

Deep Neural Networks Based Approach for Battery Life Prediction

Sweta Bhattacharya¹, Praveen Kumar Reddy Maddikunta¹, Iyapparaja Meenakshisundaram¹,
Thippa Reddy Gadekallu¹, Sparsh Sharma², Mohammed Alkahtani³ and Mustufa Haider Abidi^{4,*}

¹School of Information Technology and Engineering, Vellore Institute of Technology, Vellore, India

²Department of Computer Science and Engineering, Baba Ghulam Shah Badshah University, Rajouri, India

³Industrial Engineering Department, College of Engineering, King Saud University, Riyadh, Saudi Arabia

⁴Raytheon Chair for Systems Engineering, Advanced Manufacturing Institute, King Saud University, Riyadh, Saudi Arabia

*Corresponding Author: Mustufa Haider Abidi. Email: mabidi@ksu.edu.sa

Received: 27 December 2020; Accepted: 20 February 2021

Abstract: The Internet of Things (IoT) and related applications have witnessed enormous growth since its inception. The diversity of connecting devices and relevant applications have enabled the use of IoT devices in every domain. Although the applicability of these applications are predominant, battery life remains to be a major challenge for IoT devices, wherein unreliability and shortened life would make an IoT application completely useless. In this work, an optimized deep neural networks based model is used to predict the battery life of the IoT systems. The present study uses the Chicago Park Beach dataset collected from the publicly available data repository for the experimentation of the proposed methodology. The dataset is pre-processed using the attribute mean technique eliminating the missing values and then One-Hot encoding technique is implemented to convert it to numerical format. This processed data is normalized using the Standard Scaler technique. Moth Flame Optimization (MFO) Algorithm is then implemented for selecting the optimal features in the dataset. These optimal features are finally fed into the DNN model and the results generated are evaluated against the state-of-the-art models, which justify the superiority of the proposed MFO-DNN model.

Keywords: Battery life prediction; moth flame optimization; one-hot encoding; standard scaler

1 Introduction

With the rapid development of smart technologies and communication systems, the Internet of Things (IoT) has received a great deal of attention and prominence for energy efficient networks with limited resources. IoT is a giant network in which “things” or embedded sensor devices are connected via private or public network. The functions of IoT devices are remotely controlled and monitored [1]. Once IoT devices capture the data, the sensed information is shared between the communication media using communication protocols. Smart IoT devices can vary from simple, smart gadgets and industrial applications. The prospective importance of IoT is



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

apparent from its application in almost all sectors of our daily human lives. IoT transformation has been increasing exponentially due to rapid advances in hardware technology, increased bandwidth, spectrum efficiency, server resources, etc. Industry 4.0 enabled technologies such as cobots, blockchain, big data analytics, digital twins, federated learning have promoted sustainable IoT growth in various spectrums such as industrial sectors, military services, medical care, transportation, etc. However, IoT devices face multiple challenges in terms of security, latency and energy optimization [2]. Energy optimization is considered to be one of the major challenges, as IoT devices are power-restricted as they perform large computations continuously, consuming enormous amount of energy to conduct the computational tasks. Numerous energy optimization techniques have been introduced in IoT-based systems to minimize energy consumption. The sink node in these devices loses its energy levels at an early stage leading to network failures, while the other sensors in the network still retain sufficient energy. Although the remaining sensors hold sufficient energy levels, they become non-functional. This problem has led the researchers focus on other energy optimization techniques by selecting the optimal Cluster Head (CH) for increasing the life time of a network [3]. In the optimal CH selection process, the optimal node is chosen on the basis of different performance metrics such as energy, load, temperature, and delay. The selected optimal node is called the CH and all nodes in the same cluster transfer their data to the selected CH, ensuring seamless transfer of data from the CH to the base station. The unexpected failures of the IoT devices occur as the battery drainage results in the loss of performance and call for recurrent maintenance requirements. To overcome this challenge, it is essential to predict the battery life of IoT devices, thereby increasing its lifetime. The life span of the IoT node depends on battery life conservation. Accurate prediction of the remaining battery life of the IoT devices is essential for improving stability of the IoT network [4]. Advanced Machine Learning (ML) algorithms must be designed to accurately and efficiently predict the remaining battery life of the IoT devices which would further increase the life time of the network. The applicability of any IoT devices with sensors embedded ceases to exist when its battery drains out. In such instances, the devices fail to capture important information, which could lead to crucial inferences and predictions being lost. Hence, battery life prediction is an absolute necessity in the present world scenario.

The current study focuses on predicting the remaining battery life of the beach-configured IoT sensors [5]. All such sensors capture and monitor data from the beach such as battery life of the device, wave height, turbidity, water temperature, transducer depth, beach name, time stamp measurement and wave period. Predicting the battery life of these IoT devices well in advance with human effort is a cumbersome task. Also, this type of beach-configured IoT sensors need to be active and working 24/7 in order to capture all of the above stated data. The battery life of these devices needs to be un-interrupted as any exceptions could lead to non-inclusion of important data, missing out on important facts, inferences and finally leading to less accurate predictions. The objective of the present study is aligned to the same necessity, wherein deep learning based algorithm is used for prediction of the battery life of the IoT sensors. The ML algorithms and Deep Learning (DL) models help to predict the battery life well in advance so that we can recharge the batteries or change the batteries of the sensors early enough. Experiments are conducted in this study using the publicly available Chicago Park District Beach Water Dataset. Attribute mean technique is used to fill the missing values in the dataset. To transform the categorical data to numerical data, one-hot encoding is used to make it commensurate for the ML model. Later, this numerical data is normalized using the StandardScaler technique and Moth Flame Optimization (MFO) algorithm is used for feature extraction. Post feature extraction,

the dataset containing the optimal features are fed to the Deep Neural Network (DNN) model to predict the remaining battery life of the IoT sensors.

The main contributions of the proposed model are as follows:

- 1) The existence of similar and higher number of features in a dataset tends to make a machine learning model susceptible to overfitting. Dimensionality reduction or feature extraction is thus necessary to ensure improvement in prediction accuracy and explainability of the ML model. Bio-inspired algorithms help to resolve complex, non-linear problems relevant to time constraints and high dimensionality. The MFO algorithm is used in the present study to extract the most important and relevant attributes that have significant impact on the predictions generated from the ML model.
- 2) To predict the remaining battery life of IoT based dataset, an optimized DNN is used.
- 3) An extensive survey is done on MFO and state-of-the-art meta-heuristic algorithms for effective feature selection.

Rest of the paper is organized as follows: Section 2 discusses the review of the literature. Section 3 sets out the preliminary work and the method proposed. Section 4 presents the results and discussions, and Section 5 describes the conclusions and recommendations for future work.

2 Literature Survey

Numerous IoT networks produce enormous amount of data that need to be handled and reacted upon in extremely shorter time frames. One of the significant difficulties in this regard is the high energy utilization of the IoT networks due to the transmission of information to the cloud through IoT devices. The work in [6] proposed an energy saving system for IoT network for an intelligent vehicle system. Here, the authors have divided the whole work into two phases. In the first phase, the authors have applied fast error-bounded lossy compressor on the collected data before its transmission, as it is considered as the highest battery consumer of the IoT devices. In the second phase, the transmitted data was rebuilt on the edge node and processed using machine learning (ML) algorithms. Finally, the authors validated the work by analyzing the energy consumption while transmitting the data between driver and the IoT devices. However, the study also highlighted that real time applications of decompressing time for the proposed model during the transmission of data to the ML model is quite high. The usage of carbon electrodes in electric double layer capacitors (EDLCs) has visualized gradual increase in the past few years due to large surface areas and low cost.

In [7], the authors have established the quantitative correlation between energy and power density behaviors of EDLC in association with structural behavior of activated carbon electrodes using ML algorithms, namely, support vector machine, generalized linear regression, random forest and artificial neural network (ANN). The authors have applied various algorithms for identifying the activated carbon which would help in optimizing battery efficiency and subsequently enhance energy storage. The experimental results have proved that ANN has yielded best performance in predicting the desired activated carbon leading to high energy and power density. Also, the results of the proposed model did not concentrate on the synthesis of carbon electrodes in order to improve performance of the super capacitor.

Battery plays a crucial part in the functioning of electric vehicles. Degradation of battery performance leads to great difficulties in the transportation processes. In [8], the authors have proposed a novel aging phenomenon for handling cloud storage data in vehicle battery modelling. Firstly, the battery degradation phenomenon are quantified and then aging index is generated for

cloud battery data using the rain-flow cycle counting (RCC) algorithm in the proposed battery aging trajectory extraction framework. Secondly, the deep learning algorithm is employed to mine the aging features of the battery. For experimental purposes, the authors have applied the proposed model on electric buses in Zhengzhou and validated the same to justify its practical performance. The results finally proved the potential of the proposed model in simulating the characteristics of battery accurately. Also, it is revealed that the terminal voltage and state of charge estimation error could be limited within 2.17% and 1.08%, respectively. The authors in [9] have reviewed the state-of-the-art ML techniques in designing of rechargeable battery materials. In this survey, the authors have focused on three methods, namely, wrapper feature selection, embedded feature selection, and combination of both for obtaining the essential features of physical/chemical/other properties of battery materials. Also, the applications of ML are reviewed in rechargeable battery designs of liquid and solid electrolytes. The authors have emphasized on the key challenges of ML in rechargeable battery relevant to material science. In response to these challenges, the authors have proposed possible counter measures and potential future directions.

The authors in [10] have focused on two key challenges: edge devices communicating with each other in wide range networks, and battery life of off-grid IoT applications. The authors in this study have found solutions to these challenges by including ML techniques into edge devices. This has resulted in the performance of low power transmission through LoRa network. The study also demonstrated suitable solutions through field trial conducted in China for sow activity classification. Finally, based on experimental results it has been proved that transmitted data could be compressed down to almost 512 times and the battery life could be increased to 3 times in comparison to the existing methods.

Estimating the battery capacity is a crucial task for safe operation and decision making in various real-time processes. In [11], the authors have concentrated on distinct measurable data in battery management systems for exploiting multi-channel charging profile like voltage, current, and temperature. The usage of neural network algorithms, such as, feed forward, convolutional, long short term memory have helped in analyzing the data and understand the relationship between battery capacity and charging profiles. Experimental results confirmed that the proposed multi-channel charging profile technique based on the specified data has outperformed the convolutional neural network. However, the work does not consider internal parameters of physics based equations affecting the results in real time applications. A review of the existing studies is presented in [12] for estimating the battery state of electric vehicles. The authors have surveyed various battery state estimation methods such as feed-forward neural network, recurrent neural network, SVM, hamming networks and radial basis functions. The methods were further compared for estimating state of charge (SOC) and state of health (SOH) in order to achieve enhancement in the availability of battery data considering data quality, test conditions, battery quality, battery type, input and output. Finally, the method was trained 50 times with 300 epochs that achieved quite accurate results with an error of 1%. This work could be further extended to train and deploy the machine learning models with computational complexity.

In [13], the authors have emphasized on predicting the residual battery life of lithium ion batteries and its capacity using advanced ML techniques. Initially, the intrinsic mode function of battery capacity data is decomposed by using empirical mode decomposition (EMD) method. Then long short term memory algorithm (LSTM) is applied with Gaussian process regression (GPR) sub model for estimating the residual which fits to the intrinsic mode function with uncertainty level. Finally, the performance of the proposed LSTM with GPR model is compared with LSTM/GPR/LSTM+EMD/GPR+EMD. It is evident from the comparison that the proposed

LSTM+GPR yields better adoptability, reliability, lower uncertainty quantification for diagnosis of battery health. However, the work could be further improved by focusing on robust data driven model with uncertainty management for improving remaining useful life of the battery.

Since the world is progressing towards a greener environment, it is the need of the hour to empower the energy of battery storage systems. The authors in [14] have concentrated on enhancing the timings of energy stored in the battery. They have suggested ridge and lasso ML regression methods to enrich SOC of the battery. These methods have been used in association with linear regression, SVM and ANN for estimating SOC of the battery. Experimental results proved that the proposed machine learning method with non-convolutional features has provided high accuracy and speed in computing battery state of the charge estimation. The need for the IoT devices in the professional and personal domain have been ever increasing and these devices require use of sensors which creates necessity for energy optimization. In [15], the authors have focused on energy conservation of IoT devices with calibrated power sources to increase the resource utilization using reinforcement ML techniques. The authors have designed a sensor based close grid resource management system and introduced Deep Q Learning (DQL) model to explore power conservation and resource utilization of IoT sensors. Simulation results have proved that the proposed model has yielded high utilization of resources and energy conservation of the IoT devices in real time applications.

DNN has been implemented successfully for the analysis and decision making in the fields of computer vision, drug design, disease prediction, intrusion detection predictions, image processing, Blockchain, and various others [16–21]. Machine learning models has shown great potential in pattern recognition and prediction of outcomes for diversified datasets through the development of models that yield accurate decision making [22,23].

In [24,25], the authors have suggested ML model for predicting the battery life of IoT devices. Here, the authors have implemented proposed model involving ‘Beach water quality-automated sensors’ dataset generated by IoT sensors in Chicago city, USA. Experimental results have proved that the proposed model generated 97% predictive accuracy in preserving battery life of IoT devices, which is better than the existing regression algorithms. However, this work can be enhanced further by using deep neural network for battery life preservation.

The above literature about battery life preservation and energy utilization gives the insights to design an efficient simulation model with advanced ML techniques for enhancing battery state of charge in IoT devices.

3 Preliminaries and Proposed Methodology

3.1 *Moth-Flame Optimization (MFO)*

There exists almost 160,000 different types of insect species in the world. Moths belong to the butterfly family species. A life of a moth revolves around fulfilment of two main objectives—creation of larvae and adults. The larvae gets converted to a moth within the cocoons. Moths navigate using a very specialized yet efficient mechanism in the night time. Moths use moonlight to fly in the night utilizing the traverse orientation mechanism that aids them to navigate. The moths fly maintaining a specific angle with the moon, which helps them to traverse longer distance in a straight path. Although the traverse orientation mechanism is found to be extremely effective, the moth flying motion is often observed to be random and spiral around the light. These exceptions occur in the presence of artificial lights, wherein moths perceive the artificial light and traverse

straight towards such light maintaining a specific angle. But the artificial light being in such close proximity confuses the moths to engage in a deadly spiral fly leading to deadly consequences.

The Moth-Flame optimization algorithm is a meta-heuristic algorithm that follows similar behavioral pattern as moths and thus plays a significant role in feature selection [26]. The optimal position of the moths is presented using the following mathematical model wherein the list of moths is expressed using the following Eq. (1).

$$P = \begin{bmatrix} p_{1,1} & p_{1,2} & \cdots & \cdots & p_{1,n} \\ p_{2,1} & p_{2,2} & \cdots & \cdots & p_{2,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ p_{x,1} & p_{x,2} & \cdots & \cdots & p_{x,n} \end{bmatrix} \quad (1)$$

Here x refers to the no. of moths and n refers to the dimensions. Eq. (2) reflects the array for storing the fitness value of the moths.

$$FP = \begin{bmatrix} FP_1 \\ FP_2 \\ \vdots \\ FP_x \end{bmatrix} \quad (2)$$

The list of flames is shown in Eq. (3) that syncs with the previous equations, x denote the number of moths and n indicates the number of dimensions. Eq. (4) reflects the array for storing the fitness value of the flames.

$$Q = \begin{bmatrix} Q_{1,1} & Q_{1,2} & \cdots & \cdots & Q_{1,n} \\ Q_{2,1} & Q_{2,2} & \cdots & \cdots & Q_{2,n} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ Q_{x,1} & Q_{x,2} & \cdots & \cdots & Q_{x,n} \end{bmatrix} \quad (3)$$

$$FQ = \begin{bmatrix} FQ_1 \\ FQ_2 \\ \vdots \\ FQ_x \end{bmatrix} \quad (4)$$

Here, x denotes the number of flames. The MFO provides a three tuple solution which quantifies and presents the original solution as per the following Eq. (5).

$$PQF = (R, S, T) \quad (5)$$

R is a mathematical function that generates arbitrary moth sample size and fitness value as presented in Eqs. (6) and (7). S is a primary function which keeps the moths moving across the orbit considered. If the criterion gets fulfilled in Eq. (8), T generates the value 1 or else 0 in case the criterion is not met. The upper and lower limits of the parameters are expressed in the form of an array as defined in Eqs. (9) and (10).

$$R: \emptyset \rightarrow \{P, FP\} \quad (6)$$

$$S: P \rightarrow P \quad (7)$$

$$T: P \rightarrow \{1, 0\} \quad (8)$$

$$W = [W_1, W_2, W_3, \dots, W_{x-1}, W_x] \quad (9)$$

$$w = [w_1, w_2, w_3, \dots, w_{x-1}, w_x] \quad (10)$$

The position of the moth with reference to the flame is presented in Eq. (11).

$$P_i = Z(P_i, Q_j) \quad (11)$$

Here P_i presents the i^{th} moth, Q_j presents the j^{th} flame and Z represent spiral functions. The moth update process is computed using a logarithmic spiral model. The moth's position towards the direction of the flames follows a spiral pattern following the below mentioned conditions:

- 1) The original point of the spiral should start with the moth.
- 2) The termination point of the spiral should end with the flame.
- 3) The spectrum of the spiral should not go beyond or away from the search space which is given in Fig. 1.

$$Z(P_i, Q_j) = L_i \cdot g^{fh} \cdot \cos(2\pi h) + Q_j \quad (12)$$

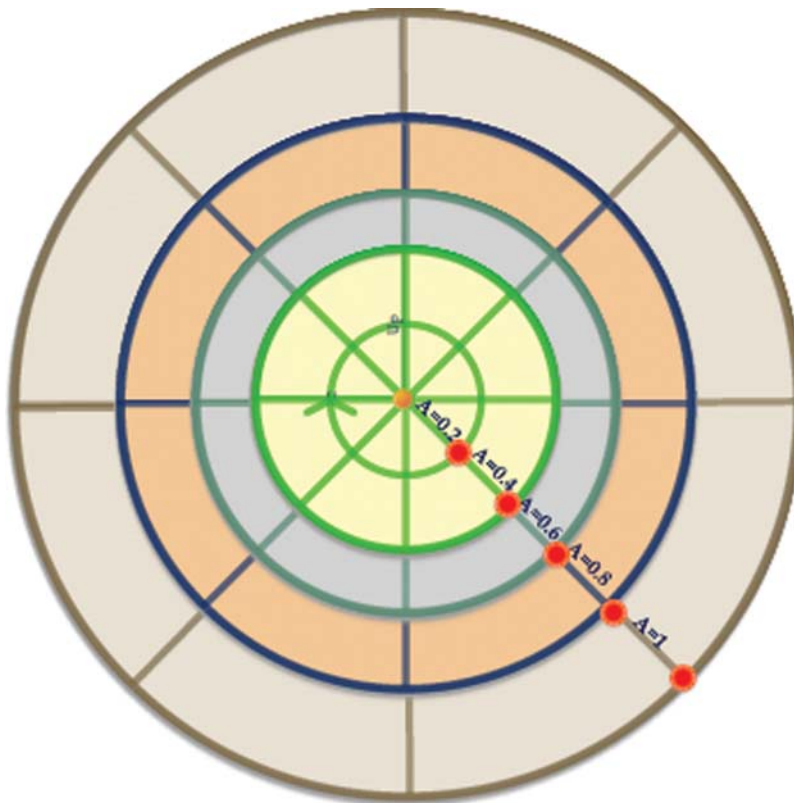


Figure 1: Best solution with in the search space

Here Y_i reflects the distance between i^{th} moth and the j^{th} flame. Also, f is a value that reflects the shape of the spiral and h is a random number ranging from $[-1, 1]$. The distance between the i^{th} moth and j^{th} is calculated using Eq. (13) as shown below:

$$Y_i = |Q_j - P_i| \quad (13)$$

The above mentioned work flow of the MFO algorithm can be summarized basically into two steps as follows:

- a. Generation of the initial position of the Moths, and
- b. Updation of the number of Flames.

Algorithm 1: Pseudo code of MFO Algorithm

Initialize the list of moths using Eq. (1).

Initialize the list of flames using Eq. (3).

Generate the arbitrary moth sample size using Eq. (5).

Compute the fitness value of each search agent using Eqs. (6) and (7).

Update the position of the moth with reference to the flame using Eq. (11).

The moth update process is computed using a logarithmic spiral model using Eq. (12).

for moth's position towards the direction of the flames

The original point of the spiral should start with the moth.

The termination point of the spiral should end with the flame.

The spectrum of the spiral should not go beyond or away from the search space.

for $i = 1$ to n do

for $x = 1$ to n do

Update P , Q values.

Update the distance between the i^{th} moth and j^{th} flame using Eq. (13).

Update the position of moth using Eq. (12).

3.2 Deep Neural Network

Deep learning is a field of machine learning. The basic framework is based on the functioning and biological structure of the brain that enables machines to achieve human like intelligence. The basic version of Deep Neural Network represents a hierarchical organization of neurons which pass messages to other neurons based on the input thereby forming a complex network learning from the feedback mechanism. The input data is fed into the non-hidden neurons of the first layer wherein output of this layer is fed as an input to the subsequent layer and this continues until the final output is achieved. The output is represented in the form of a probability resulting in the prediction of either “Yes” or “No”. The neurons in each layer compute a small function termed as “activation function” which helps to pass the signal to next relevant neurons [27].

The neurons of two successive layers are associated through weights. These weights determine the importance of a feature in predicting the target value. Fig. 2 depicts the DNN with four hidden layers. The weights initially are random but as the model gets trained, the weights get updated iteratively in order to learn and predict the final output. Since the initiation of advanced computational power and data storage, deep learning models have been adopted in almost spheres of our digital yet regular lifestyle. Deep learning model exists in all spectrum of the industry

verticals starting from healthcare to aviation, banking, retail services, telecommunications and many others as well.

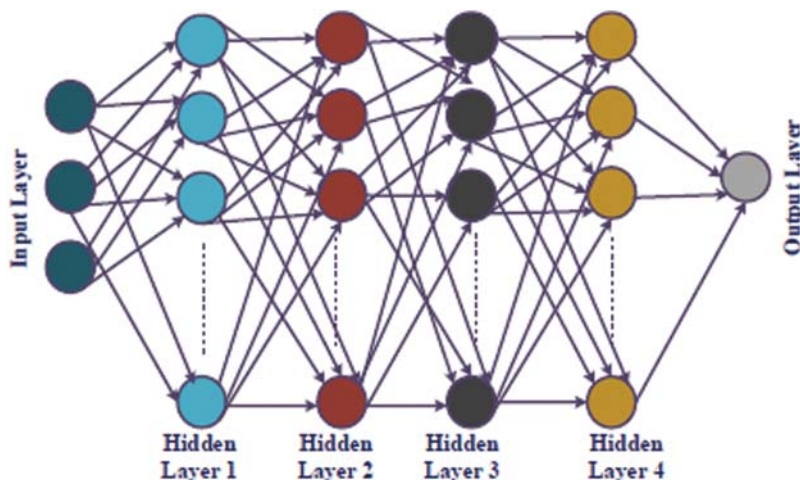


Figure 2: Deep neural network with multiple hidden layers

3.3 XGBoost

XGBoost is a decision tree based ensemble algorithm that uses the gradient boosting framework in machine learning. Although artificial neural network yields extremely good results in case of unstructured data, but in case of small-to-medium sized structured data, decision tree based algorithms are considered as the best. The algorithm is popularly used for its scalability which enables faster learning in cases of parallel and distributed learning ensuring efficient memory usage. XGBoost is basically an ensemble learning method wherein a systematic solution is offered to combine the prediction capability of multiple learners. The resultant model generates an aggregated output achieved from several models. These models which constitute the ensemble are called base learners which could be from the same or different learning algorithms. To be more specific Bagging and Boosting are two predominantly used ensemble learners. These two techniques are implemented in association with decision trees. Decision tree models generate results having high variances. On the contrary, bagging and boosting helps to reduce the variance in the learners. The various decision trees which are generated in parallel, form the basic learner for the Bagging technique [28]. The data is fed into these learners for training purposes and the final prediction is generated averaging all the output from these learners. In Boosting, the decision trees are built sequentially such that each subsequent tree helps to reduce the errors of the previous tree. The subsequent tree learns from its preceding tree, updates the residual error and likewise, the next tree in the sequence learns from the updated version of the residual. Thus XGBoost appears to be one of the most powerful, upfront and robust solution for data processing.

3.4 Linear Regression

Linear regression is one of the most common methods used in prediction analysis. The objective of regression is firstly to analyze the competence of a predictor variable in predicting an

outcome of the dependent variable; Secondly, identification of the specific variables contributing towards the outcome variable. Basically the regression helps in understanding the relationship between one dependent and multiple independent variable. The dependent variable in the regression model is called the outcome variable. The independent variables on the contrary are called predictor variables. The regression analysis thus helps in determining the capability and strength of the predictor variable thereby forecasts an effect and generates the trend for forecasting [29].

3.5 One Hot Encoding

Preparation of the dataset is an essential part of machine learning wherein encoding constitutes a major aspect of the same. The basic idea is to represent the data in a form which the computer understands. One hot encoding is one such label encoding technique which converts categorical values to a form which the computer understands. One hot encoding technique represents the categorical values as binary vectors. At the outset, the categorical values are mapped to integer values and then each of these integers are represented as a binary vector having all zero values except the index of the integer being marked as 1 [30,31].

3.6 Dataset Description

The dataset used in the study is a real world dataset consisting of Chicago Part District Beach Water Quality Automated Sensor Data. The Chicago Beach Park District had deployed various sensors at the beaches along Lake Michigan Lakefront. These sensors were deployed in six locations by the district administration to monitor information pertinent to various water quality attributes namely Water Temperature, Turbidity, Transducer Depth, and Wave Height, Wave Period and basic information like Beach Name and the time stamp. The data was collected on an hourly basis during summer time which ensured collection of 39.5 K instances of the data with 10 features [5].

3.7 Proposed Model

The proposed model involves use of the Chicago park beach dataset collected from the publicly available repository. One of the major contributions of the current work is to identify the optimal features that can be used to train the DNN. In order to extract the most important features without affecting the prediction accuracy of the proposed model, the MFO algorithm is used due to its ability of fast convergence. Also, it does not fall into the local minima trap easily, which is a crucial factor in identifying the optimal features. The steps involved in the proposed model are discussed as follows:

- a. **Pre-processing:** In this phase, the dataset is preprocessed for making it suitable for the prediction algorithm. As part of the pre-processing, treating of missing values, standardization and normalization of the dataset is performed. The missing values in the dataset are filled with the attribute mean to ensure that no data is lost by eliminating the instances of the missing values. The resultant dataset is subjected to one-hot encoding method wherein the main objective is to standardize the dataset converting all categorical values to numerical form making it suitable for the training of the DNN model. In order to fit all the values of the dataset into a common scale, standard scalar technique is used which successfully normalizes the data.

- b. **Feature Extraction:** The next step after pre-processing is the training of the DNN. Training the DNN with all the attributes have possibilities of resulting in poor performance of the model. There could be some attributes in the dataset which have minimal or no role in the prediction accuracy. On the contrary there could be some attributes which have negative impact on the performance of the DNN. Hence, dimensionality reduction/feature selection plays a vital role in the performance of the DNN or any machine learning model. In order to extract the most significant features of the IoT dataset, MFO algorithm is used in the present work. The preprocessed dataset is fed into to MFO optimization algorithm to select the optimal features or attributes from the dataset.
- c. **Training the DNN:** The dimensionally reduced dataset is then fed to the DNN model for the purpose of prediction. Several parameters of the DNN such as number of layers, optimization function, activation function, number of epochs etc. are chosen with the help of random search algorithm. 70% of the dataset is used for training the DNN and rest of the 30% dataset is used to test the performance of the DNN using several metrics namely root mean square, variance, etc.
- d. **Validating the model:** The results generated from the MFO-DNN model is further evaluated against the traditional state of the art ML models in order to analyze and justify its superiority. Fig. 3 depicts the proposed model for battery life prediction.

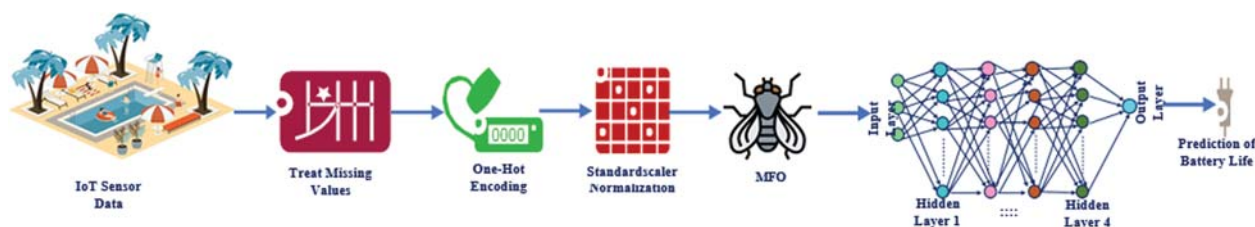


Figure 3: Proposed model for battery life prediction

4 Experimental Results and Discussion

The experimentation is carried out in Google colab, an online GPU platform by Google Inc. The programming language used for the purpose of experimentation is Python 3.7. The chicao park beach dataset generated from the randomly located IoT devices is used in the study collected from the publicly available Kaggle repository and implemented in Google colab. The dataset has higher variability, and hence required pre-processing. At the outset, the missing values were treated using the attribute mean technique. It is a known fact that ML and DNN can be implemented only on numerical dataset values. The original dataset consisted of non-numerical values and hence had to be subjected to One Hot Encoding technique to ensure its compatibility to be fed into the ML model. This standardized and pre-processed data was further normalized using StandardScaler technique. StandardScaler normalization method contributed to fit in all values of the attributes to a particular scale. Next, in order to select the optimal features that would significantly impact the class labels (Actual battery remaining life), MFO algorithm was used. This reduced dataset was then fed into a DNN model for prediction of remaining battery life of IoT devices. A total

of 70% of the dataset was used to train the ML models and the rest was used for testing the model. The results of the proposed MFO-DNN model were finally evaluated against the state of the art traditional ML models namely logistic regression (LR), XGBoost, DNN with PCA, LR with PCA, XGBoost with PCA, DNN with antlion optimization (ALO), LR with ALO, XGBoost with ALO, DNN with genetic algorithm (GA), LR with GA, XGBoost with GA, LR with MFO, and XGBoost with MFO. The metrics used in this work are mean absolute error, mean squared error, root mean squared error and test variance score. With an objective to choose the optimal parameters like number of layers, optimization function, activation function, etc. for the DNN model, grid search algorithm is used. The details of the parameters chosen are Optimizer: Adam optimizer, Number of epochs: 25, Batch size: 64, Number of layers: 5, Activation function at intermediate layers: Leaky ReLU, Activation function at output layer: softmax, Loss function: mean squared error. The number of parameters chosen at each dense layer in the DNN are summarized in the [Tab. 1](#).

Table 1: Battery life prediction model of DNN-MFO with dimensionality reduction

Layer	Type	Shape	Parameters
Dense 17	Dense	128	21423
Dense 18	Dense	256	30125
Dense 19	Dense	256	62178
Dense 20	Dense	256	62178
Dense 21	Dense	256	62178
Dense 22	Dense	128	29128
Dense 23	Dense	64	7128
Dense 24	Dense	1	56
Total parameters			274394
Trainable parameters			274394

[Fig. 4](#) depicts the prediction results attained by LR and its variants, namely, LR+PCA, LR+GA, LR+ALO, and LR+MFO. It is evident from the figure that LR with MFO yielded enhanced prediction results in comparison to the other variants of LR. Similarly, [Figs. 5](#) and [6](#) depict the results obtained by XGBoost, DNN and their variants i.e., in combinations with PCA, GA, ALO, and MFO respectively. It is further evident from this figure that ML algorithms, in combination with MFO for feature extraction have outperformed the other counterparts. It could thus be established that the proposed MFO-DNN has been successful in generating better results in comparison to the other models considered in the study. [Tab. 2](#) summarizes the results attained by all the considered models. [Tab. 3](#) highlights the enhanced performance of the proposed model against the other state-of-the-art models for the metrics mean absolute error, mean squared error, root mean squared error and test variance score. The results also conclude the fact that considering the fast convergence rate, error reductions in estimation of the global optima, MFO algorithm has been successful in attaining better prediction results utilizing significantly less training time.

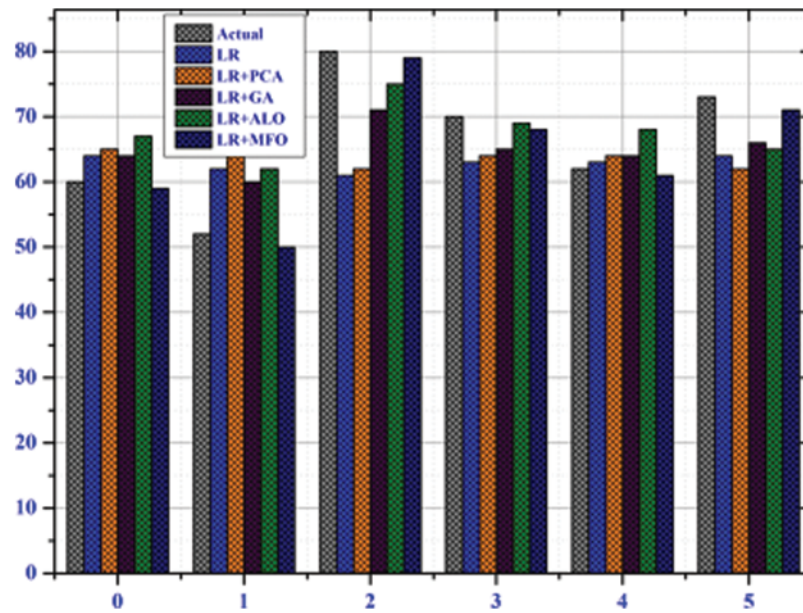


Figure 4: Remaining battery life vs. predicted remaining battery life using LR

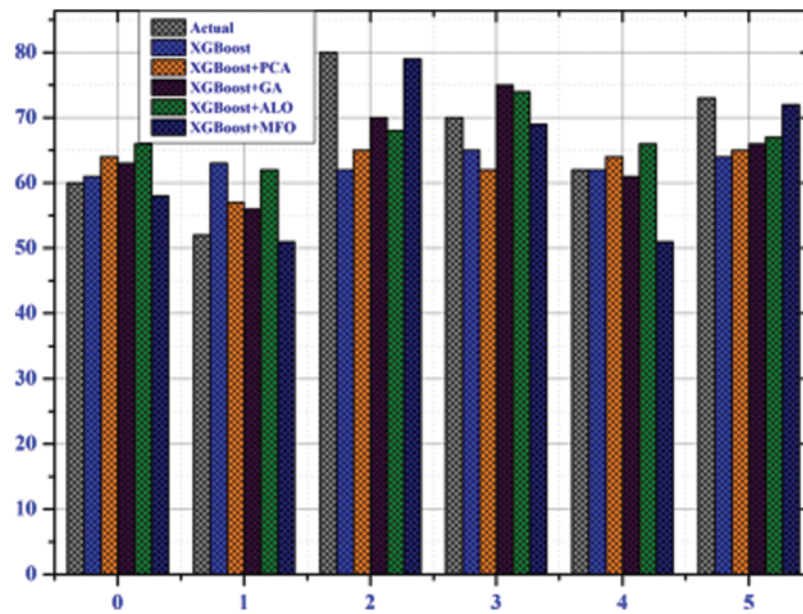


Figure 5: Remaining battery life vs. predicted remaining battery life using XGBoost

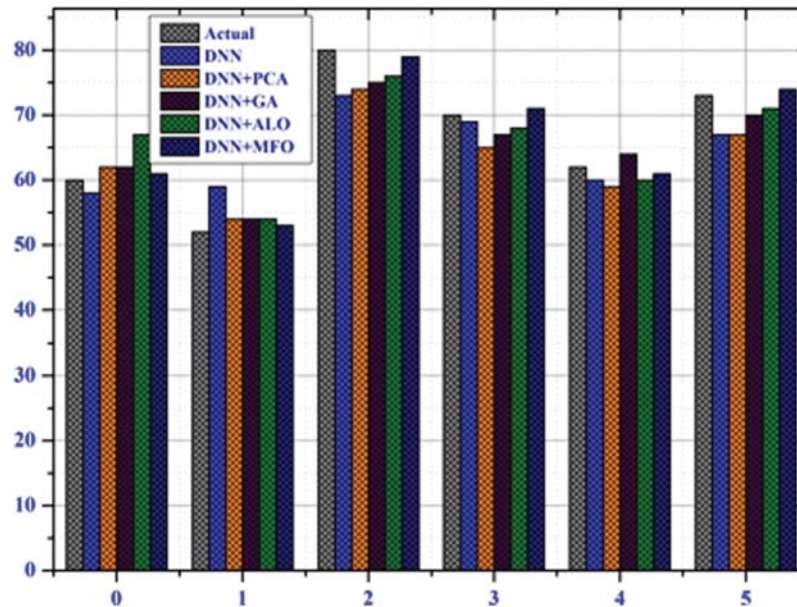


Figure 6: Remaining battery life vs. predicted remaining battery life using DNN

Table 2: Remaining battery life vs. predicted remaining battery life

Remaining battery life	Predicted remaining battery life														
	LR	LR+PCA	LR+GA	LR+ALO	LR+MFO	XGBoost	XGBoost+PCA	XGBoost+GA	XGBoost+ALO	XGBoost+MFO	DNN	DNN+PCA	DNN+GA	DNN+ALO	DNN+MFO
60	64	65	64	67	59	61	64	63	66	58	58	62	62	67	61
52	62	64	60	62	50	63	57	56	62	54	59	54	54	54	53
80	61	62	71	75	79	62	65	70	68	83	73	74	75	76	79
70	63	64	65	69	68	65	62	75	74	68	69	65	67	68	71
62	63	64	64	68	61	62	64	61	66	51	60	59	64	60	61
73	64	62	66	65	71	64	65	66	67	72	67	67	70	71	74

Table 3: Performance analysis of proposed model vs. existing models

Model used	Mean absolute error	Mean squared error	Root mean squared error	Test variance score
LR	988.72	1681700169.89	8.53	25 876 324.98
XGBoost	5.38	45.23	6.72	0.30
DNN	5.17	35.70	5.82	0.19
LR+PCA	5.99	78.37	8.85	0.21
XGBoost+PCA	5.25	46.69	6.83	0.28
DNN+PCA [32]	5.08	44.49	6.51	0.12
LR+GA	5.88	44.45	6.57	0.33
XGBoost+GA	5.98	33.56	5.58	0.45
DNN+GA	5.76	31.78	5.28	0.55
LR+ALO	5.98	31.33	5.15	0.35
XGBoost+ALO	5.55	31.78	5.25	0.42
DNN+ALO	5.68	32.86	5.60	0.56
LR+MFO	5.78	31.78	5.28	0.38
XGBoost+MFO	5.45	31.25	5.15	0.35
RF+PCA [25]	4.55	30.34	5.05	0.33
DNN+MFO (Proposed)	4.01	28.36	4.96	0.11

5 Conclusion and Future Scope

The Internet of things (IoT) have evolved to create an immense impact in all spheres of personal, societal and professional lives. Regardless of the area of implementation, be it a wearable device that sends information to smart phones or network connected motion detection devices in a smart home alarm system, battery life optimization remains to be the top prioritized requirement. The present study has used the Chicago park beach dataset collected from the publicly available repository for the prediction of the battery life. Selection of the optimal features from this dataset acts as a primary task of the study since the record is of high dimensionality. The missing values in the dataset are treated using the attribute mean method. The categorical values in the dataset are converted to numerical values using the One hot encoding technique and the Standard Scaler technique is implemented for data normalization. The optimal features in the dataset are selected using the MFO algorithm, which are fed into the DNN model for battery life prediction. The results generated from the proposed MFO-DNN model are compared with the traditional state-of-the-art ML models and the results justify the superiority of the proposed model considering the evaluation metrics—Mean Absolute Error, Mean Squared Error, Root Mean Squared Error and Test Variance Score. The current work can be extended by using big-data analytics and blockchain to address the scalability and security issues [33–42].

Funding Statement: The authors are grateful to the Raytheon Chair for Systems Engineering for funding.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] A. D. Dwivedi, G. Srivastava, S. Dhar and R. Singh, “A decentralized privacy-preserving healthcare block chain for IoT,” *Sensors*, vol. 19, no. 2, pp. 326, 2019.
- [2] G. Srivastava, R. M. Parizi, A. Dehghantanha and K. K. R. Choo, “Data sharing and privacy for patient IoT devices using block chain,” in *Int. Conf. on Smart City and Information*, Guangzhou, China, Springer, pp. 334–348, 2019.
- [3] M. Reddy and M. R. Babu, “Energy efficient cluster head selection for internet of things,” *New Review of Information Networking*, vol. 22, no. 1, pp. 54–70, 2017.
- [4] G. Ma, Y. Zhang, C. Cheng, B. Zhou, P. Hu *et al.*, “Remaining useful life prediction of lithium-ion batteries based on false nearest neighbors and a hybrid neural network,” *Applied Energy*, vol. 253, pp. 113626, 2019.
- [5] C. P. District, “Chicago beach sensor life prediction,” 2017. [Online]. Available: <https://www.chicagoparkdistrict.com/parks-facilities/beaches> (Accessed on November 12, 2020).
- [6] J. Azar, A. Makhoul, M. Barhamgi and R. Couturier, “An energy efficient IoT data compression approach for edge machine learning,” *Future Generation Computer Systems*, vol. 96, pp. 168–175, 2019.
- [7] M. Zhou, A. Gallegos, K. Liu, S. Dai and J. Wu, “Insights from machine learning of carbon electrodes for electric double layer capacitors,” *Carbon*, vol. 157, pp. 147–152, 2020.
- [8] M. Lucu, E. Martinez-Laserna, I. Gandiaga, K. Liu, H. Camblong *et al.*, “Data-driven nonparametric li-ion battery ageing model aiming at learning from real operation data—part A: Storage operation,” *Journal of Energy Storage*, vol. 30, pp. 101409, 2020.
- [9] Y. Liu, B. Guo, X. Zou, Y. Li and S. Shi, “Machine learning assisted materials design and discovery for rechargeable batteries,” *Energy Storage Materials*, vol. 31, pp. 434–450, 2020.
- [10] V. M. Suresh, R. Sidhu, P. Karkare, A. Patil, Z. Lei *et al.*, “Powering the IoT through embedded machine learning and LORA,” in *IEEE 4th World Forum on Internet of Things*, Singapore, pp. 349–354, 2018.

- [11] Y. Choi, S. Ryu, K. Park and H. Kim, "Machine learning-based lithium-ion battery capacity estimation exploiting multi-channel charging pro-files," *IEEE Access*, vol. 7, pp. 75143–75152, 2019.
- [12] C. Vidal, P. Malysz, P. Kollmeyer and A. Emadi, "Machine learning applied to electrified vehicle battery state of charge and state of health estimation: State-of-the-art," *IEEE Access*, vol. 8, pp. 52796–52814, 2020.
- [13] K. Liu, Y. Shang, Q. Ouyang and W. D. Widanage, "A data-driven approach with uncertainty quantification for predicting future capacities and remaining useful life of lithium-ion battery," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 4, pp. 1–1, 2020.
- [14] A. S. M. J. Hasan, J. Yusuf and R. B. Faruque, "Performance comparison of machine learning methods with distinct features to estimate battery SoC," in *IEEE Green Energy and Smart Systems Conf.*, Long Beach, CA, USA, pp. 1–5, 2019.
- [15] A. Ashiquzzaman, H. Lee, T. Um and J. Kim, "Energy-efficient IoT sensor calibration with deep reinforcement learning," *IEEE Access*, vol. 8, pp. 97045–97055, 2020.
- [16] G. Thippa Reddy, N. Khare, S. Bhattacharya, S. Singh, M. Praveen Kumar Reddy *et al.*, "Deep neural networks to predict diabetic retinopathy," *Journal of Ambient Intelligence Humanized. Computer*, 2020. <https://doi.org/10.1007/s12652-020-01963-7>.
- [17] R. M. Swarna Priya, M. Praveen Kumar Reddy, M. Parimala, S. Koppu, G. Thippa Reddy *et al.*, "An effective feature engineering for DNN using hybrid PCA-GWO for intrusion detection in IoMT architecture," *Computer Communications*, vol. 160, pp. 139–149, 2020.
- [18] G. Thippa Reddy, R. Dharmendra Singh, M. Praveen Kumar Reddy, K. Lakshmana, S. Bhattacharya *et al.*, "A novel PCA-whale optimization-based deep neural network model for classification of tomato plant diseases using GPU," *Journal of Real-Time Image Processing*, pp. 1–14, 2020. <https://doi.org/10.1007/s11554-020-00987-8>.
- [19] S. Bhattacharya, M. Praveen Kumar Reddy, P. Quoc-Viet, G. Thippa Reddy, C. Chiranji Lal *et al.*, "Deep learning and medical image processing for coronavirus (COVID-19) pandemic: A survey," *Sustainable Cities and Society*, vol. 65, pp. 102589, 2020.
- [20] M. Iyapparaja and B. Sharma, "Augmenting SCA project management and automation framework," *IOP Conf. Series: Materials Science and Engineering*, Tamil Nadu, India, vol. 263, pp. 1–8, 2017.
- [21] M. Sathish Kumar and M. Iyapparaja, "Improving quality-of-service in fog computing through efficient resource allocation," *Computational Intelligence*, vol. 36, no. 4, pp. 1–21, 2020.
- [22] B. K. Tripathy and A. Mitra, "Some topological properties of rough sets and their applications," *International Journal of Granular Computing, Rough Sets and Intelligent Systems*, vol. 1, no. 4, pp. 355–369, 2010.
- [23] B. K. Tripathy, R. K. Mohanty and T. R. Sooraj, "On intuitionistic fuzzy soft set and its application in group decision making," in *Int. Conf. on Emerging Trends in Engineering, Technology and Science*, Pudukkottai, India, IEEE, pp. 1–5, 2016.
- [24] S. Siva Rama Krishnan, M. Alazab, M. K. Manoj, A. Bucchiarone, C. Chiranji Lal *et al.*, "A framework for prediction and storage of battery life in IoT devices using DNN and block chain," *arXiv preprint arXiv: 2011.01473*, 2020.
- [25] P. K. Reddy Maddikunta, G. Srivastava, T. Reddy Gadekallu, N. Deepa and P. Boopathy, "Predictive model for battery life in IoT networks," *IET Intelligent Transport Systems*, vol. 14, no. 11, pp. 1388–1395, 2020.
- [26] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," *Knowledge-Based Systems*, vol. 89, pp. 228–249, 2015.
- [27] P. Schramowski, W. Stammer, S. Teso, A. Brugger, F. Herbert *et al.*, "Making deep neural networks right for the right scientific reasons by interacting with their explanations," *Nature Machine Intelligence*, vol. 2, no. 8, pp. 476–486, 2020.
- [28] A. B. Parsa, A. Movahedi, H. Taghipour, S. Derrible and A. K. Mohammadian, "Toward safer highways, application of XGBoost and shap for real-time accident detection and feature analysis," *Accident Analysis & Prevention*, vol. 136, pp. 105405, 2020.

- [29] M. Ali, R. Prasad, Y. Xiang and R. C. Deo, "Near real-time significant wave height forecasting with hybridized multiple linear regression algorithms," *Renewable and Sustainable Energy Reviews*, vol. 132, pp. 110003, 2020.
- [30] H. ElAbd, Y. Bromberg, A. Hoarfrost, T. Lenz, A. Franke *et al.*, "Amino acid encoding for deep learning applications," *BMC Bioinformatics*, vol. 21, no. 1, pp. 1–14, 2020.
- [31] R. Kaluri, D. S. Rajput, Q. Xin, K. Lakshmana, S. Bhattacharya *et al.*, "Roughsets-based approach for predicting battery life in IoT," *Intelligent Automation & Soft Computing*, vol. 27, no. 2, pp. 453–469, 2021.
- [32] T. Reddy, S. P. RM, M. Parimala, C. L. Chowdhary, S. Hakak *et al.*, "A deep neural networks based model for uninterrupted marine environment monitoring," *Computer Communications*, vol. 157, pp. 64–75, 2020.
- [33] M. Tang, M. Alazab and Y. Luo, "Big data for cybersecurity: Vulnerability disclosure trends and dependencies," *IEEE Transactions on Big Data*, vol. 5, no. 3, pp. 317–329, 2017.
- [34] G. T. Reddy, M. P. K. Reddy, K. Lakshmana, R. Kaluri, D. S. Rajput *et al.*, "Analysis of dimensionality reduction techniques on big data," *IEEE Access*, vol. 8, pp. 54776–54788, 2020.
- [35] R. Kumar, P. Kumar, R. Tripathi, G. P. Gupta, T. R. Gadekallu *et al.*, "SP2F: A privacy-preserving framework for smart agricultural unmanned aerial vehicles," *Computer Networks*, vol. 187, pp. 107819, 2021.
- [36] U. Bodkhe, S. Tanwar, K. Parekh, K. P. Khanpara, S. Tyagi *et al.*, "Blockchain for industry 4.0: A comprehensive review," *IEEE Access*, vol. 8, pp. 1–37, 2020.
- [37] A. D. Dwivedi, L. Malina, P. Dzurenda and G. Srivastava, "Optimized blockchain model for internet of things based healthcare applications," in *2019 42nd Int. Conf. on Telecommunications and Signal Processing*, Budapest, Hungary, pp. 135–139, 2019.
- [38] N. Deepa, Q. V. Pham, D. C. Nguyen, S. Bhattacharya, T. R. Gadekallu *et al.*, "A survey on blockchain for big data: Approaches, opportunities, and future directions," *arXiv preprint arXiv: 2009.00858*, 2020.
- [39] T. R. Gadekallu, M. K. Manoj, S. Krishnan, N. Kumar, S. Hakak *et al.*, "Blockchain based attack detection on machine learning algorithms for IoT based E-health applications," *arXiv preprint arXiv: 2011.01457*, 2020.
- [40] S. Venkatraman and M. Alazab, "Use of data visualisation for zero-day malware detection," *Security and Communication Networks*, vol. 2018, Article ID 1728303, 2018. <https://doi.org/10.1155/2018/1728303>.
- [41] P. K. Sharma, N. Kumar and J. H. Park, "Blockchain-based distributed framework for automotive industry in a smart city," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4197–4205, 2018.
- [42] M. J. Chowdhury, M. S. Ferdous, K. Biswas, N. Chowdhury, A. S. M. Kayes *et al.*, "A comparative analysis of distributed ledger technology platforms," *IEEE Access*, vol. 7, pp. 167930–167943, 2019.