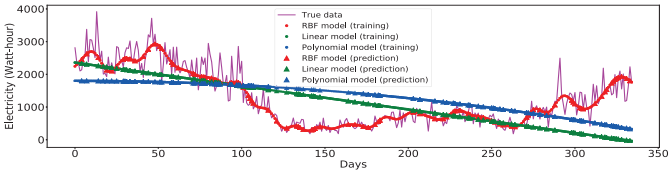


(b) Large-scale prediction using linear and polynomial regression

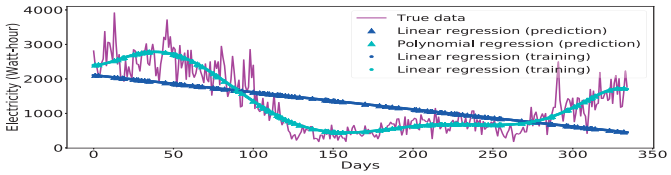


(c) Large-scale Prediction using FFNN

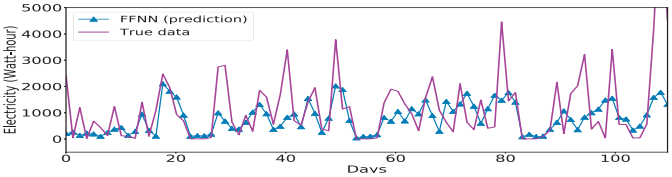
Fig. 5. Large-scale: predicting next month based on past four months



(a) 70-30 ratio prediction using SVR



(b) 70-30 ratio prediction using linear and polynomial regression



(c) 70-30 ratio prediction using FFNN

Fig. 6. 70-30 ratio prediction

We also evaluated our models by randomly choosing 70% of the total data as training data and predicting the rest 30% as shown in Fig. 6. We use the polynomial regression of order seven which gives a high accuracy of 77%. If we increase the order, it becomes overfit and the accuracy decreases. The best observed model is FFNN, which gives an accuracy of 87.5%. SVR with Gaussian kernel gives a very good accuracy of 81.9%, but linear and polynomial with degree two regressions show very poor performance due to high data fluctuations. Increasing the degree of the polynomial improves the results, but the computational complexity becomes very high.

The accuracy of the presented models is summarized in Table II. In what follows, we highlight the performance of each prediction model.

Linear regression model gives the best results for small-scale prediction (82.23% accurate), i.e. when the energy consumption is forecasted based on short training period. As we increase the training period, this model gradually weakens and the accuracy drops for large-scale prediction. The reason may be attributed to more fluctuations in large dataset which cannot be simply represented by a straight line.

For Polynomial Regression, the degree should increase with the data size to cater for more variations. For example, a polynomial of degree two fits well for medium-scale prediction, but we used degree three for large-scale and degree seven for 70-30 ratio predictions. However, we can only increase the degree up to a certain limit after which accuracy starts to drop.

SVR with linear kernel becomes more accurate for smaller training data and always achieves better accuracy than linear kernel. SVR with polynomial kernel with degree one is similar to the linear kernel. If we increase its degree to three, it becomes a better measure for forecasting than the linear kernel.

TABLE II
EVALUATION OF PREDICTION MODELS

Prediction Models	NRMSE (%): The lower, the better				NMAPE (%): The lower, the better				R^2 (COD) Maximum 1: The higher, the better			
	Small Scale	Medium Scale	Large Scale	70-30 ratio	Small Scale	Medium Scale	Large Scale	70-30 ratio	Small Scale	Medium Scale	Large Scale	70-30 ratio
Linear Regression	17.761	21.23	23.382	51.01	15.6	20.15	14.6	83.97	0.14	0.29	0.45	0.28
Polynomial Regression	15.882	16.647	22.737	23.1	14.29	14.7	27.54	26.1	0.14	0.3	0.57	0.82
SVR linear kernel	11.5	16.511	21.603	53.97	12.69	27.36	15.9	55.23	0.13	0.24	0.44	0.23
SVR Polynomial	16.421	35.196	24.4	62.2	15.32	33.1	22.52	72.58	0.12	0.19	0.54	0.039
SVR Gaussian model	9.491	16.23	24.1	23.21	36.13	12.17	16.3	18.185	0.37	0.46	0.64	0.87
FFNN	19.608	13.831	7.536	12.489	16.335	8.43	11.54	10.43	0.18	0.59	0.49	0.148

If we go even higher, it becomes over-fit model which drops the accuracy and also makes the program too complex.

SVR with Gaussian kernel is not a very accurate metric for small-scale prediction, however, its accuracy increases with the size of the dataset. The behavior of this model is very sensitive to the tuning parameter γ. As the dataset increased in size, we used smaller γ to achieve better accuracy. We used 0.001 in large-scale, 0.01 in medium-scale, and 0.1 in small-scale.

FFNN gives better accuracy in all scenarios except for small-scale prediction, where it is around 80.4% accurate. Its high accuracy can be attributed to its ability of mapping highly complex trends and patterns in dimensionally intensive datasets (i.e. the use of five hidden layers and three input layers). This is achieved using the sigmoid activation function that quantifies any non-linearities in a model. As a result, we observe 86.1% accuracy for medium-scale prediction and 92.46% accuracy for large-scale prediction scenarios.

V. CONCLUSION

We presented in this paper a smart grid including data communication, storage, processing, analytics, and communicating recommendations and warnings back to both homes and utilities. In this context, we proposed our model for data visualization based on different filters to study the consumer behavior at different times and made recommendations to them to optimize the grid's load. We also conducted performance analysis by comparing several forecasting models and found out that the performance of these models depends on the considered scenarios as discussed in the results' section.

After analyzing the performances of all models, it became apparent that the best prediction model for large datasets is FFNN. This is due to its complex non-linear mapping ability allowing to learn the factors that contribute to many of the semi-random fluctuations occurring in the model. However, for small datasets, linear and polynomial regressions give fine accuracy if tuned adequately.

As a future extension, we aim at strengthening the security of our communication system. Moreover, for more optimized resource allocation, we aim at coupling prescription analysis with the description and prediction for demand response.

REFERENCES

[1] A. A. Munshi and Y. A. I. Mohamed, "Cloud-based visual analytics for smart grids big data," in *2016 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, Sept 2016, pp. 1–5.

[2] P. D. Diamantoulakis, V. M. Kapinas, and G. K. Karagiannidis, "Big data analytics for dynamic energy management in smart grids," *Big Data Res.*, vol. 2, no. 3, pp. 94–101, Sept 2015. [Online]. Available: http://dx.doi.org/10.1016/j.bdr.2015.03.003

[3] A. A. Munshi and A.-R. M. Yasser, "Big data framework for analytics in smart grids," *Electric Power Systems Research*, vol. 151, pp. 369–380, Oct 2017.

[4] Y. Simmhan, S. Aman, A. Kumbhare, R. Liu, S. Stevens, Q. Zhou, and V. Prasanna, "Cloud-based software platform for big data analytics in smart grids," *Computing in Science & Engineering*, vol. 15, no. 4, pp. 38–47, 2013.

[5] F. L. Quilumba, W.-J. Lee, H. Huang, D. Y. Wang, and R. L. Szabados, "Using smart meter data to improve the accuracy of intraday load forecasting considering customer behavior similarities," *IEEE Transactions on Smart Grid*, vol. 6, no. 2, pp. 911–918, Mar 2015.

[6] J. Kwac, J. Flora, and R. Rajagopal, "Household energy consumption segmentation using hourly data," *IEEE Transactions on Smart Grid*, vol. 5, no. 1, pp. 420–430, 2014.

[7] J. Kwac and R. Rajagopal, "Demand response targeting using big data analytics," in *2013 IEEE International Conference on Big Data*, Santa Clara, California, 2013, pp. 683–690.

[8] C.-Y. Kuo, M.-F. Lee, C.-L. Fu, Y.-H. Ho, and L.-J. Chen, "An in-depth study of forecasting household electricity demand using realistic datasets," in *Proceedings of the 5th international conference on Future energy systems*. ACM, 2014, pp. 145–155.

[9] X. Dong, L. Qian, and L. Huang, "Short-term load forecasting in smart grid: A combined CNN and K-means clustering approach," in *Big Data and Smart Computing (BigComp), 2017 IEEE International Conference on*. IEEE, 2017, pp. 119–125.

[10] W. Shen, V. Babushkin, Z. Aung, and W. L. Woon, "An ensemble model for day-ahead electricity demand time series forecasting," in *Proceedings of the fourth international conference on Future energy systems*. ACM, Jan. 2013, pp. 51–62.

[11] H. Daki, A. El Hannani, A. Aqqal, A. Haidine, and A. Dahbi, "Big data management in smart grid: concepts, requirements and implementation," *Journal of Big Data*, vol. 4, no. 1, p. 13, 2017.

[12] R. E. Edwards, J. New, and L. E. Parker, "Predicting future hourly residential electrical consumption: A machine learning case study," *Energy and Buildings*, vol. 49, pp. 591–603, 2012.

[13] R. Shyam, B. G. HB, S. Kumar, P. Poornachandran, and K. Soman, "Apache spark a big data analytics platform for smart grid," *Procedia Technology*, vol. 21, pp. 171–178, 2015.

[14] J. Zheng, C. Xu, Z. Zhang, and X. Li, "Electric load forecasting in smart grids using long-short-term-memory based recurrent neural network," in *2017 51st Annual Conference on Information Sciences and Systems (CISS)*, Baltimore, Maryland, 2017, pp. 1–6.

[15] "Umasstracerepository. 2017. Smart data set for sustainability." [Online]. Available: http://traces.cs.umass.edu/index.php/Smart/Smart

[16] IEEE, "IEEE adoption of smart energy profile 2.0 application protocol standard," *IEEE Std 2030.5-2013*, pp. 1–348, Nov 2013.