Power Forecasting and Anomaly Detection with MIDAS IoT-based Sensor

Bharath Muppasani [1], C.J. Anand [2], Chinmayi Appajigowda [2], Biplav Srivastava [1], Lokesh Johri [2],

[1] AI Institute, University of South Carolina, Columbia, South Carolina, USA
[2] Tantiv4, San Jose, California, USA
bharath@email.sc.edu, anand@tantiv4.com, chinmayi@tantiv4.com, biplav.s@sc.edu, lokesh@tantiv4.com

Abstract

Time series forecasting is an important aspect of machine learning since it has a wide range of applications, including forecasting power usage, traffic, and air quality, to name a few. For all active participants in the electrical market, load forecasting has become an important aspect of planning and operation. The penalty costs for under or over contracting electricity have increased dramatically as a result of the new market structure, making prediction error minimization more crucial than ever. Forecasting end user load for power purchase has become critical in the context of electricity suppliers. Additionally, abnormal power usage detection can be helpful in detecting excessive unwanted power usage and sometimes hinting on the device failures. Our research focuses on forecasting energy consumption, which will serve as an alerting system in the event of abnormal power consumption conditions, as well as assisting users in planning their power usage. We have used the continuous power data being recorded by the installed MiDAS sensor for our research.

1 Introduction

The growth in the deployment of Internet of Things (IoT) sensors across different industries has opened several opportunities for the economy. One of them is the collection of IoT data that companies can use to build smarter solutions. These IoT sensors, while performing their assigned tasks, can also help collect data from real-world objects or devices for analysis and gain intelligence to improve the latter's capabilities. One such prominent application is the collection and analysis of Electricity Consumption Data (ECD) for a robust and reliable energy management system at any organization.

There is a rich body of work on data-based energy management but much of it is in forecasting and with limited access to data. Accurate forecasting of energy consumption has the potential to save large utility bills. These savings can be realised once the forecasted load knowledge is used to control operations and decisions of the power utility. Mainly in the power systems, the economy of operations and control of operations are sensitive to forecasting errors. Existing forecasting methods can be characterized as conventional [1] or statistical [20] and they focus on short-term load forecasting. These approaches use the trends in the power data to develop a suitable model and use the model for forecasting the future load [1]. Based on the characteristics of the time series data, there are different statistical models popularly used. Some of them are auto-regressive (AR), auto-regressive moving-average (ARMA), auto-regressive integrated moving-average (ARIMA). In [11], authors used Gaussian features of the load to determine the model of ARMA dynamically. Complex models can be used to make high-precision predictions, but this is challenging given the high complexity, irregularity, randomness, and non-linearity of real world data. Machine learning techniques can be used to create nonlinear prediction models based on a significant amount of historical data. Typical machine learning models include support vector machines (SVM) [19] or kernel based classification, artificial neural network (ANN) [7], tree-based ensemble methods such as gradient-boosted regression or decision trees [13] or long short-term memory units (LSTM) [3, 4, 5] or transformers [22]. Authors in [17] focused on providing rule based explanations for a particular forecast, considering the global forecasting model as a black-box model trained across multi-variate time series.

More recent work in energy management has focused on Non-Intrusive Load Monitoring (NILM) [2], [9] where, from the aggregate power data, the aim is to dis-aggregate and estimate individual load. This technique is especially appealing to the industry due to its low cost and easy implementation.

2 Related Work

In the following section we discuss about the literature review of the related work.

2.1 Anomaly Detection Methods

Anomalies or outliers of a data is defined by Hawkins as "an observation which deviates so significantly from other observations as to arouse suspicion that it was generated by a different mechanism" [10]. Outlier detection has been studied in variety of application domains such as credit card fraud detection [18], intrusion detection in cyber security [14], or fault diagnosis in industry [16].

Authors in [21] has proposed a LSTM based model for anomaly detection in which the LSTM model is trained to forecast the power consumption data. Later this LSTM model is used to compare the original reading and the forecasted value to decide on the anomalous state of a reading. Authors in [12] has provided a visual representation of the anomalous scores to direct analysts to anomalous points using a clustering-based anomaly detection method.

2.2 Time Series Forecasting Methods

In recent years, several research works have been proposed in the field of time series data forecasting. Some of the works focused on household level power consumption data [8] from the data recorded by the real world sensors. While the works in [3], [4] and [5] focused on state/union territory level power consumption data sampled at 15 minutes interval.

The works in [3] and [4] considered Chandigarh UT data and analysis of the data was done by clustering the data into 3 major groups - summer, rainy, winter seasons. In [3] authors considered 4 days (monday, wednesday, friday, sunday) for each seasons and build 12 LSTM models. In this work the authors has used Emperical Mode Decomposition (EMD) to decompose the signal into a set of Intrinsic Mode Functions (IMFs). The authors suggest that their work has achieved a percentage error that varies from 5% to 8%. In [4] authors considered 3 days (monday - start of the week, wednesday - peak demand day of the week, sunday - weekend) for each seasons and build 9 LSTM models. On an average the LSTM models developed are shown to have a percentage error of 7% to 10% considering all the experimental settings.

The authors in [5] considered the time series data of Himachal Pradesh(HP) by clustering the Electricity Consumption Data (ECD) into 3 groups. Daily level analysis was done by considering 3 days (monday, wednesday, sunday) per season The authors have used Variance Mode Decomposition (VMD) to decompose the ECD into 8 modes/sub-series. These sub-series are fed to a autoencoder model to extract the latent space representation (k optimal features) of the time series. These optimal features along with the original time series data are used to train a LSTM-MIMO forecasting model. In this case they built a total of 9*k LSTM models. Authors also tried to chose the window size of the LSTM model dynamically by performing intracorrelation with FFT convolution. The forecasted time series for k optimal signals are passed through the trained decoder module. The final forecasted time series signal is given as a summation of the decoded signals for an individual day in each cluster. The authors state that the proposed architecture using the VMD method provides a Mean Absolute Percentage Error (MAPE) of 3.04%.

3 Data Collection and Usage

We now discuss how the data is collected by the MiDAS device installed by Tantiv4.

MiDAS IoT Sensors: The MiDAS IOT sensor measures phase voltages (three-phase), phase currents (three-phase), neutral current, power factor (three-phase), active power (three-phase), apparent power (three-phase), reactive power (three-phase), frequency and phase (three-phase) values every 300ms. The device is also capable of collecting three-phases of current and voltage harmonics data from 2 to 32 harmonic levels along with total harmonic distortion for each phase of current and voltage every 500ms. It interfaces using current sensors with a clamp format for easy installation. Voltage sensors are internal to the device. Field terminals can take up to 1.5 sq. mm cables.

Data Collection: To demonstrate the generality of the SIP problem, we are making data-sets available from different economic industries: Manufacturing, Hospital & Educational institutions. The dataset obtained from MiDAS device is of two forms: Electricity Consumption Data and Harmonics Data. The harmonics information obtained by the MiDAS sensor can be used for further analysis of electric device performance.

Electricity Consumption Dataset: contains 28 different features of electricity consumption data. The features are: Current (IA, IB, IC INCURRENT), Voltage (VA, VB, VC), Power Factor (PFA, PFB, PFC, PFT), Phase (PhaseA, PhaseB, PhaseC), Active Power (ActivePA, ActivePB, ActivePC, ActivePT), Reactive Power (ReactivePA, ReactivePB, ReactivePC, ReactivePT), Apparent Power (ApparentPA, ApparentPB, ApparentPC, ApparentPT), Frequency (FREQ), and Time Stamp.

Harmonics Dataset: contains 193 different harmonics data features. The features are: Current (AI_HR[2 to 32], AI_THD, BI_HR[2 to 32], BI_THD, CI_HR[2 to 32], CI_THD), Voltage (AV_HR[2 to 32], AV_THD, BV_HR[2 to 32], BV_THD, CV_HR[2 to 32], CV_THD) and Time Stamp.

- **India-1:** The sensor is connected to a single Laser cutting machine with power readings between 2 to 25 Amps. The machine laser cuts stainless, carbon steel, aluminum, brass, titanium, and more.
- **India-2:** The sensor is connected to a hospital's main incoming supply and the power load includes machines for CT scan, ECG, EEG, Digital Xray, USG, and C-ARM diagnostic services. The power consumption fluctuates between 35 to 110 Amps.
- **India-3:** The sensor is connected to a single device lathe machine which is used to perform various operations such as cutting, sanding, knurling, drilling, deformation, facing, and turning, with tools that are applied to the work piece to create an object with symmetry about that axis with power consumption between 2 to 40 Amps.
- **India-4:** The sensor is connected to the main supply of a manufacturing plant which comprises of devices such as multiple CNC (computer numerically controlled) machines, Lathe machines, Lifts etc. and the power consumption is between 15 to 60 Amps.
- **India-5:** The sensor is connected to a single CNC machine which is used for testing roughness, waviness, flatness, curvature etc of objects and the power consumption is between 3 to 25 Amps.
- **India-6:** The sensor is connected to the main supply of design & drafting division, comprising of less than

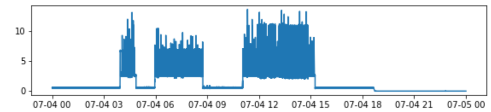
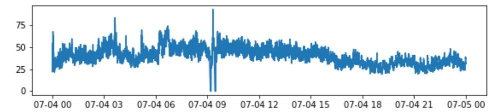
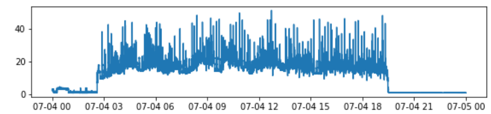
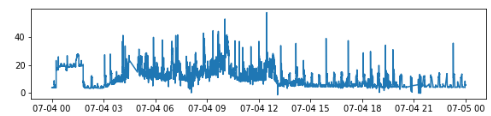
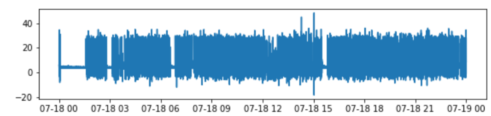
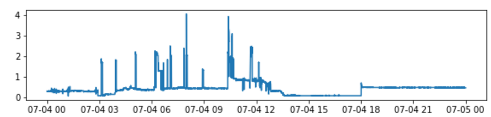
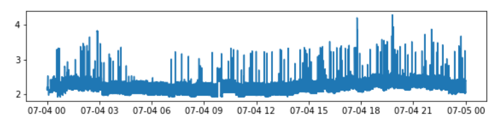
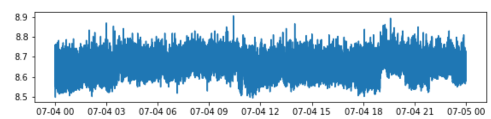
Location	Industry	Load Illustration	Load Figures (ActivePT vs datetime)*
India-1	Manufacturing	Laser Cutting Machine	
India-2	Hospital	Main Supply	
India-3	Manufacturing	Lathe Machine	
India-4	Manufacturing	Main Supply	
India-5	Manufacturing	CNC Machine	
India-6	Manufacturing	Main Supply	
USA-1	Education	AI/ML Lab	
USA-2	Education	Data center	

Table 1: Characteristics of data collected locations; *illustrative power usage for a day - Jul 04 (except India-5: Jul 18, 2022).

10 employees, equipped with dedicated plotters, jumbo photo copiers, blue printer, spiral binder etc. and the power consumption is between 0.5 to 10 Amps.

- **USA-1:** The sensor is connected to the main supply of a research center at a University with 10-30 daily users who bring their devices or use servers.
- **USA-2:** The sensor is connected to the main supply of a server room at a Computer Science department of a University being used for various computational loads.

Data Cleaning and Augmentation: Many real world sensors face with the problem of missing data. The data can be missing due to various reasons such as power outage, sensor failure, sensor maintenance work or the sensor is not connected to the network. To solve this issue of missing data, we filled in a timestamp's missing data by taking the mean of two previously and subsequently available occurrences of the same timestamp. For example, if the data is missing at 02-01-2022 09:00:00, we looped back(forward) to the previous(future available) days where the data is available at 09:00:00 timestamp and considered mean of these observations. The main idea of considering the same timestamp is that, given a working environment, the characteristics of the power consumption data for a given timestamp would be similar across the working/non-working days.

4 Approaches

In the following section we discuss about the different approaches used for Time series forecasting.

4.1 ARIMA

One of the most famous and commonly used statistical models for time-series forecasting is the Auto Regressive Integrated Moving Average (ARIMA) model. It's a category of statistical techniques that captures the standard temporal dependencies that are particular to time series data.

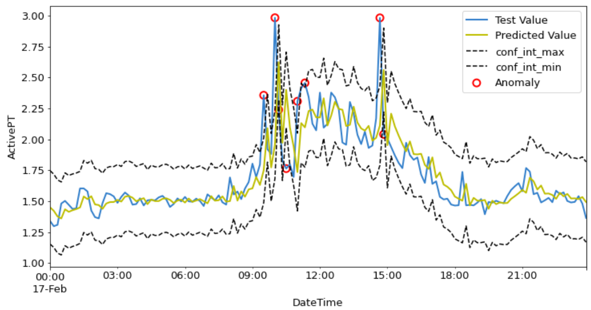


Figure 1: ARIMA model as anomaly detector performed on USA-1 location

4.2 LSTM

Statistical forecasting approaches, such as the ARIMA model mentioned earlier, are popular and reliable, but they lack the general generizability that memory-based models may provide. The Long Short Term Memory (LSTM) Network is a kind of Recurrent Neural Network (RNN) that can deal with long-term dependencies in a dataset. Considering the relationships between the time stamps while working with the Time-Series dataset is essential to effective prediction of future time stamp values.

4.3 Gradient Boost Regression for Time Series

Gradient Boosting is a machine learning approach that can be used to solve problems in classification and regression. It is based on the idea that a group of weak learners may produce a more accurate predictor when they work together. Gradient boosting works by gradually developing smaller prediction models, each of which attempts to predict the error left behind by the preceding model. As a result, the algorithm has a tendency towards overfitting.

5 Evaluation

The following section will discuss the evaluation and results of the proposed methods.

ARIMA

The ARIMA model is chosen using the pmdarima library's auto-arima function. The ARIMA model with the lowest AIC values is returned by this function. The ADF statistical test determines if the dataset is stationary or trend-stationary. For a given time series data, this test produces a p-value. We can determine the dataset's stationariness based on this p-value, which in turn determines whether or not to utilize the differencing parameter in ARIMA model. The model parameters and RMSE value for one of our test datasets are shown in Table 1. We have used ARIMA model as a anomaly detector by comparing the 95% confidence intreval level of forecasted value against the actual value. The data points exceeding this level are categorised as anomalies. The result of the ARIMA model as snomaly detector is shown in Figure 1(a).

Model	Parameters of the Model	RMSE
ARIMA	SARIMA p=0, d=1, q=2	0.059
LSTM	1 layer with 200 units of LSTM	0.134
GBRT	150 units	0.262

Table 2: Result comparison among different models

LSTM

LSTM architecture selection is influenced by a number of factors. On the same train and test data splits, we experimented with several model settings and compared the performance of various architectures. We devised an optimal model equation that takes into account the RMSE values, the number of epochs, and the length of the train dataset in order to select the best model with the best performance. All three parameters in the equation should be minimized for a model to be optimum, even when deployed on an edge device. As a result, the model with the lowest value in the optimal equation was chosen as the best. The details of the model selection, experimental settings and the results obtained are discussed in the next section.

GBRT

Authors in [6] tried converting the time series forecasting challenge into a window-based regression problem, similar to deep neural network (DNN) models. Furthermore, authors have feature-engineered the GBRT model's input and output structures so that the target values are concatenated with external features for each training window and then flattened to generate one input instance for a multi-output GBRT model. We used their GBRT model implementation for electricity dataset on our MiDAS power data with some changes in the input and output shapes. The details of the experimental settings and the results obtained are discussed in the further sections.

6 From Point to Window Prediction

While the point forecast is useful for detecting anomalies and activating an alarm system, multiple output prediction might be useful for giving the user a visual representation of forecasted power usage for the day, week, or a month. This aids in the proper planning of power usage.

LSTM

We performed Neural Architecture Exploration to choose the appropriate model for deployment. We explored 3 different settings: 1) LSTM with 1 layer with single layer, 2) LSTM with 2 layers with equal units in both the layers of LSTM, and 3) LSTM with 2 layers with varying number of units in both the layers of LSTM. Each of these settings was evaluated on two different depths of training and testing data: More days into the past and few days into the past. Each of these settings was evaluated on a different number of epochs. All these different parameter settings while choosing LSTM architecture resulted in varying root means square error (RMSE) values and also the training time which is directly dependent on the number of epochs, days to train, and the width of the neural network. To choose a desired

layers	units	epochs	data	Training Time	RMSE	eq_result
1	[100]	20	50 Days in Past	52.12	0.108	1.508
1	[100]	20	20 Days in Past	27.46	0.255	1.155
1	[100]	50	50 Days in Past	125.46	0.101	2.101
1	[100]	50	20 Days in Past	67.73	0.11	1.61
1	[200]	20	50 Days in Past	85.15	0.094	1.494
1	[200]	20	20 Days in Past	47.14	0.248	1.148
1	[200]	50	50 Days in Past	184.1	0.098	2.098
1	[200]	50	20 Days in Past	95.46	0.129	1.629
2	[100, 100]	20	50 Days in Past	93.4	0.5	1.9
2	[100, 100]	20	20 Days in Past	49	0.307	1.207
2	[100, 100]	50	50 Days in Past	240.08	0.104	2.104
2	[100, 100]	50	20 Days in Past	124.93	0.158	1.658
2	[200, 200]	20	50 Days in Past	131.85	0.263	1.663
2	[200, 200]	20	20 Days in Past	65.31	0.576	1.476
2	[200, 200]	50	50 Days in Past	322.02	0.288	2.288
2	[200, 200]	50	20 Days in Past	203.38	0.443	1.943
2	[100, 50]	20	50 Days in Past	59.09	0.322	1.722
2	[100, 50]	20	20 Days in Past	37.46	0.529	1.429
2	[100, 50]	50	50 Days in Past	156.91	0.448	2.448
2	[100, 50]	50	20 Days in Past	77.88	0.555	2.055
2	[200, 100]	20	50 Days in Past	126.6	0.327	1.727
2	[200, 100]	20	20 Days in Past	62.33	0.421	1.321
2	[200, 100]	50	50 Days in Past	394.37	0.486	2.486
2	[200, 100]	50	20 Days in Past	259.01	0.478	1.978

Table 3: LSTM Neural Architecture Exploration performed on USA-2 location

model, weights were assigned to each of these parameters and a mathematical equation was devised to determine the best model.

$$val = W1*RMSE+W2*dataVal+W3*epochVal \quad (1)$$

The equation 1 shows the weight equation which considers RMSE, number of data points in training, and number of epochs. The weights were assigned to each of these parameters in order to determine the best model. Currently we have given equal weights for all the parameters. The model resulting in the least value for equation 1 was chosen as the best model. The results for LSTM neural architecture exploration results for different settings is shown in Table 3.

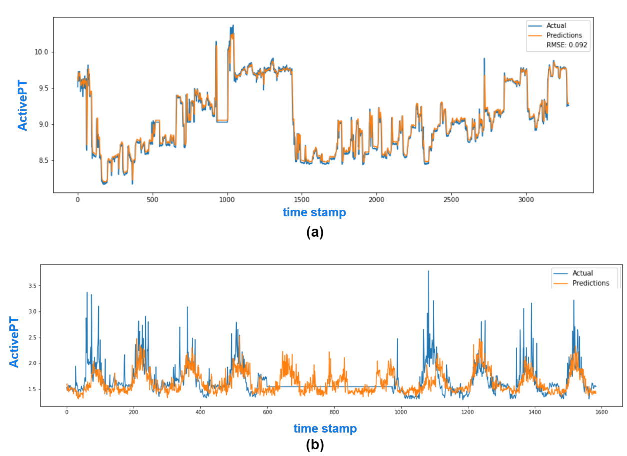


Figure 2: (a) LSTM-MIMO prediction result performed on USA-2 location (b) GBRT prediction result performed on USA-1 location

With the help of sliding window technique we have built a Multi-Input Multi-Output (MIMO) based LSTM model. The window size is considered to be 6 in case of 10 minute sampled data which corresponds to 1 hour. The output size of the LSTM network is equal to 2 data points. The resulting LSTM-MIMO model with 200 units and trained on one month data was shown to have the best results according to equation 1. The forecasted results for the LSTM-MIMO model built are shown in Figure 2(a).

GBRT

We have used the Gradient Boost Regression for Time Series (GBRT) model for forecasting multiple days of the power data. Along with the active power data, the model is also trained with the parameters of the timestamp like month, day, hour, minute, day of the week, day of the year and week of the year as additional features of the data. These features help the model's prediction accuracy. We have used the sliding window technique while training the model with input size of 14 days and output size of 7 days with a 10 minute sampled data. The forecasted results for the GBRT model are shown in Figure 2(b).

7 Conclusion

In this paper we have presented the different energy forecasting techniques using both statistical and machine learning techniques. The data collected by the MiDAS IoT sensor in a diverse range of settings, including manufacturing, education, and hospitals, is used to test the proposed techniques. We demonstrated the use of a statistical ARIMA model, by dynamically changing the model's parameters based on the data being analyzed, as an anomaly detector by comparing the recorded values with the forecasted values using 95% confidence interval as a threshold. While the statistical ARIMA model had the lowest RMSE value of 0.059 in terms of forecasting energy consumption, it can only make point predictions. In order to predict the energy usage several days in advance, we have also looked into the window-based prediction methods employing LSTM and GBRT models. Neural architecture exploration along with the modeled weight equation are used to select the desired LSTM model for the dataset. The RMSE value is used to evaluate the model's prediction performance. The RMSE value for the chosen LSTM-MIMO model is observed to be 0.134, while the RMSE value for the GBRT model, which was used to forecast energy demand for multiple days, is 0.262.

In future, one can extend this work by exploring emerging methods for time series analysis like transformers [22]. The dataset can also be used for new tasks like state identification where the goal is to identify the states, i.e., distinct usage patterns, of a system whose power data is collected, are determined using unsupervised methods [15].

References

[1] Alfares, H. K.; and Nazeeruddin, M. 2002. Electric load forecasting: literature survey and classification of methods. *International journal of systems science*, 33(1): 23–34.

[2] Batra, N.; Kelly, J.; Parson, O.; Dutta, H.; Knottenbelt, W.; Rogers, A.; Singh, A.; and Srivastava, M. 2014. NILMTK: An open source toolkit for non-intrusive load monitoring. In *Proceedings of the 5th international conference on Future energy systems; details at https://github.com/nilmtk/nilmtk/blob/master/README.md*, 265–276.

[3] Bedi, J.; and Toshniwal, D. 2018. Empirical mode decomposition based deep learning for electricity demand forecasting. *IEEE access*, 6: 49144–49156.

[4] Bedi, J.; and Toshniwal, D. 2019. Deep learning framework to forecast electricity demand. *Applied energy*, 238: 1312–1326.

[5] Bedi, J.; and Toshniwal, D. 2020. Energy load time-series forecast using decomposition and autoencoder integrated memory network. *Applied Soft Computing*, 93: 106390.

[6] Elsayed, S.; Thyssens, D.; Rashed, A.; Jomaa, H. S.; and Schmidt-Thieme, L. 2021. Do we really need deep learning models for time series forecasting? *arXiv preprint arXiv:2101.02118*.

[7] Faruk, D. Ö. 2010. A hybrid neural network and ARIMA model for water quality time series prediction. *Engineering applications of artificial intelligence*, 23(4): 586–594.

[8] Ghofrani, M.; Hassanzadeh, M.; Etezadi-Amoli, M.; and Fadali, M. S. 2011. Smart meter based short-term load forecasting for residential customers. In *2011 North American Power Symposium*, 1–5. IEEE.

[9] Hart, G. W. 1992. Nonintrusive appliance load monitoring. *Proceedings of the IEEE*, 80(12): 1870–1891.

[10] Hawkins, D. M. 1980. *Identification of outliers*, volume 11. Springer.

[11] Huang, S.-J.; and Shih, K.-R. 2003. Short-term load forecasting via ARMA model identification including non-Gaussian process considerations. *IEEE Transactions on power systems*, 18(2): 673–679.

[12] Janetzko, H.; Stoffel, F.; Mittelstädt, S.; and Keim, D. A. 2014. Anomaly detection for visual analytics of power consumption data. *Computers & Graphics*, 38: 27–37.

[13] Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; and Liu, T.-Y. 2017. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30.

[14] Luo, Y.; Xiao, Y.; Cheng, L.; Peng, G.; and Yao, D. 2021. Deep learning-based anomaly detection in cyber-physical systems: Progress and opportunities. *ACM Computing Surveys (CSUR)*, 54(5): 1–36.

[15] Muppasani, B.; Anand, C. J.; Appajigowda, C.; Srivastava, B.; and Johri, L. 2022. A Dataset and Baseline Approach for Identifying Usage States from Non-Intrusive Power Sensing With MiDAS IoT-based Sensors. In *arXiv, 2209.00987*.

[16] Purarjomandlangrudi, A.; Ghapanchi, A. H.; and Esmalifalak, M. 2014. A data mining approach for fault diagnosis: An application of anomaly detection algorithm. *Measurement*, 55: 343–352.

[17] Rajapaksha, D.; and Bergmeir, C. 2022. LIMREF: Local Interpretable Model Agnostic Rule-based Explanations for Forecasting, with an Application to Electricity Smart Meter Data. *arXiv preprint arXiv:2202.07766*.

[18] Rezapour, M. 2019. Anomaly detection using unsupervised methods: credit card fraud case study. *International Journal of Advanced Computer Science and Applications*, 10(11).

[19] Sapankevych, N. I.; and Sankar, R. 2009. Time series prediction using support vector machines: a survey. *IEEE computational intelligence magazine*, 4(2): 24–38.

[20] Tso, G. K.; and Yau, K. K. 2007. Predicting electricity energy consumption: A comparison of regression analysis, decision tree and neural networks. *Energy*, 32(9): 1761–1768.

[21] Wang, X.; Zhao, T.; Liu, H.; and He, R. 2019. Power consumption predicting and anomaly detection based on long short-term memory neural network. In *2019*

IEEE 4th international conference on cloud computing and big data analysis (ICCCBDA), 487–491. IEEE.

[22] Wen, Q.; Zhou, T.; Zhang, C.; Chen, W.; Ma, Z.; Yan, J.; and Sun, L. 2022. Transformers in Time Series: A Survey. In *arXiv:2202.07125*.