I. J. Engineering and Manufacturing, **2022, 3, 38-50**
Published Online June 2022 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijem.2022.03.04

Home Occupancy Classification Using Machine Learning Techniques along with Feature Selection

Abdullah-Al Nahid, Niloy Sikder, Mahmudul Hasan Abid, Rafia Nishat Toma, Iffat Ara Talin
Electronics and Communication Engineering Discipline, Khulna University, Khulna-9208, Bangladesh
E-mail: nahid.ece.ku@gmail.com, niloysikder333@gmail.com, abidkst@hotmail.com, rafiatoma.eceku@gmail.com, talin.iffat@gmail.com

Lasker Ershad Ali
Mathematics Discipline, Khulna University, Khulna-9208, Bangladesh
E-mail: ershad@math.ku.ac.bd

Received: 31 March 2022; Revised: 22 April 2022; Accepted: 04 May 2022; Published: 08 June 2022

Abstract: Monitoring systems for electrical appliances have gained massive popularity nowadays. These frameworks can provide consumers with helpful information for energy consumption. Non-intrusive load monitoring (NILM) is the most common method for monitoring a household's energy profile. This research presents an optimized approach for identifying load needs and improving the identification of NILM occupancy surveillance. Our study suggested implementing a dimensionality reduction algorithm, popularly known as genetic algorithm (GA) along with XGBoost, for optimized occupancy monitoring. This exclusive model can masterly anticipate the usage of appliances with a significantly reduced number of voltage-current characteristics. The proposed NILM approach pre-processed the collected data and validated the anticipation performance by comparing the outcomes with the raw dataset's performance metrics. While reducing dimensionality from 480 to 238 features, our GA-based NILM approach accomplished the same performance score in terms of accuracy (73%), recall (81%), ROC-AUC Score (0.81), and PR-AUC Score (0.81) like the original dataset. This study demonstrates that introducing GA in NILM techniques can contribute remarkably to reduce computational complexity without compromising performance.

Index Terms: Occupancy, Energy Consumption, XGBoost, Genetic Algorithm, Feature Selection.

1. Introduction

One of the most precious commodities is electrical energy. Global power demand has expanded dramatically as the world's population has grown. Global warming, water and air pollution, acid rain, depletion, and alteration of our natural ecosystem are all unquestionably tied to energy. Many governments have recognized the relevance of environmental issues and have enacted legislation to reduce yearly emissions of carbon dioxide and urban wastage by 2050 [7]. Traditional fossil-based energy sources are expected to be replaced with non-conventional energy sources in this manner, while present power infrastructure will be turned into an intelligent grid. According to [11], power usage accounts for 40% of carbon dioxide emissions in the United States. Commercial buildings and families utilize over 40% of total energy consumption. At least 20% of this energy can be retained by applying efficient modifications [16]. As we are seeing, energy waste has become a major concern, particularly in metropolitan areas. As a result, it is clear that lowering power use in buildings may greatly reduce energy waste. Energy monitoring and immediate feedback, such as tailored suggestions or real-time usage at the appliance level, have been demonstrated in studies to be particularly helpful. They may be able to lower power bills and make residents more aware of their home's energy profile [27]. However, the advantages of power disaggregation are not limited to inhabitants. Data from appliances can help with research and development.

It has been demonstrated, for example, that energy suppliers can recognize usage trends in power consumption data to estimate future electricity demand. It is crucial for energy suppliers to predict consumer energy consumption. Disproportionate electricity generation can devastate the entire system. Inadequate energy supply produces a decline in electrical frequency, which, if unchecked, might collapse the whole power system and create a catastrophic blackout [26]. When there is an excess of energy supply relative to demand, the electrical frequency rises over the customary limit. Because power plants run at a certain frequency, greater frequency increases the risk of grid disconnection.

This work is open access and licensed under the Creative Commons CC BY 4.0 License.

On top of that, storing electricity for future use is a cumbersome as well as extravagant process [17]. As a result, evaluating power usage is crucial in energy generation control. Occupancy monitoring is extremely likely to be among the solutions to not only energy waste but also power generation balance. Several investigations were conducted by researchers to measure the usage of electrical energy using occupancy monitoring. Wang et al. [4] presented a model that investigated the stochastic link between occupant behavior and equipment energy usage. Silva et al. used an Incremental Summarization and Pattern Characterization (ISPC) technique to explore and extract data from electricity meters [2]. Pattern recognition algorithms were used by Abreu et al. to anticipate energy usage in houses [29]. Amasyali et al. gave an excellent evaluation of research that linked occupancy behavioral monitoring with energy consumption prediction [3].

Introducing Appliance Load Monitoring (ALM) provides a solid foundation for building occupancy monitoring to reinforce smart grid as well as home and office automation systems. Non-Intrusive Load Monitoring (ILM) and Intrusive Load Monitoring (NILM) are the two types of ALM [15]. The ILM technique can provide the most precise appliance consumption information. Regardless, monitoring individual appliances necessitates additional hardware installation and adds system complexity. As a result, implementing (ILM) on a broad scale is not a practicable technique. NILM, on the other hand, can break down the power usage of each device using only single sensor. As a result, NILM is a commonly used technique in occupancy surveillance [20, 22]. Keeping track of aggregated energy consumption provides scrupulous forecasting of electricity consumption behavior.

Two solid assessments of similar work in NILM were presented by Zeifman et al. [22] and Zoha et al. [20]. Many NILM approaches have been investigated since George Hart initially proposed them in 1992 [21]. Several of them rely on machine learning techniques [24], the cross-entropy approach [18], deep neural networks [19, 25], optimum classifiers [6], time series distance [28], and factorial hidden Markov models [5]. The determination of the unique collection of features, which employs signal processing techniques to choose characteristics from the measurement of voltage (V) as well as current (I), is a critical step in NILM. The sample rate determines the sort of characteristics that may be recovered from the measurements of voltage (V) and current (I) [22]. Depending on the condition of the observed waveform, the characteristics are also classified as constant or transient [20].

Several NILM algorithms rely solely on active power. Besides Many intelligent meters, measure other characteristics like power factor (PF), reactive power, apparent power, and total harmonic distortion (THD). These kind of signals might be exploited for better categorization. The active-reactive power signal is a common characteristic that was also employed in Hart's original NILM approach [21]. Individual appliance categorization may be enhanced by using more characteristics. While increasing features may boost the classification performance of NILM algorithms, it is not necessarily the ideal solution owing to the increasing time complexity. Furthermore, there is no assurance that adding more features promotes classification accuracy because adding too many characteristics may result in over-fitting.

To address the aforementioned issues, some works recommend employing a feature selection strategy, where the objective is to develop a lighter model with a small number of characteristics [13]. The pioneering work in NILM feature selection is [10], which employs a neural network. After this, [14] selects the features using the wavelet transform and the short-time Fourier transform. Recent studies on feature selection may be found in [8] and [1]. A recursive feature reduction technique is utilised in [8] to determine the most appealing feature set from the PLAID dataset [15]. This methodology is quite complex and is primarily focused on the deletion of features using heuristic approaches. [1] Provides a forward selection technique that selects features by studying rapid transients from four independent datasets. In [48], the authors incorporated different pattern recognition techniques, and mathematical models, along with committee decision approach in order to disaggregate load. For this purpose, they incorporated current waveform, instantaneous admittance and power waveform, real/reactive power, switching transient waveform, harmonics and eigenvalues. [49] used active/reactive power, quantized currents along with voltages waveforms, harmonics, also V–I binary image to achieve superior results than features from a single category. Furthermore, [9, 12] transform correlated features to non-correlated features in order to ease the required classification procedure. The data discussion is based on the principal component analysis (PCA) transform that is employed for new data representation in this study.

But even so, when certain features are excluded from the dataset, feature selection approaches in any case would lose information. Feature selection frequently decreases model complexity at the expense of accuracy. Our aim in this research is to select features without compromising model performance. To this end, the Genetic Algorithm based feature selection method works really good for NILM dataset as our work provides a model that can operate with the same performance as the entire set of features with a much-reduced amount of characteristics. On the other hand, the methods which are practiced in literature are either complex or not give promising performance, as described above in the literature reviewed.

The remaining of our research study is broken into four sections. The techniques of our research, as well as a brief overview of the datasets and classifier employed, are provided in Section II. Our work's outcome is shown in Section III. Here we've also included a summary of our research's findings, outcomes, and analyses. Our research comes to an end with Section I

2. Methodology

In this study, we have classified a so-called ECO dataset, and followed a few steps such as data preparation, feature extraction, feature selection, and classification. Our entire operation is depicted in Fig. 1. The following

subsections detail all of the steps involved in this research.

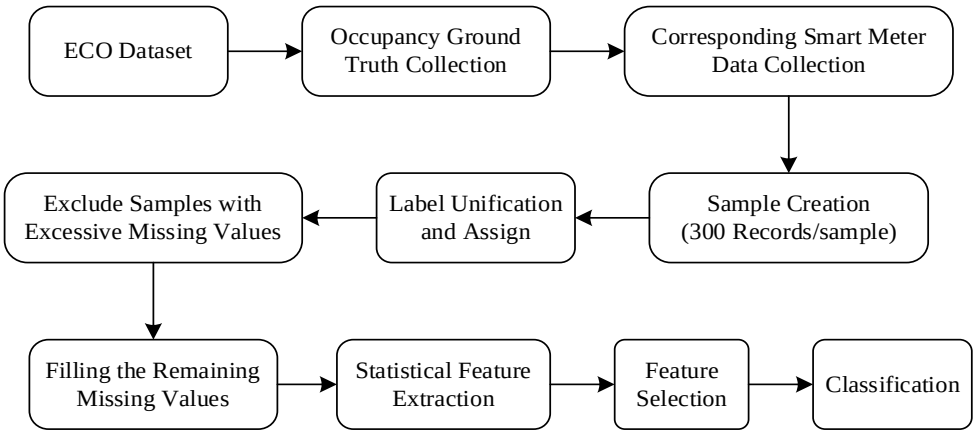

Fig.1. Workflow of data processing.

2.1 Used Dataset and Preprocessing of Data

In this study, we used the Electricity Consumption and Occupancy (ECO) dataset for occupancy detection [38]. ECO dataset is one of the largest publically available smart meter datasets containing electricity consumption data from smart meters and the occupancy ground truths [39], [40]. The data were collected from five households in Switzerland between June 2012 and January 2013, which amounts to more than six months' data. Records were gathered at a frequency of 1 Hz using commercial digital power meters installed in the households. In total, 106,337,104 records were collected using five meters, where each record expresses the average power (in watts) consumed by a particular household during the exact second of measurement. Besides aggregate power, they also provide the three-phase power, neutral current, three-phase current, three-phase voltage, and phase angles of V12, V13, I1−V1, I2−V2, and I3-V3 measurements [41]. Apart from these records, the dataset also contains device-specific power consumption data. However, in this study, we only used the aggregate data of the entire household. The ground truth values were manually assigned using five Galaxy Tab P7510 devices. The procedure of data collection, specifications of the used devices and sensors, and the initial data preprocessing steps have been described in detail in [42]. Fig. 2 illustrates a typical day's power consumption data along with occupancy ground truths from the ECO dataset.

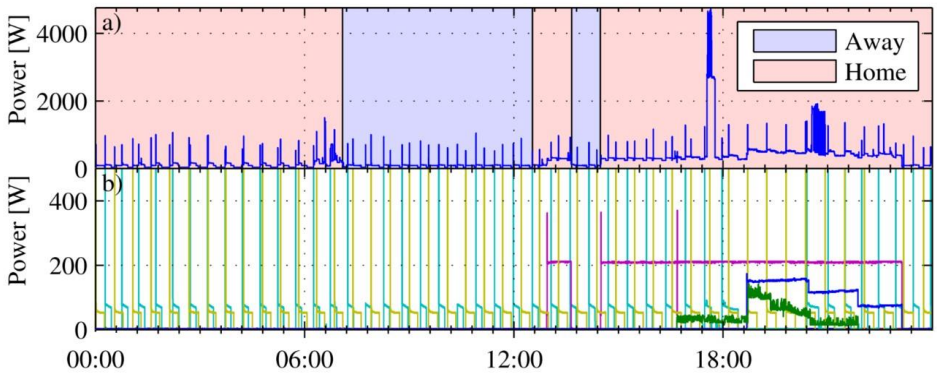

Fig. 2. (a) Aggregate power consumption data of a whole day with occupancy labels and (b) appliance-level consumption data during the day.

A. ECO Data Processing

The occupancy labels of the ECO dataset are provided separately in terms of households (from 1 to 5) and seasons (summer and winter) in Comma Separated Value (.csv) files. Each .csv file contains the occupancy label ("0" for away/unoccupied and "1" for home/occupied) of specific days. However, the occupancy labels were not recorded or provided for all days when power data were collected, meaning all the records do not have ground truths. Nonetheless, since we aim to design a supervised learning model for occupancy detection, having the ground truths is a must for training and determining the testing performance. Hence, we only considered the days that have occupancy labels associated with them. Combining all the households, we found 451 days for which occupancy labels were provided.

Upon recording the dates of those days (and the households), we sequentially accessed their corresponding smart meter data, which are provided in house and date-specific MATLAB data (.mat) files. Each .mat file contains 16 streams of data that express various voltage, current, and power phase measurements. We considered them as individual

streams and processed them separately throughout the study. Since the sampling rate of the dataset is 1 Hz, there are 86,400 records for each day. We opted to take 5 minutes' data or 300 records in each sample in this study. This resulted in 288 samples per day and 129,888 samples in total. We processed the 16 streams of data as 16 different channels of a sample. In terms of the labels, the dataset has an occupancy label associated with each record. However, for the sake of ease in processing, we grouped them to have one label for each sample. In this study, we considered the house as occupied only if someone was present for 90% or more of the 5 minutes of a given sample. After this step, we had 23,563 unoccupied and 106,323 occupied samples.

The ECO dataset has a lot of missing smart meter values (identified as NaN values). If a sample has too many NaN values, it may not contribute much to the training process. Therefore, the presence of such values was checked in all the samples, and samples with more than 75% NaN values were omitted. For the existing samples, the NaN values were replaced with the sample's median values of the corresponding channel. After this step, we acquired 80,639 samples, of which 23,497 belong to the unoccupied and 57,142 belong to the occupied category. Fig. 6. illustrates the entire workflow of data preparation.

B. Feature Extraction

To reduce the dimensionality of the data and the complexity of the learning model, we extracted 30 statistical features from each channel's data of each sample. These features represent the corresponding samples at the learning and classification stage. Table 1 provides a list of the features.

Table 1. Description of the features

No.	Feature	No.	Feature
1	Maximum	16	Root-sum-of-squares level
2	Minimum	17	Number of times the signal crosses zero
3	Mean	18	Entropy
4	Harmonic mean	19	Shanon's entropy
5	Median	20	Mean frequency
6	Mode	21	Median frequency
7	Variance	21	Peak frequency
8	Standard deviation	23	Sum of all NFFT coefficients
9	Mean absolute deviation	24	SNR
10	Range	25	Band power
11	Interquartile range of timeseries data	26	Energy
12	RMS	27	DC power
13	Kurtosis	28	Peak power
14	Skewness	29	Spurious free dynamic range
15	Maximum-to-minimum difference	30	Peak-magnitude-to-RMS ratio

2.2 Feature Selection

When it comes to Machine learning-based classification, not all of a sample set's features are equally significant. Some of them may have a significant impact, while others may have only a minor impact. If we maintain only essential features and discard the rest, the classification result will not be affected significantly, but the data size will be reduced considerably. There are a variety of feature selection algorithms to choose from. Genetic Algorithm, Sequential Floating Forward/ Backward Search, Particle Swarm Optimization, and more algorithms are examples. We employed a Genetic Algorithm in this study to pick key features from the dataset.

Natural genetics inspired the concept of the genetic algorithm. Let's take a look at some of the key terminology in genetic algorithms that are analogous to natural genetics shown as Table 2.

Table 2. Natural Genetics vs Genetic Algorithm

Natural Genetics	**Genetic Algorithm**
Genotype	Structure
Locus	String Position
Allele	Feature Value
Gene	Features
Chromosome	String

Generic algorithms begin with a population of strings chosen at random (In most cases). This string population is used to generate the string's subsequent population. We have three operations for generating a new population:

a. Reproduction- According to biological science, "Reproduction is a biological process in which an organism generates offspring that is biologically identical to it." In the GA perspective, reproduction is the process of copying individual strings. This copying procedure is carried out based on the value of their objective function, i.e. fitness value. Strings with a high fitness value have a good probability of surviving. Strings with a low fitness rating have a slim to non-existent probability of surviving. The survived strings are moved to the mating pool, and a new population (offspring) is produced.
b. Crossover- according to the biological world, "The process through which homologous chromosomes swap portions with one another is known as crossing-over." If we go deep, Genes are contained in chromosomes. So it can be said, by crossing over, genes between two homologous chromosomes be interchanged. This process results in genetic diversity. In the GA perspective, a little portion of an offspring's string is replaced by a little portion of another string. That is, part of the features set (Weather used or unused) between two strings should be swapped from the same place of two strings to ensure population diversity.
c. Mutation- According to biological science- "mutations are changes in our DNA sequence that happen as a result of errors made during DNA copying or environmental influences like Ultraviolet rays and tobacco smoking." In GA, we make deliberate alterations to offspring's strings in order to increase the variation between two consecutive generations further. There are some defined techniques for mutation in GA. They are- Random Resetting, BitFlip Mutation, Scramble Mutation, Swap Mutation, Inversion Mutation, etc. According to the model or performance, the GA designer should choose the best mutation operator for his problem.

All of these stages should be repeated until no substantial offspring have been produced. Fig 2. below represents the whole procedure,

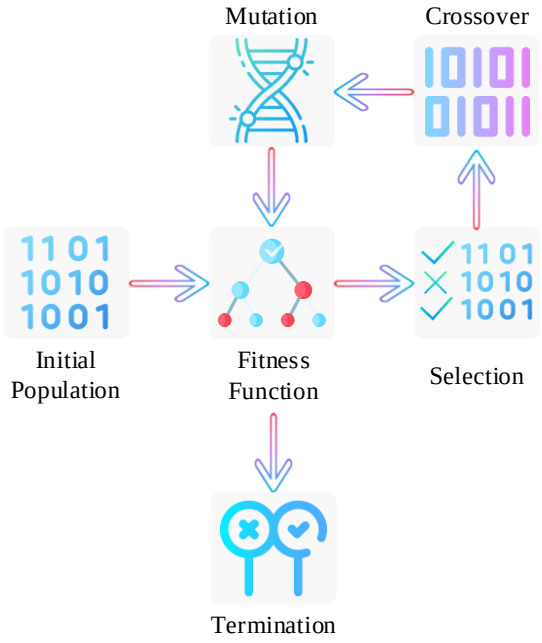

Fig. 3. Feature selection procedure using genetic algorithm.

2.3 Used Classifier

The following considerations influence which machine learning algorithm is best for a specific problem: the size of the dataset, labeled or unlabeled dataset, expected output type, feature number, number of tunable parameters in the algorithm, model training time, overfitting issues, and sparsity of data. We used the Microsoft Azure machine learning cheat sheet [37] to get a better understanding in the process of algorithm choosing for a specific problem. The ECO dataset, which is being utilized for the load prediction in this research, is a labeled dataset with 480 features as mentioned in section 2.1 and each label consisting of 80639 data points, making the dataset rather huge. The K-NN and SVM algorithms can be eliminated at this point because they perform poorly as the dataset grows larger. Because of its ability to handle big datasets, the XGBoost algorithm will be a good fit. According to the literature [30-35], the XGBoost algorithm is a candidate which can operate with huge and sparse datasets, has strong numerical feature performance, has various hyperparameters that can be adjusted for best performance, and, finally, has a rapid processing speed and accurate prediction performance.

The XGBoost algorithm is a decision tree-based algorithm that is developed on the Gradient Boost (GB) framework. The primary operational pattern of XGB is the same as GB. But 'weight' is a new concept introduced in XGB. In XGB, weights are assigned to all the independent variables before feeding to the first decision tree. Then likewise to GB, the decision tree is fitted and evaluated by the 'loss function'. Then further tree is added following the rule which are followed for GB. But before feeding the newly added tree, a new job is done in XGB. The weight of variables that were mispredicted by the tree is updated. So, in XGB, features' weights are updated (if needed) along with the decision trees in every iteration.

2.4 Model Training

The XGBoost model was constructed using the Google Co-Lab and the Python programming language. The data was saved as a CSV file at first and then imported into the Python IDE. The dataset has 238/480 characteristics, having a label of '0' or '1', with '0' indicating unoccupied and '1' indicating an occupied event. The characteristics and label of the dataset are extracted as X and Y, respectively. The Scikitlearn's train test split function is then used to split the dataset into training and test datasets. The XGBoost algorithm's hyperparameters are first set to default settings, and the training process begins. The model is then put to the test using the test dataset, and the predictive performance and confusion matrix are computed. The settings of the hyperparameters are changed by trial and error until the best predictive accuracy is achieved. The tree booster hyperparameters used in the model are listed below [36].

i. Learning rate ['0.3' is default]
   - Reduces the weights on each step to make the model more robust.
   - The following are typical final numbers to use: 0.01-0.2

ii. min_child_weight ['1' is default]
   - It's utilized to keep over fitting in control.
   - Identifies the child's minimum weighted total of all required observations.

iii. max_depth [default=6]
   - A tree's greatest depth.
   - The model will be able to learn extremely specific relationships to a given sample if the depth is increased.. Hence it can be used to control over-fitting.
   - 3–10 are typical values.

iv. gamma [default=0]
   - For each tree, this is the percentage of observations that will be randomly sampled.
   - Typical ranges are 0.5-1.

v. subsample [default=1]
   - For each tree, this is the percentage of observations that will be randomly sampled.
   - 0.5-1 is the typical range.

vi. colsample_bytree ['1' is default]
   - For each tree, this value represents the percentage of columns that will be randomly sampled.
   - 0.5-1 includes average values.

vii. colsample_bylevel ['1' is default]
   - For each split, in each level, this is the subsample ratio of columns.

viii. lambda [default=1]
   - This was where XGBoost's regularization was handled. Despite the fact that many data scientists do not utilize it frequently, it should be investigated in order to avoid overfitting.

ix. alpha [default=0]

- When there is a lot of dimensionalities, this method can be utilized to make the algorithm run faster.

x. n_estimators [‘100’ is default]

- Numbers of trees to fit.

2.5 Performance Analysis

First, in the result and discussion area, the fitness curve obtained from GA applicants was displayed. We utilized a confusion matrix to examine the performances at various stages of our work. The number of right and wrong predictions can be shown with count values using a confusion matrix. Precision, recall, and F1-score were also used for each individual class. As we know,

$$precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{1}$$

That means precision is- out of all positive predictions, how many is got right.

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{2}$$

That means recall is- out of all positive Truth how many is got right.

$$F1\ score = 2 * \frac{precision - recall}{precision + recall} \tag{3}$$

It’s the harmonic mean of recall and precision that determines a model’s overall health or performance.

In addition, the ROC Curve has been used. The False Positive rate is shown on the X-axis of the ROC curve, while the True Positive rate is shown on the Y-axis. The area there under the ROC curve is measured by the AUC value, which goes from 0 to 1. An ideal model's AUC score would be '1'.

3. Result and Discussion

3.1 Application of GA

Influenced by the process of natural selection, GA is a beautiful tool for narrowing down the features from any dataset, settling upon their importance in the classification task. The fitness curve produced by our GA application on the original dataset is shown in Fig. 4. The population size of every generation was 30 in our application.

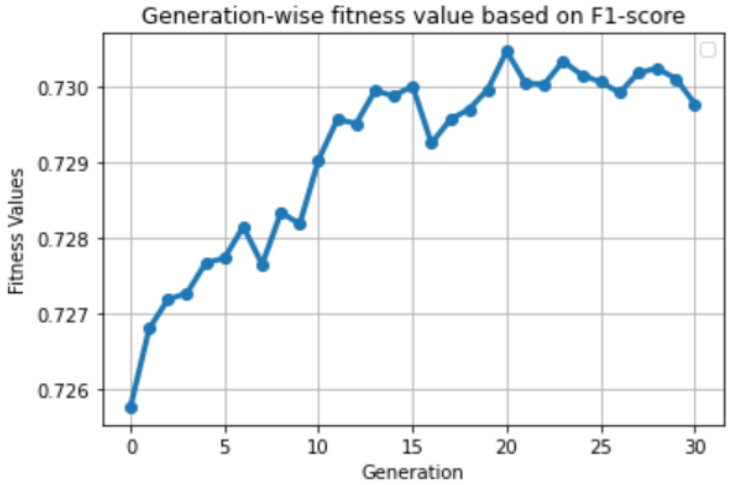

Fig. 4. Evolution with GA

From Fig. 4, it can be observed that the fitness values were not very spectacular during the first few generations. The first few generations had a fitness rate below 72.8%. However, in the 10th generation, it increased to 72.9%, a significant increase. Prior to the 16th generation, the values gradually increased. Following a decline in fitness value in the 16th generation, the value progressively grew again, reaching its peak in the 20th generation. After attaining the maximum value in the 20th generation, the value of fitness value began to go up and down, but not exceeding the peak

value. GA chose 238 features from a total of 480 for the 20th generation. Those 238 features are widely regarded as the most influential features. The selected features are-
[3, 6, 7, 8, 10, 14, 15, 16, 17, 18, 19, 20, 21, 23, 24, 26, 27, 28, 29, 30, 31, 32, 33, 35, 36, 39, 40, 42, 43, 44, 46, 50, 51, 52, 53, 54, 58, 59, 60, 61, 63, 65, 66, 68, 69, 70, 72, 73, 77, 79, 80, 83, 85, 87, 88, 89, 91, 92, 95, 96, 97, 98, 101, 102, 103, 106, 114, 115, 116, 117, 122, 124, 125, 128, 129, 130, 131, 133, 134, 136, 141, 145, 146, 147, 148, 149, 150, 155, 159, 164, 165, 167, 168, 173, 174, 178, 179, 181, 183, 184, 186, 187, 189, 190, 195, 197, 198, 199, 204, 205, 210, 211, 212, 213, 214, 218, 219, 221, 222, 226, 228, 232, 237, 240, 241, 245, 247, 248, 251, 256, 257, 258, 260, 261, 266, 269, 271, 276, 282, 283, 285, 290, 291, 293, 295, 297, 298, 302, 306, 307, 308, 309, 312, 314, 316, 317, 324, 325, 328, 332, 333, 335, 337, 338, 340, 344, 348, 349, 352, 355, 356, 359, 365, 367, 368, 369, 370, 371, 373, 374, 375, 378, 379, 380, 381, 384, 385, 388, 392, 397, 398, 399, 400, 402, 403, 405, 406, 408, 410, 413, 415, 416, 417, 418, 419, 422, 423, 424, 425, 426, 427, 428, 432, 433, 435, 436, 439, 440, 441, 443, 445, 447, 448, 452, 454, 456, 457, 460, 461, 462, 463, 466, 467, 468, 470, 471, 476, 478]

3.2 Classification Performance of Raw Dataset vs. Dataset with the GA Selected Features

In predictive analytics, a confusion matrix is a breakdown of forecast outcomes on a classification problem.. This essential representation may also be used to calculate various parameters of performance analysis, including— the accuracy, F1 score. Precision, and. Recall. Fig. 4 and 5 reveal the confusion matrix of our classification tasks with the raw dataset and the feature-selected dataset, respectively.

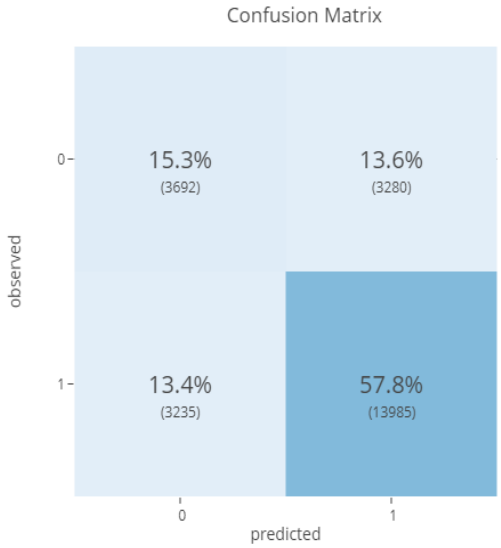

Fig. 5. Confusion matrix of classification of the raw dataset

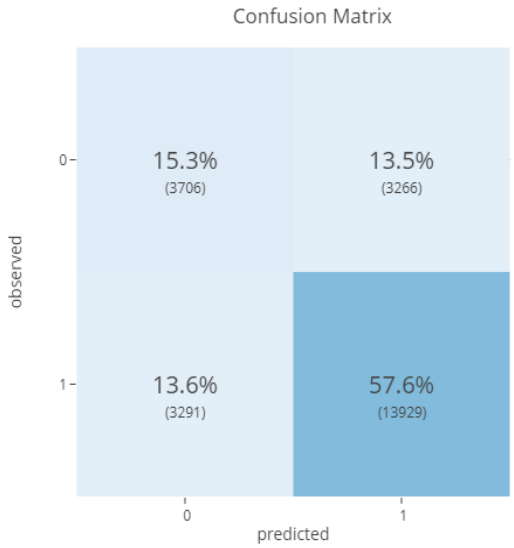

Fig. 6. Confusion matrix of classification of the feature-selected dataset

The confusion matrices show that the feature-selected dataset performed nearly identical to the raw dataset. The value of true positive outcome was the same in both circumstances. Moreover, in the case of the feature-selected dataset, the false positive value decreases, although the true negative value also decreases.

The ROC curve in Fig. (7 & 8) depicts the classifier performance across every classification threshold, both before and after applying feature selection. Our suggested classifier model, coupled with feature selection, yielded the same results as the raw dataset. So, it is essential to say that the GA-based feature selection for this study is as functional as the entire dataset but with fewer features.

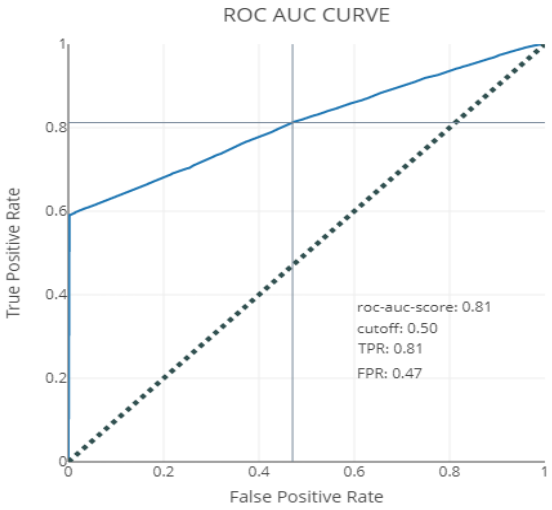

Fig. 7. ROC curve before applying GA

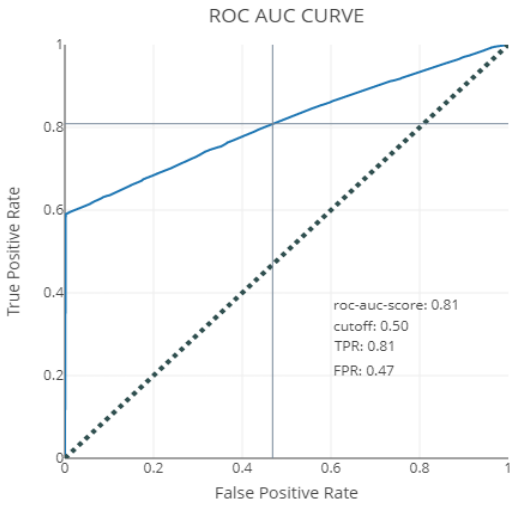

Fig. 8. ROC curve before applying GA

The table 3. below illustrates the detailed classification performance of our raw dataset and feature-selected dataset.

Table 3. Detail classification report for both the dataset

Raw Dataset		Feature-selected Dataset	
Matric	**Score**	**Matric**	**Score**
Accuracy	0.731	Accuracy	0.729
Precision	0.810	Precision	0.81
Recall	0.812	Recall	0.809
F1 Score	0.811	F1 Score	0.809
ROC-AUC Score	0.811	ROC-AUC Score	0.812
PR-AUC Score	0.929	PR-AUC Score	0.930
Log loss	0.404	Log loss	0.404

The table shows that the accuracy with selected 238 features performed almost the same as the raw dataset. Not only this, the precision and log loss value of the raw dataset and feature selected subset are the same. However, the recall and F1-score with chosen features were marginally lower, but not substantially worse, when compared to the raw dataset. In the case of ROC AUC and PR AUC scores, performance with genetic algorithm-suggested features was slightly better than with the entire collection of features. The PR curves for both cases are given below (Fig. 9 & 10).

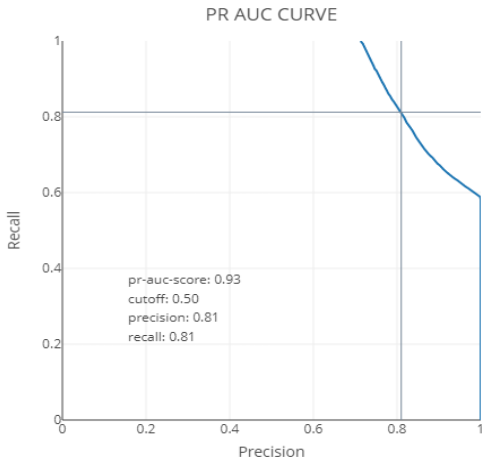

Fig. 9. PR curve before applying GA

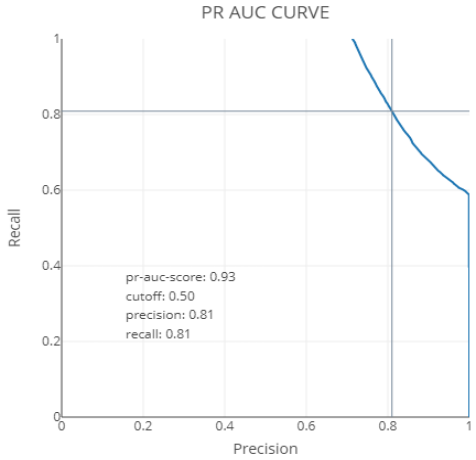

Fig.10. PR curve after applying GA

So, from the comparison study discussed above, it can be said, when all of the performance study results for both using and not using GA-based dimensionality reduction processes are compared, it was evident that our suggested model is capable of performing effectively while reducing a significant computing complexity.

4. Conclusion

Occupancy monitoring plays a salient role in forging not only home automation but also efficient energy supply. This study unfolds a comprehensive approach to estimate the odds-on power consumption profile through occupancy monitoring. This state-of-art can accurately foresee the engagement of electrical appliances with half of the reduced voltage-current characteristics. In our study, we introduced XGBoost accompanying GA to estimate the occupancy of appliances with optimized performance. With reduced dimensionality, the proposed model provided an accuracy of 73%, f1-score, and recall of 81%, which is identical to the raw dataset's performance metrics which can be a milestone for this state-of-art. In aggregate, the findings provided here are interesting and could be beneficial in systems aimed at identifying electric loads, as well as in smart meter algorithms for smart grids. In the future, others classification algorithms, for example, Support Vector Machines, could be investigated, along with other feature extraction and dimensionality reduction methods, such as Particle Swarm Optimization, Fast Fourier Transform, Principal Component Analysis, and Sequential Floating Forward/ Backward Search.

Acknowledgement

This research is supported by Research Grants Program (RGP), Khulna University Research Cell (KURC). Abdullah-Al Nahid acknowledges Khulna University research Cell (KURC)

References

[1] M. Kahl, A. Haq, T. Kriechbaumer, and H. Jacobsen. (1994). "A Comprehensive Feature Study for Appliance Recognition on High Frequency Energy Data," e-Energy, 2017, doi: 10.1145/3077839.3077845.Franklin, M.A. & Pan, T.. Performance Comparison of Asynchronous Adders. In: Symp. on Advanced Research in Asynchronous Circuits and Systems, pp. 117-125.

[2] D. De Silva, X. Yu, D. Alahakoon, and G. Holmes. (2011). "A Data Mining Framework for Electricity Consumption Analysis From Meter Data," IEEE Transactions on Industrial Informatics, vol. 7, no. 3, pp. 399–407, doi: 10.1109/TII.2011.2158844.

[3] K. Amasyali and N. M. El-Gohary. (2018). "A review of data-driven building energy consumption prediction studies," Renewable and Sustainable Energy Reviews, vol. 81, pp. 1192–1205, doi: 10.1016/j.rser.2017.04.095.

[4] Z. Wang and Y. Ding. (2015). "An occupant-based energy consumption prediction model for office equipment," Energy and Buildings, vol. 109, pp. 12–22, Dec. doi: 10.1016/j.enbuild.2015.10.002.

[5] J. Z. Kolter and T. Jaakkola, (2012). "Approximate Inference in Additive Factorial HMMs with Application to Energy Disaggregation," in Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics, pp. 1472–1482. Accessed: Mar. 11, 2022. [Online]. Available: https://proceedings.mlr.press/v22/zico12.html

[6] F. Li, Xu et al., (2018). "Classifier economics of Semi-Intrusive Load Monitoring," International Journal of Electrical Power & Energy Systems, vol. 103, pp. 224–232, doi: 10.1016/j.ijepes.2018.05.010.

[7] "Climate Change Act 2008." https://www.legislation.gov.uk/ukpga/2008/27/contents/enacted (accessed Mar. 11, 2022).

[8] N. Sadeghianpourhamami, J. Ruyssinck, D. Deschrijver, T. Dhaene, and C. Develder, (2017). "Comprehensive feature selection for appliance classification in NILM," Energy and Buildings, vol. 151, pp. 98–106, doi: 10.1016/j.enbuild.2017.06.042.

[9] T. Saitoh, T. Osaki, R. Konishi, and K. Sugahara, (2009). "Current Sensor Based Home Appliance and State of Appliance Recognition," SICE Journal of Control, Measurement, and System Integration, vol. 3, pp. 86–93, doi: 10.9746/jcmsi.3.86.

[10] H.-T. Yang, H.-H. Chang, and C.-L. Lin. (2007). "Design a Neural Network for Features Selection in Non-intrusive Monitoring of Industrial Electrical Loads," 2007 11th International Conference on Computer Supported Cooperative Work in Design, doi: 10.1109/CSCWD.2007.4281579.

[11] "EIA's Annual Energy Outlook 2017," Global Energy Institute. https://www.globalenergyinstitute.org/eias-annual-energy-outlook-2017 (accessed Mar. 11, 2022).

[12] J. Alcalá, J. Ureña, Á. Hernández, and D. Gualda, (2017). "Event-Based Energy Disaggregation Algorithm for Activity Monitoring From a Single-Point Sensor," IEEE Transactions on Instrumentation and Measurement, vol. 66, no. 10, pp. 2615–2626, doi: 10.1109/TIM.2017.2700987.

[13] M. Dash and H. Liu. (1997). "Feature selection for classification," Intelligent Data Analysis, vol. 1, no. 1, pp. 131–156, doi: 10.1016/S1088-467X(97)00008-5.

[14] Y.-C. Su, K.-L. Lian, and H.-H. Chang, (2011). "Feature Selection of Non-intrusive Load Monitoring System Using STFT and Wavelet Transform," in 2011 IEEE 8th International Conference on e-Business Engineering, pp. 293–298. doi: 10.1109/ICEBE.2011.49.

[15] L. Mengqi, J. Gao, and Z. Li. (2018). Functional Intrusive Load Monitor (FILM): A Model-based Platform for Non-Intrusive Load Monitoring System Development.

[16] K. Carrie Armel, A. Gupta, G. Shrimali, and A. Albert, (2013). "Is disaggregation the holy grail of energy efficiency? The case of electricity," Energy Policy, vol. 52, no. C, pp. 213–234.

[17] G. Castagneto Gissey, P. E. Dodds, and J. Radcliffe, (2018). "Market and regulatory barriers to electrical energy storage innovation," Renewable and Sustainable Energy Reviews, vol. 82, pp. 781–790, doi: 10.1016/j.rser.2017.09.079.

[18] R. Machlev, Y. Levron, and Y. Beck, (2019). "Modified Cross-Entropy Method for Classification of Events in NILM Systems," IEEE Transactions on Smart Grid, vol. 10, no. 5, pp. 4962–4973, doi: 10.1109/TSG.2018.2871620.

[19] M. Xia, W. Liu, K. Wang, Z. Xu, and Y. Xu. (2019). "Non-intrusive load disaggregation based on deep dilated residual network," Electric Power Systems Research, vol. 170, pp. 277–285, doi: 10.1016/j.epsr.2019.01.034.

[20] A. Zoha, A. Gluhak, M. A. Imran, and S. Rajasegarar, (2019). "Non-Intrusive Load Monitoring Approaches for Disaggregated Energy Sensing: A Survey," Sensors, vol. 12, no. 12, Art. no. 12, doi: 10.3390/s121216838.

[21] G. W. Hart, (1992) "Nonintrusive appliance load monitoring," Proceedings of the IEEE, vol. 80, no. 12, pp. 1870–1891 doi: 10.1109/5.192069.

[22] M. Zeifman and K. Roth, (2011). "Nonintrusive appliance load monitoring: Review and outlook," IEEE Transactions on Consumer Electronics, vol. 57, no. 1, pp. 76–84, doi: 10.1109/TCE.2011.5735484.

[23] J. Gao, S. Giri, E. C. Kara, and M. Bergés, (2014). "PLAID: a public dataset of high-resoultion electrical appliance measurements for load identification research: demo abstract," in Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings, New York, NY, USA, Nov. pp. 198–199. doi: 10.1145/2674061.2675032.

[24] I. Rahman, M. Kuzlu, and S. Rahman. (2018). "Power disaggregation of combined HVAC loads using supervised machine learning algorithms," Energy and Buildings, vol. 172, pp. 57–66, doi: 10.1016/j.enbuild.2018.03.074.

[25] J. P. Kelly and M. A. James. (2016). "Radiographic Outcomes of Hemiepiphyseal Stapling for Distal Radius Deformity Due to Multiple Hereditary Exostoses," Journal of Pediatric Orthopaedics, vol. 36, no. 1, pp. 42–47, doi: 10.1097/BPO.0000000000000394.

[26] J. A. Short, D. G. Infield, and L. L. Freris. (2007). “Stabilization of Grid Frequency Through Dynamic Demand Control,” IEEE Transactions on Power Systems, vol. 22, no. 3, pp. 1284–1293, doi: 10.1109/TPWRS.2007.901489.
[27] S. Darby, (2006) “The Effectiveness of Feedback on Energy Consumption: A Review of the Literature on Metering, Billing and Direct Displays,” .
[28] K. Basu, V. Debusschere, A. Douzal-Chouakria, and S. Bacha, (2015). “Time series distance-based methods for non-intrusive load monitoring in residential buildings,” Energy and Buildings, vol. 96, pp. 109–117 doi: 10.1016/j.enbuild.2015.03.021.
[29] J. Abreu, F. Pereira, and P. Ferrão, (2012). “Using pattern recognition to identify habitual behavior in residential electricity consumption,” Energy and Buildings, vol. 49, pp. 479–487, doi: 10.1016/j.enbuild.2012.02.044.
[30] T. Chen and C. Guestrin. (2016). “XGBoost: A Scalable Tree Boosting System,” in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, pp. 785–794. doi: 10.1145/2939672.2939785.
[31] “Chen and He - xgboost eXtreme Gradient Boosting.pdf.” Accessed: Mar. 11, 2022. [Online]. Available: https://cran.r-project.org/web/packages/xgboost/vignettes/xgboost.pdf
[32] Computer Science and Engineering, SRM Institute of Science and Technology, Chennai, India. et al. (2019). “Network Intrusion Detection System using XG Boost,” IJEAT, vol. 9, no. 1, pp. 4070–4073, doi: 10.35940/ijeat.A1307.109119.
[33] H. Musa, D. A. Y. Gital, F. U. Zambuk, A. Umar, A. Y. Umar, and J. U. Waziri. (2005). “A COMPARATIVE ANALYSIS OF PHISHING WEBSITE DETECTION USING XGBOOST ALGORITHM,” . Vol., no. 5, p. 10.
[34] H.-S. Choi et al. (2018). “XGBoost-Based Instantaneous Drowsiness Detection Framework Using Multitaper Spectral Information of Electroencephalography,” in Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics, New York, NY, USA, pp. 111–121. doi: 10.1145/3233547.3233567.
[35] C. Krupitzer, T. Sztyler, J. Edinger, M. Breitbach, H. Stuckenschmidt, and C. Becker. (2019). “Beyond position-awareness—Extending a self-adaptive fall detection system,” Pervasive and Mobile Computing, vol. 58, p. 101026, doi: 10.1016/j.pmcj.2019.05.007.
[36] “XGBoost Parameters | XGBoost Parameter Tuning.” https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/ (accessed Mar. 11, 2022).
[37] lgayhardt, “Machine Learning Algorithm Cheat Sheet - designer - Azure Machine Learning.” https://docs.microsoft.com/en-us/azure/machine-learning/algorithm-cheat-sheet (accessed Mar. 11, 2022).
[38] “ECO data set (Electricity Consumption & Occupancy) A Research Project of the Distributed Systems Group.” http://vs.inf.ethz.ch/res/show.html?what=eco-data.
[39] W. Kleiminger, C. Beckel, and S. Santini. (2015). “Household occupancy monitoring using electricity meters,” UbiComp 2015 - Proc. 2015 ACM Int. Jt. Conf. Pervasive Ubiquitous Comput., pp. 975–986, doi: 10.1145/2750858.2807538.
[40] C. Beckel, W. Kleiminger, R. Cicchetti, T. Staake, and S. Santini. (2014). “The ECO data set and the performance of non-intrusive load monitoring algorithms,” BuildSys 2014 - Proc. 1st ACM Conf. Embed. Syst. Energy-Efficient Build., pp. 80–89, doi: 10.1145/2674061.2674064.
[41] C. Oh and J. Jeong, (2020). “Non-intrusive Load Monitoring Based on Regularized ResNet with Multivariate Control Chart,” Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 12250 LNCS, pp. 646–661, doi: 10.1007/978-3-030-58802-1_47.
[42] W. Kleiminger, C. Beckel, T. Staake, and S. Santini. (2013) “Occupancy detection from electricity consumption data,” BuildSys 2013 - Proc. 5th ACM Work. Embed. Syst. Energy-Efficient Build., doi: 10.1145/2528282.2528295.
[43] Fazli Wahid, Rozaida Ghazali, Muhammad Fayaz, Abdul Salam Shah. (2017)."Statistical Features Based Approach (SFBA) for Hourly Energy Consumption Prediction Using Neural Network", International Journal of Information Technology and Computer Science (IJITCS), Vol.9, No.5, pp.23-30, 2017. DOI: 10.5815/ijitcs.2017.05.04.
[44] N. Chabbah Sekma, A. Elleuch, N. Dridi, (2016). "Automated Forecasting Approach Minimizing Prediction Errors of CPU Availability in Distributed Computing Systems", International Journal of Intelligent Systems and Applications(IJISA), Vol.8, No.9, pp.8-21, 2016. DOI: 10.5815/ijisa.2016.09.02.
[45] Manisha Verma, Neelam Bhardwaj, Arun Kumar Yadav,"Real Time Efficient Scheduling Algorithm for Load Balancing in Fog Computing Environment", International Journal of Information Technology and Computer Science(IJITCS), Vol.8, No.4, pp.1-10, 2016. DOI: 10.5815/ijitcs.2016.04.01.
[46] M. Hasan, R. N. Toma, A.-A. Nahid, Mm. Islam, and J.-M. Kim, (2019). “Electricity theft detection in smart grid systems: A CNN-LSTM based approach,” Energies, vol. 12, no. 17, p. 3310, 2019.
[47] R. N. Toma, M. N. Hasan, A.-A. Nahid, and B. Li, (2019). “Electricity theft detection to reduce non-technical loss using support vector machine in smart grid,” in 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), 2019, pp. 1–6.
[48] “Load Signature Study—Part I: Basic Concept, Structure, and Methodology | IEEE Journals & Magazine | IEEE Xplore.” https://ieeexplore.ieee.org/document/5337912 (accessed May 07, 2022).
[49] J. Gao, E. Kara, S. Giri, and M. Bergés, (2015) “A feasibility study of automated plug-load identification from high-frequency measurements,” 2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP), doi: 10.1109/GlobalSIP.2015.7418189.

Authors' Profiles

Dr. Abdullah-Al Nahid is currently working as Professor in Department of Electronics and Communication Engineering, Khulna University, Khulna, Bangladesh. He received the B.Sc. degree in Electronics and Communication Engineering from Khulna University, Khulna, Bangladesh, in 2007, the M.Sc. degree in telecommunication engineering from the Institute for the Telecommunication Research (ITR), University of South Australia (UniSA), Australia, in 2014, and the Ph.D. degree from Macquarie University, Sydney Australia, in 2018. His research interests include machine learning, biomedical image processing, data classification, and smart grid.

Niloy Sikder is currently working as a research assistant at the Faculty of Technology and Bionics at Rhine-Waal University of Applied Sciences, Kleve, Germany, and as a PhD student at the Donders Center for Cognitive Neuroimaging (DCCN) at Radboud University, Nijmegen, The Netherlands. He received his BSc in Electronics and Communication Engineering (ECE) in 2017 and MSc in Computer Science and Engineering (CSE) in 2020 from Khulna University, Khulna, Bangladesh. His present research work at the Donders Sleep & Memory Lab is focused on investigating big sleep datasets with biomedical data processing and machine learning strategies.

Mahmudul Hasan Abid is a student of B.Sc. in Electronics and Communication Engineering, Khulna University, Khulna, Bangladesh. He is currently in the final year of his undergraduate. His research interests include machine learning, IOT, smart grid, microwave communication and digital communication.

Rafia Nishat Toma received the B.Sc. degree in Electronics and Communication Engineering from Khulna University, Khulna, Bangladesh in 2012 and the M.Sc. degree in 2016. She is currently pursuing a Ph.D. degree in computer engineering at the University of Ulsan, South Korea, where she has been a Graduate Research Assistant with the Ulsan Industrial Artificial Intelligence (UIAI) Laboratory since 2019. She is working as Assistant Professor in Electronics and Communication Engineering Discipline at Khulna University and is currently on study leave. Her current research interests include artificial intelligence, signal processing, fault diagnosis, current signal-based condition monitoring of industrial machinery, and feature engineering.

Iffat Ara Talin is a student at Khulna University in Khulna, Bangladesh, studying Electronics and Communication Engineering. She is currently in her final year of Honors. Microwave/Digital communication, Machine learning and smart grid are among her research interests.

Dr. Lasker Ershad Ali received the Bachelor of Science (B.Sc.) degree in Mathematics and Masters of Science (M.Sc.) degree in Applied Mathematics from Khulna University in 2006 and 2008, respectively. He also received the Doctor of Natural Science degree from Peking University in 2018. After working as, a Lecturer (from 2008), an Assistant Professor (from 2010), and an Associate Professor (from 2015), he has been a Professor of Mathematics at Khulna University since 2019. His research interest includes Statistical Learning and Information Intelligence, Biometric, Image Processing, Pattern Recognition, Machine Learning as well as Deep Learning, Computer Vision and Applied Mathematics. He is a life member of Bangladesh Mathematical Society (BMS).

How to cite this paper: Abdullah-Al Nahid, Niloy Sikder, Mahmudul Hasan Abid, Rafia Nishat Toma, Iffat Ara Talin, Lasker Ershad Ali, " Home Occupancy Classification Using Machine Learning Techniques along with Feature Selection ", International Journal of Engineering and Manufacturing (IJEM), Vol.12, No.3, pp. 38-50, 2022. DOI: 10.5815/ijem.2022.03.04