JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Industrial IoT Machine Learning-Based Predictive Maintenance

[1] Ms. Vennila P, [2] Dr. Maniraj V
[1]Research Scholar, [2]Associate Professor
[1]Department of Computer Science,
[1] A. Veeriya Vandayar Memorial Sri Pushpam College (Autonomous),
Poondi, Thanjavur, 613503, Tamil Nadu, India.
Affiliated to Bharathidasan University

Abstract:- **The use of predictive maintenance Machine learning techniques aid systems or machines in lowering the occurrence of certain types of machine failures via prediction and the use of specific methods. An essential tactic for improving the efficiency and reliability of industrial equipment and optimizing maintenance operations is predictive maintenance (PdM). Machine learning-based predictive maintenance helps businesses reduce unscheduled downtime, maintenance expenses, and operational efficiency by identifying and fixing potential equipment issues in advance.**

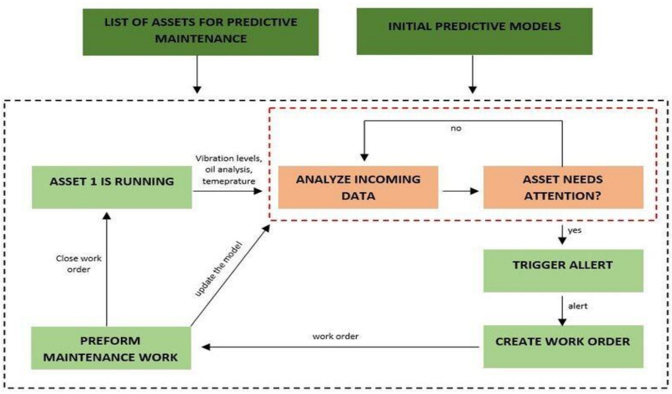
Keywords:- *Machine Learning, Jupiter, Proactive, Nonintrusive, Vibration, and Predictive Maintenance*.

I. INTRODUCTION

A survey of the literature on industrial IoT (IoT) and the use of DL and ANN to generate fast predictions for maintenance and other IoT applications is presented in this work. An method to plant and equipment maintenance called predictive maintenance (Pd.M.) seeks to anticipate possible problems and stop them before they arise. By accurately predicting when failures will occur, the objective is to prolong service life while preventing them. Forecasts are based on current conditions and accumulated knowledge (Mohammadi, Al-Fuqaha, Sorour, & Guizani, 2018).

IoT setups generate vast amounts of data. Big Data features like the "5V's"—volume, variety, velocity, variability, and veracity—are essential for data analytics on such massive datasets. The complexity and volume of data overwhelm current data processing techniques. Due to shifting data streams and real-time requirements, the existing high needs will only increase. Sensors will often continuously deliver streams of data. Data streams are ongoing data sources that are typically produced quickly. In fully automated industrial environments, the capacity to receive data in real-time and react promptly is essential. Machine-to-machine (M2M) communication is crucial in Internet of Things scenarios. By sharing information and communicating with one another, smart sensors and devices are able to sense their environment and react quickly. The feature of capturing a picture of the entire data set and doing calculations with uncertain reaction times conflicts with the need for real-time communication and continuously flowing data streams in these IoT contexts. The use of self-adaptive algorithms is essential to meet such demands. These algorithms improve and learn their models continuously. These algorithms must also function exceptionally well and behave in real time. This is the case regardless of whether they are using Internet of Things devices, fog and edge systems, or reliable cloud systems (Mocanu, Nguyen, Gibescu, & Kling, 2016).

Fig1 Maintenance Workflow



A narrative review is used as the methodological methodology in this study. The authors decided to concentrate on the qualitative components of the research rather than choosing the articles to be analyzed quantitatively. Papers that compiled the most recent results from studies on DL-based fast predictions in IoT were given priority. The themes of DL in (I)IoT are covered in a large number of publications. To the best of our knowledge, Pd.M. regarding DL and (I)IoT is not covered in any of the literature currently in publication.

This overview classifies a number of different deep learning approaches that have been considered for use in business and the Internet of Things. Furthermore, it talks about real-time processing and data streams related to the previously mentioned deep learning approaches. The strategies that are analyzed and categorized are those that aim to improve the stream-processing and real-time capabilities of the different approaches that are covered in the evaluated publications. The ability of the approaches that have been put out to provide forecasts is given special attention. A summary and a perspective on future developments are provided in the study's final part (Pani, Pattnaik, & Pattanayak, 2024).

II. MACHINE LEARNING APPROACHES IN INDUSTRIAL INTERNET OF THINGS (IOT)

This part begins with a brief introduction to ANN and DL. After that, a taxonomy of the various DL techniques that have been detailed for use in the Internet of Things and industry will be provided. Based on the theoretical approaches, application domains, and the advantages and disadvantages of each system, the requirements of Pd.M. in IoT settings will be categorized. The evaluated publications addressed a wide range of topics, such as deep learning techniques in Cyber-Physical Systems (CPS), Internet of Things (IoT), and Industry 4.0 (I4.0), as well as real-time and data stream processing.

One might consider Deep Learning (DL) to be a subclass of Machine Learning (ML), as opposed to ML, which is a subfield of Artificial Intelligence (AI). A class of effective artificial neural networks (ANNs) with several layers (hidden layers) is what many refer to as deep learning. Apart from facilitating other attributes like automated feature extraction or unsupervised learning, the large number of layers and neurons makes it possible to abstract more complex issues. Neural network technologies include Deep Neural Networks (DNN), Deep Belief Networks (DBN), and Recurrent Neural Networks (RNN) (Sane, 2020).

The biological neural network present in mammalian brains is intended to be replicated by the artificial neural network (ANN). Neurons, also known as nodes in ANNs, and the connections that exist between them comprise a neural network (ANN). The system uses the data that is fed into it to create nonlinear output data that is layered by the nodes. The output of one node may be moved to the input of another node with the aid of the connections that exist between the nodes. The weights assigned to each link determine the importance of the signal that is transmitted. This is similar to how a threshold function controls a neuron's (node's) output signal in biological neural networks. An artificial neural network (ANN) requires that each weight be initialized to a value, which is frequently only a crude estimate. To make sure the network is balanced and legitimate, those weights are changed holistically during the training process, adhering to a preset learning rate. "Connections developing over time with training" is another phrase that's frequently used to explain this process. Over the course of the more than 50 years that ANNs have been in use, a number of techniques have been developed.

Examples of deep learning models that could be applied to Internet of Things (IoT) applications include Auto-encoder (AE), Recurrent Neural Network (RNN), Restricted Boltzmann Machine (RBN), Deep Neural Network (DBN), Long Short-term Memory (LSTM), Convolutional Neural Network (CNN), Variational Auto-encoder (VAE), Generative Adversarial Network (GAN), and Ladder Net. Generative techniques (AE, RBM, DBN, VAE), discriminative approaches (RNN, LSTM, CNN), and hybrid approaches (GAN, Ladder Net), which combine the two previously mentioned approaches, are the three main groups into which deep learning models fall. This classification mostly relates to the learning method being employed, with discriminative approaches essentially sticking to the notion of supervised learning and generative methods essentially adhering to the concept of unsupervised learning. When choosing a deep learning strategy, the underlying learning method is crucial. This is in addition to determining the complexity, or the required number of layers. The categorization of methods as either discriminative or generative, which was chosen by, is widely present in many later investigations. Several DL models are categorized based on their performance in Internet of Things applications. The ability to work with (partially) unlabeled data (feature extraction, feature discovery), the size of the required training dataset, the ability to reduce the dimensionality of the data, the ability to deal with noisy data and time series data, and the general performance classification of these characteristics are some of the pertinent characteristics that are mentioned. For the reduction of high-dimensional data and for handling unlabeled data, it is advised to combine recurrent neural networks (RNN), deep neural networks (DBN), and artificial intelligence (AE). DBN and AEs are frequently employed as an upfront layer that provides categorized data to a subsequent RNN if the system is meant to provide predictions, as is the case with PdM systems (Elkateb, Métwalli, Shendy, & Abu-Elanien, 2024).

Since recurrent neural networks (RNNs) yield robust results when the data is evolving sequentially, they are recommended for use in situations involving spatial-temporal data, such as mobility data. However, if the data also includes long-term dependencies, RNNs are not a good choice. This is a result of RNNs' inability to recall previous states and results. A technique for handling sequential data streams originating from human mobility and transportation transition models with long-term dependencies (behaviors) is described in the paper. The proposed solution consists of a combination of LSTM and RNN in the form of a particular RNN architecture. In addition to managing long-term dependencies, the LMST combines these two functions with labeling and predictive capabilities. In order to handle data streams or time-series data with long-term dependencies (such specific behaviors or the wear and tear of machinery), a large number of other studies combine recurrent neural networks (RNN) with long short-term memory (LSTM).

Selecting the best artificial neural network (ANN) to provide predictions from time-series and data streams is the subject of the study "IoT Data Analytics Using Deep Learning." Trends and forecasts are retrieved using a combination of LSTM and Naive Bayes models, and those trends and predictions are validated concurrently through anomaly detection. The linear support vector machine (LSTM) is in charge of making predictions on data streams, as opposed to the Naive Bayes model, which is in charge of identifying anomalies based on the LSTM's output. The fact that simple feedforward artificial neural networks (FNN) that employ standard backpropagation (BP) for training, such Single-layer Perceptrons (SLP) and Multi-layer Perceptrons (MLP), are frequently not a good choice is also taken into consideration in this research. This is a result of these neural networks' poor performance on data streams with long-term dependencies and in complex scenarios. This is especially crucial to remember if the data streams in question are time series data and the model's goal is to predict future events or trends. Over time, dependencies between time-series data and data streams frequently emerge. These dependencies offer important insights and are typical for data gathered by the Internet of Things. Under the assumption that input data is independent of output data, data is expected to flow linearly through the layers of simple artificial neural networks (ANNs). Because of this, it is impossible to remember the input and output states (or results) that have previously occurred. Data from the past that is linked to data from the present is problematic. When used to time-series and data streams, RNN may yield better results than other techniques. The connections between nodes in a recurrent neural network (RNN) are in the form of loops or sequences, which enables the recall of previous states. The only thing that is often kept in mind to stop gradient outbursts is a view state. For this reason, only short-term dependencies are considered. In complex Internet of Things scenarios, it is advised to use long short-term memory (LSTM) to detect long-term dependencies in the data source. One type of RNN that adds memory units to the network is called an LSTM. These memory units have the capacity to forget less important states in addition to remembering important ones (Rippel, Lutjen, & Freitag, 2017).

It is crucial to recognize that more intelligent systems are needed to produce accurate predictions regarding the future use of energy in order to predict how energy systems will behave in the context of smart grids. An option that shows promise is ANN-based prediction approaches, as stated in the study "Deep Learning for estimating building energy consumption." This is due to the fact that these techniques are capable of handling large amounts of highly non-linear time series data that come from a variety of heterogeneous data sources (like SmartMeter) and contain a significant amount of uncertainty (unlabeled data). Throughout the research article, they used a synthetic benchmark dataset to compare two different approaches to the RBN: the Conditional Restricted Boltzmann Machine (CRBM) and the Factored Conditional Restricted Boltzmann Machine (FCRBM). The experiment's authors have determined that FCRBN outperforms RNN, Support Vector Machine (SVM), and CBRM in terms of performance because it includes a factored conditional history layer. Two layers make up RBMs, a type of stochastic artificial neural network (ANN): a visible layer and a hidden layer. Every possible value that can be found in the input data is represented by a node in the visible layer of a recurrent neural network (RBM), whilst the hidden layer is in charge of establishing the value categories. An RBN is useful for feature classification, feature extraction, and complexity reduction (by identifying which characteristics are most important) since each visible layer node in an RBM is connected to every hidden layer node. RBMs might be packed for DL. The RBM is extended with a conditional history layer, or CRBM, which helps the RBN find long-term correlations in time-series data.

This is done in an effort to reduce the overall number of possible compositions.

Another paper published in the field of energy management likewise highlights DL's highly effective prediction capabilities. The use of AE and LSTM to predict how much power solar systems will produce is explained in this article. The accuracy attained by a combination of AE and LMST (Auto-LSTM) is compared to that of a physical model and other neural networks, specifically MLP. Data for the benchmark was gathered from 21 real solar power plants, while the benchmark itself was acquired using an experimental setup described in. The average root-mean-square deviation (RMSD), average mean absolute error (MAE), average absolute deviation (Abs. Dev.), average BIAS, and average correlation are used to set benchmarks. The measured results show that the results from all ANN- and DL-based models are significantly better than those from the physical model. Auto-LSTM is the best choice for this situation and data set when it comes to artificial neural networks (ANN) and deep learning (DL) models. The ability to extract features from unlabeled data is one of the most crucial aspects that is considered when it comes to making predictions.

The paper "An enhancement deep feature fusion method for rotating machinery fault diagnosis" demonstrates AEs' proficiency in feature extraction and feature learning. The study explains the deep feature fusion technique, which reduces the impact of background noise while also enhancing feature learning capabilities. Deep AE (noise reduction) and Contractive AE (better feature recognition) are stacked to achieve this. An essential part of the forecasting process.

III. MACHINE LEARNING-BASED FAST PREDICTIONS

A number of Internet of Things applications require processing in real time. For example, in a PdM system, a high latency could lead to unintentional reactive maintenance because there would be insufficient advance time to plan the maintenance procedures. The required speed at which real-time processing must be carried out is greatly influenced by the application scenario. The phrase "real-time" refers to the amount of time measured in microseconds in the context of micro-manufacturing systems, which generate vast quantities of micro components at a quick pace. shows how increasing processing speed with defect detection and PdM systems can lower the rejection rate of produced micro components. The term "real-time" can sometimes refer to the passing of seconds, minutes, or even hours. For example, the frequency of data availability in PdM applications for offshore wind turbines is mostly minutes and hours (Xie, Wu, Liu, & Li, 2022).

The article "Metro Density Prediction with Recurrent Neural Network on Streaming CDR Data" details the development of a real-time crowd prediction system for public transit. This system combines parallel streaming analytical programming with a weight-sharing recurrent neural network. A robust neural network model with a high learning capacity, for instance, offers a wide range of new insights, but this runs counter to the requirement for a fast reaction time. Real-time analysis is required to have a quick reaction time to emergent events, such as admission records at metro stations mixed with data from telecommunication. There are three steps involved in achieving this goal, which are as follows: A) the adoption of a Recurrent Neural Network (RNN) model; b) the application of methods for parallelizing RNNs; and c) the employment of parallel streaming analytical algorithms as a component of a cloud-based stream processing platform in order to improve its ability to function on data streams. In the project described, each metro station is represented by an independent recurrent neural network (RNN). Shared layers are constructed to dynamically share weights from stations in similar "situations" (e.g., a downtown station during rush hour). This makes it possible for many models to share weights. Another advantage of weight-sharing is the capacity to co-train concurrently (Gensler, 2021).

For effective data analysis, recurrent neural networks (RNNs) and their numerous variations are also recommended. In comparison to other models, RNNs may perform better, especially when used with increasingly prevalent sensor data types including time-series, serial, and stream data. This type of sensor data dominates the great majority of PdM applications.

In order to build and permanently modify models on vast amounts of data, including individual behavior, geographical and temporal characteristics, and transportation capacity, it is essential to possess the abilities of real-time processing and real-time learning. This work describes a deep LSTM learning architecture that can perform several tasks. The basic idea behind this strategy is to use multiple LSTM tasks that are separated by their respective domains (e.g., a separate job for mobility and transportation mode prediction) instead of using a composite feature vector. This architecture is used for parallel learning, and the necessary insights are combined with the findings. It is crucial that automotive assistance technologies, such traffic sign recognition, deliver accurate findings with little latency.

The paper describes how to implement DNN within the parameters of this topic. In addition to receiving entirely unlabeled input (raw photographs), the system model is updated continuously through online learning. For image recognition, a CNN with nine layers is employed. Max-pooling layers are alternatingly combined with convolutional layers to improve the system's overall performance. The convolutional layers carry out the convolution operation using the 2D input pixel maps as a guide. The max-pooling layer acts as a pre-processor between two convolutional layers, converting the output of one convolutional layer into the input of a subsequent convolutional layer. It accomplishes this by eliminating regions in the pixel mappings that overlap. This eliminates the requirement for the convolutional layers' infamously challenging and time-consuming redundant processing. The technique described in the conventional method is called Multi-Column DNN (MCDNN).

The paper describes a system that focuses on real-time methods for traffic sign recognition and detection. Parallel processing is crucial because multiple traffic signs must be identified simultaneously. This is the main area of focus. According to Song, Kanasugi, and Shibasaki (2016), this technique similarly uses CNN for image processing and combines it with AdaBoost to improve performance and parallel GPU processing.

Because LSTM models have memory cells, they are a good fit for data with long-term dependencies. If the data structure allows the construction of groups of entities and the separation of individual entities with their behavior, then processing each entity and each group with its neural network will probably be possible.

This creates new processing opportunities by allowing the distinct neural networks to operate in parallel. The outputs of each neural network that is processed separately and in parallel are frequently received by an aggregating layer. The ultimate result is then produced by combining all of the outputs in this layer. The author of the paper "A Hierarchical Deep Temporal Model for Group Activity Recognition," describes how to recognize a number of situations that could occur during a volleyball match. Each player has a single LSTM model that uses long-term dependencies to recall his previous actions during the match and anticipate future conduct. Each and every situation that arises throughout the game is then modeled as a team of players. All participating players' LSTM models are subordinated to those of a specific scenario in the hierarchical arrangement of the LSTMs. Based on the photos, CNN is used to compile details about the scenes and player behavior.

Altering the configuration of layers and connections is one of the topics covered in the paper "Simulation of Maintenance Activities for Micro-Manufacturing Systems." The reason for this is the demands made on real-time processing methods. Fully linked computer systems, where each node in one layer is connected to every other node in the layer below it, are capable of solving complex problems, but they also require a large amount of processing power. One strategy to reduce a deep learning network's complexity and, thus, its computing demand without appreciably sacrificing accuracy is to eliminate any connections that have no discernible impact on the result. You may talk about max-pooling layers, batch normalization, and transfer learning as additional performance-enhancing strategies in addition to dropout.

The research "An Analysis of Deep Neural Network Models for Practical Applications" makes the case that a large number of the deep learning models described in the literature are just not suitable for use in real-world scenarios. For example, this can be attributed to their long processing times or excessive electricity use. In his work, he argues that since performance issues are significant factors in real-world deep learning applications, they ought to receive more attention. This study compares the accuracy, memory footprint, parameters, operations count, inference time, and power consumption of fourteen different deep learning projects, including Alex Net and Google Net. The study discussed in this article shows that even a little gain in accuracy can lead to a large increase in processing power and calculation time. It is highly recommended that each DL project have a maximum energy usage and that the accuracy be modified to reflect that restriction.

III. CONCLUSION

This paper provides a narrative evaluation of selected literature that uses deep learning techniques to produce quick predictions of maintenance issues in the industrial internet of things (IoT) space. The papers state that the application of DL in PdM and the Internet of Things is a significant topic in the commercial sector. Many different applications are currently in use in practice, and new and improved versions of these apps are constantly being developed.

It is a frequent method that has been documented to mix many deep learning models to combine their different advantages and capabilities in a single application. Furthermore, it has been demonstrated in a number of application scenarios that real-time processing of complex data and data streams is necessary. PdM is specifically one of the applications for predictions that fit under this category. To increase the real-time capacity, parallel deep learning network concepts that use intermediate layers or a final aggregate layer to simplify the system are frequently employed. While there are many efforts in the field of real-time processing of deep learning models, some voices criticize the overemphasis on accuracy. These voices are promoting lightweight applications that are appropriate for real-world use and a greater emphasis on performance. The majority of studies concur that this topic still needs a great deal more research.

REFERENCES

[1] Elkateb,S.,Métwalli,A.,Shendy,A.,&Abu- Elanien, A. E. B. (2024). Machine learning and IoT – Based predictive maintenance approach for industrial applications. *Alexandria Engineering Journal*, *88*, 298–309. https://doi.org/10.1016/j.aej.2023.12.065

[2] Gensler.(2021,February28). Deep Learning for solar power forecasting — An approach using Auto Encoder and LSTM Neural Networks | IEEE Conference Publication | IEEE Xplore. Retrieved February28, 2024, from ieeexplore.ieee.org website: https://ieeexplore.ieee.org/document/7844673

[3] Mocanu,E., Nguyen,P.H., Gibescu,M., & Kling, W. L. (2016). Deep learning for estimating building energy consumption. *Sustainable Energy, Grids and Networks*, *6*, 91–99. https://doi.org/10.1016/j.segan.2016.02.005

[4] Mohammadi, M., Al-Fuqaha, A., Sorour, S., & Guizani, M. (2018). Deep Learning for IoT Big Data and Streaming Analytics: A Survey. *IEEE Communications Surveys & Tutorials*, *20*(4), 2923– 2960. https://doi.org/10.1109/comst.2018.2844341

[5] Pani,S., Pattnaik,O., & Pattanayak,B.K. (2024). Predictive Maintenance in Industrial IoT Using Machine Learning Approach. *International Journal of Intelligent Systems and Applications in Engineering*, *12*(14s), 521–534. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/4689/ 3363

[6] Rippel, D., Lutjen, M., & Freitag, M. (2017). *Simulation of maintenance activities for micro- manufacturing systems by use of predictive quality control charts*. 3780–3791. https://doi.org/10.5555/3242181.3242511

[7] Sane,A. (2020,February 28). *Machine learning for predictive maintenance of industrial machines using IoT sensor data | IEEE Conference Publication*| IEEEXplore. Retrieved February 28, 2024, from ieeexplore.ieee.org

[8] Song, X., Kanasugi, H., & Shibasaki, R. (2016). *Deep Transport: Prediction and Simulation of Human Mobility and Transportation Mode at a Citywide Level*. Retrieved from https://www.ijcai.org/Proceedings/16/Papers/372.pdf

[9] Xie,X.,Wu,D.,Liu,S., & Li,R.(2022). *IoTData Analytics using Deep learning.* Retrieved from https://arxiv.org/pdf/1708.0385