Predictive Maintenance using Machine Learning in Industrial IoT

Article *in* International Journal of Innovative Science and Research Technology · April 2024

DOI: 10.38124/ijisrt/IJISRT24MAR984

Volume 9, Issue 3, March – 2024 International Journal of Innovative Science and Research Technology
ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR984

Predictive Maintenance using Machine Learning in Industrial IoT

Jesu Narkarunai Arasu Malaiyappan[1]
Meta Platforms Inc, USA

Gowrisankar Krishnamoorthy[2]
HCL America, USA

Suhas Jangoan[3]
Zendesk, USA

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

This study presents a literature review on the topic of industrial IoT (IoT) and the use of DL and ANN to make quick predictions for use in maintenance and other IoT applications. Predictive maintenance (Pd.M.) is an approach to plant and equipment upkeep that aims to foresee potential faults and prevent them from happening. The goal is to maximize the service life while simultaneously avoiding breakdowns by timely forecast of when they will occur. Accumulated knowledge and present circumstances form the basis of the forecasts (Mohammadi, Al-Fuqaha, Sorour, & Guizani, 2018).

The data generated by IoT settings is enormous. Data analytics on such large datasets necessitates the use of Big Data characteristics, such as the "5V's": volume, variety, velocity, variability, and veracity. Current data processing methods are overwhelmed by the complicated and large amounts of data. The needs are already high, and they're only going to become higher due to changing data streams and real-time requirements. In most cases, sensors will continuously provide data streams. Data streams are continuous sources of data, usually created at a rapid pace. The ability to receive data in real-time and respond instantly is crucial in completely automated manufacturing settings. The importance of machine-to-machine (M2M) communication is significant in IoT contexts. Smart sensors and gadgets can sense their surroundings and respond instantly by exchanging data and interacting with one another. In these IoT settings, there is a need for real-time communication and constantly flowing data streams, but the feature of taking a picture of the full data set and executing computations with uncertain reaction times is at odds with this. To handle such demands, self-adaptive algorithms are crucial. These algorithms constantly learn and enhance their models. Furthermore, these algorithms need to exhibit real-time behavior and provide excellent performance. This is the case whether they're operating on robust cloud systems, fog and edge systems, or Internet of Things devices (Mocanu, Nguyen, Gibescu, & Kling, 2016).

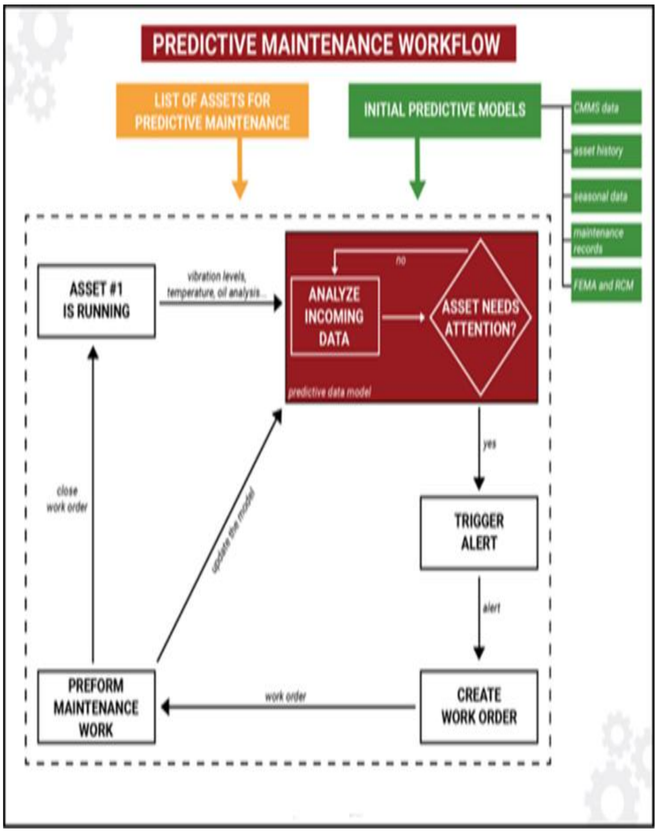

Fig 1 Maintenance Workflow

Within the scope of this study, a narrative review serves as the methodological technique. Instead of selecting the articles to be examined on a quantitative basis, the writers chose to focus on the qualitative aspects of the research. Priority was given to papers that summarized the most recent findings from research conducted about quick

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR984

predictions in IoT using DL. There are a great number of articles that discuss the topics of DL in (I)IoT. To the best of our knowledge, no work in the existing body of literature addresses the subject of Pd.M. about DL and (I)IoT.

Several distinct deep learning techniques that have been discussed for usage in industry and the Internet of Things are categorized in this overview. In addition to that, it discusses real-time processing and data streams about the deep learning methodologies that were stated initially. The techniques that are examined and classed are those that are designed to increase the real-time and stream-processing capabilities of the various methods that are discussed in the papers that are being reviewed. There is a particular emphasis placed on the capability of the methodologies that have been presented to offer forecasts. In the last section of the study, a summary and a view on future developments are presented (Pani, Pattnaik, & Pattanayak, 2024).

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

An introductory discussion on ANN and DL is included at the beginning of this section. Following that, a taxonomy of the many DL approaches that have been described for usage in industry and the Internet of Things will be presented. When it comes to the requirements of Pd.M. in IoT contexts, the categorization will be carried out based on the theoretical methods, application areas, and strengths and weaknesses of their respective systems. The articles that were assessed covered a variety of issues, including real-time and data stream processing, as well as deep learning approaches in Cyber-Physical Systems (CPS), Internet of Things (IoT), and Industry 4.0 (I4.0).

In contrast to Machine Learning (ML), which is a subfield of Artificial Intelligence (AI), Deep Learning (DL) may be seen as a subclass that falls under the umbrella of ML. Many people describe deep learning as a class of efficient artificial neural networks (ANNs) that consists of multiple layers (hidden layers). In addition to supporting additional qualities such as the capacity for unsupervised learning or automated feature extraction, the huge number of layers and neurons enables the abstraction of more complicated problems. Deep Neural Networks (DNN), Deep Belief Networks (DBN), and Recurrent Neural Networks (RNN) are some examples of Neural Network Technologies (Sane, 2020).

The artificial neural network (ANN) is designed to mimic the biological neural network that is found in the brains of mammals. A neural network (ANN) is made up of neurons, which are referred to as nodes in ANNs, and the connections that exist between those nodes. The data that is supplied into the system is used to generate nonlinear output data, which is structured in layers by the nodes. With the help of the connections that exist between the nodes, the output of one node may be transferred to the input of another node. The significance of the signal that is conveyed is determined by the weights that are allocated to each link. In the same way that a threshold function governs the output signal of a neuron (node) in biological neural networks, the same thing happens here. All the weights in an artificial neural network (ANN) need to be initialized to a value, which is often simply a basic approximation. Through the process of training the network, those weights are updated holistically, following a predetermined learning rate, to ensure that the network is both valid and balanced. "Connections developing over time with training" is another term that is often used to describe this phenomenon. ANNs have been around for more than half a century, and throughout that time, several methods have been created.

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 are shown as examples of deep learning models that may be used for Internet of Things (IoT) applications. When it comes to deep learning models, they are divided into three primary categories: generative techniques (AE, RBM, DBN, VAE), discriminative approaches (RNN, LSTM, CNN), and hybrid approaches (GAN, Ladder Net), which are a mix of the two approaches described before. This categorization mostly refers to the learning technique that is being used, with generative methods essentially adhering to the concept of unsupervised learning and discriminative approaches adhering to the idea of supervised learning. The underlying learning technique is a critical aspect in the selection of a deep learning strategy. This is in addition to the defining of the needed number of layers, which is expressed as complexity. The classification of techniques as either generative or discriminative, which was selected by, can be substantially found in a great number of subsequent studies. Numerous DL models are classified according to how well they work in Internet of Things applications. The relevant characteristics that are mentioned include the capability to work with (partially) unlabeled data (feature extraction, feature discovery), the size of the training dataset that is required, the ability to reduce the dimensionality of the data, the capability to deal with noisy data and time series data, and the general performance classification of these characteristics. The integration of recurrent neural networks (RNN) with deep neural networks (DBN) and artificial intelligence (AE) is recommended for the reduction of high-dimensional data and for dealing with unlabeled data. If the system is intended to produce predictions, like how PdM systems are designed, DBN and AEs are often used as an upfront layer that supplies categorized data to a later RNN (Elkateb, Métwalli, Shendy, & Abu-Elanien, 2024).

Recurrent neural networks (RNNs) are suggested for usage in the situation of spatial-temporal data, such as mobility data since they provide strong results when the data is evolving sequentially. RNNs, on the other hand, are not a viable option if the data also contains long-term dependencies. This is because RNNs do not remember prior states and outcomes. The article describes a method that may be used to manage sequential data streams that originate from human mobility and transportation transition

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR984

models that have long-term dependencies (behaviors). In the form of a specific RNN architecture, the solution that has been given is a mix of LSTM on the one hand and RNN on the other. In addition to the capability of managing long-term dependencies, the LMST also incorporates labeling and predictive capabilities into the combination of these two features. There are a great number of additional works that include the combination of recurrent neural networks (RNN) and long short-term memory (LSTM) to deal with data streams or time-series data that have long-term dependencies (such as particular behaviors or the wear and tear of machinery).

Choosing the appropriate artificial neural network (ANN) to generate predictions from data streams and time-series data is the topic of discussion in the study titled "IoT Data Analytics Using Deep Learning." A combination of LSTM and Naive Bayes models is offered as a means of retrieving trends and forecasts, as well as validating those trends and predictions in parallel via the detection of anomalies. In contrast to the Naive Bayes model, which is responsible for anomaly identification based on the outputs of the LSTM, the linear support vector machine (LSTM) is responsible for producing predictions on data streams. This paper also considers the fact that Simple Feedforward Artificial Neural Networks (FNN) such as Single-layer Perceptron (SLP) and Multi-layer Perceptron (MLP) that use standard backpropagation (BP) for training are frequently not a good choice. This is because these neural networks do not perform well in complex situations and on data streams that have long-term dependencies. It is particularly important to keep this in mind when the data streams in question are time series data and the objective of the model is to forecast future occurrences or trends. Dependencies between data streams and time-series data often develop over time. These dependencies are normal for data collected by the Internet of Things and provide valuable insights. Data is assumed to pass linearly through the layers of rudimentary artificial neural networks (ANNs), with the premise that input data is independent of output data. Considering this, there is no way to recall the input and output states that occurred in the past (the outcomes that occurred in the past). It is a problem if the data from the past is connected to the data from the present. When compared to other methods, RNN has the potential to provide superior outcomes in data streams and time-series data. The ability to recall prior states is made possible by the fact that the connections between nodes in a recurrent neural network (RNN) are in the form of sequences or loops. A view state is the sole thing that is generally remembered to prevent gradient outbursts. Because of this, only short-term dependencies are taken into consideration. It is recommended that long short-term memory (LSTM) be used in complicated Internet of Things scenarios to identify long-term dependencies in the data source. LSTM is a kind of RNN that introduces memory units into the network. In addition to being able to retain significant former states, these memory units are also able to forget less significant states (Rippel, Lutjen, & Freitag, 2017).

To forecast the behavior of energy systems in the manner of smart grids, it is important to see that more intelligent systems are required to provide reliable forecasts about the future use of energy. ANN-based prediction methods are a promising approach, according to the paper titled "Deep Learning for estimating building energy consumption." This is because these methods can handle massive and highly non-linear time series data that originates from various heterogeneous data sources (for example, SmartMeter) and contains a great deal of uncertainty (unlabeled data). They benchmarked two distinct methods to the RBN, namely the Conditional Restricted Boltzmann Machine (CRBM) and the Factored Conditional Restricted Boltzmann Machine (FCRBM), using a synthetic benchmark dataset throughout the course of the research article. Because it incorporates a factored conditional history layer, the authors of this experiment have concluded that FCRBN is superior to RNN, Support Vector Machine (SVM), and CBRM in terms of performance. RBMs are a kind of stochastic artificial neural network (ANN) that has two layers: a visible layer and a hidden layer. The visible layer of a recurrent neural network (RBM) is comprised of a node for every conceivable value that is present in the input data, while the hidden layer is responsible for defining the categories of values. As a result of the fact that every visible layer node in an RBM is linked to every hidden layer node, an RBN is effective in feature categorization, feature extraction, and complexity reduction (by determining which features are the most significant). RBMs may be stacked for DL purposes. A conditional history layer, also known as CRBM, is an extension of the RBM that enables the RBN to identify long-term relationships in time-series data. In addition, the output of a single stacked CRMB layer is factored. This is done to decrease the total number of compositions that are conceivable.

The very effective predicting capabilities of DL are also highlighted in another work that is published in the realm of energy management. This article describes the use of AE and LSTM to estimate the amount of electricity that solar systems will generate. A comparison is made between the accuracy achieved by a combination of AE and LMST (Auto-LSTM) and that of other neural networks, namely MLP, as well as a physical model. 21 actual solar power plants were used to collect the data for the benchmark, and the benchmark itself was obtained via an experimental setup that was detailed in. To establish benchmarks, the following metrics are utilized: average root-mean-square deviation (RMSD), average mean absolute error (MAE), average absolute deviation (Abs. Dev.), average BIAS, and average correlation. The measured findings demonstrate that all ANN- and DL-based models provide outcomes that are much superior to those produced by the physical model. When it comes to artificial neural networks (ANN) and deep learning (DL) models, Auto-LSTM is the most suitable option for this circumstance and data set. When it comes to producing predictions, one of the most important factors that is discussed is the power to extract characteristics from unlabeled data.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR984

The work titled "An enhancement deep feature fusion method for rotating machinery fault diagnosis" highlights the capabilities of AEs in the areas of feature extraction and feature learning. The deep feature fusion approach is described in the study to further increase the capability of feature learning while simultaneously reducing the effect of background sounds. This is accomplished by stacking Deep AE (noise reduction) and Contractive AE (improved feature recognition). A critical component in the process of forecasting.

III. FAST PREDICTIONS USING MACHINE LEARNING

There are several Internet of Things applications that need real-time processing. In the case of a PdM system, for instance, a high latency might result in inadvertent reactive maintenance due to a lack of adequate lead time to schedule the maintenance operations. The application case has a significant impact on the desired speed at which real-time processing must be performed. In the context of micro-manufacturing systems, where enormous quantities of micro components are produced at a rapid pace, the phrase "real-time" refers to the amount of time that is measured in microseconds. demonstrates that the rejection rate of produced micro parts may be reduced by improving processing speed when defect detection and PdM systems are included. In certain contexts, the concept of real-time may refer to the passage of seconds, minutes, or even hours. In the case of PdM applications for offshore wind turbines, for instance, the frequency with which the data is accessible is mostly minutes and hours (Xie, Wu, Liu, & Li, 2022).

The creation of a real-time crowd prediction system for public transportation is described in the article titled "Metro Density Prediction with Recurrent Neural Network on Streaming CDR Data." This system makes use of a weight-sharing recurrent neural network in conjunction with parallel streaming analytical programming. It is necessary to do real-time analysis to have a quick reaction time to emergent events, such as admission records at metro stations mixed with data from telecommunication, for example. nonetheless, the use of a robust neural network model that has a high learning capacity provides a broad variety of fresh insights; nonetheless, this is in contradiction to the need for a quick reaction time. How to achieve this objective is broken down into three stages, which are as follows: To enhance its capacity to operate on data streams, a) the adoption of a Recurrent Neural Network (RNN) model; b) the implementation of techniques for the parallelization of RNNs; and c) the use of parallel streaming analytical algorithms as part of a cloud-based stream processing platform. An independent recurrent neural network (RNN) is used to represent each metro station in the project that is detailed in. To dynamically share weights from stations that are in comparable "situations" (for example, a downtown station during rush hour), shared layers are created. This allows for the sharing of weights across many models. The ability to co-train in parallel is another benefit of weight-sharing (Gensler, 2021).

The use of recurrent neural networks (RNNs) and the many variants of these networks for efficiently analyzing data is also advocated. RNNs have the potential to give superior performance compared to other models, particularly when applied to more common sensor data such as serial data, time-series data, and data streams. The vast majority of PdM applications are dominated by this kind of sensor data.

It is vital to have the skills of real-time processing and real-time learning to be able to construct and permanently adjust models on huge amounts of data that include the behavior of individuals as well as their geographical and temporal qualities as well as their transportation capacity. A deep LSTM learning architecture that can do several tasks is described in this study. The fundamental idea behind this technique is not to make use of a combined feature vector but rather to employ several LSTM tasks that are separated by their respective domains (for example, a separate job for mobility and transportation mode prediction). Parallel learning is carried out using this architecture, and the results are pooled by the insights that are desired.

It is essential for assistance systems in automobiles, such as traffic sign recognition, to provide precise results while maintaining a low latency. Within the scope of this topic, the paper explains how to implement DNN. The model of the system is not only supplied with data that is fully unlabeled (raw photos), but it is also continually updated via online learning. When it comes to picture identification, a CNN with nine layers is used. To enhance the overall performance of the system, max-pooling layers are merged with convolutional layers in a manner that is alternating. Based on the 2D input pixel maps, the convolutional layers conduct the convolution operation. In the process of translating the output of a previous convolutional layer into the input of a succeeding convolutional layer, the max-pooling layer functions as a pre-processor between two convolutional layers. It does this by removing overlapping areas in the pixel mappings. This removes the need for duplicate processing in the convolutional layers, which are notoriously difficult and time-consuming. Multi-Column DNN (MCDNN) is the name given to the method that is explained in the traditional method.

In the study, a solution that is focused on real-time approaches to the detection and identification of traffic signs is described. It is essential to have parallel processing since it is necessary to identify a variety of traffic signs at the same time. This is the major focus of attention. In this method, CNN is also used for image processing, and it is combined with AdaBoost to enhance performance and parallel GPU processing (Song, Kanasugi, & Shibasaki, 2016).

Data that has long-term dependencies is a strong candidate for LSTM models because of the memory cells that they include. It will likely be feasible to process each entity and each group with its neural network if the data structure permits the separation of single entities with their behavior as well as the development of groups of entities.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR984

This enables the separate neural networks to function in parallel, which opens new processing possibilities. An aggregation layer is often responsible for receiving the outputs of each single and parallel processed neural network. This layer then combines all of the outputs to provide an overall result. In the article titled "A Hierarchical Deep Temporal Model for Group Activity Recognition," the author explains how to identify several circumstances that may arise during a volleyball match. With the use of long-term dependencies, one LSTM model for each player can make predictions about the player's behavior by remembering his prior actions throughout the match. Subsequently, every single circumstance that occurs throughout the match is modeled as a group of players. The LSTMs are arranged hierarchically, with the LSTM models of all of the participants participating being subordinated to those of a particular scenario. CNN is used to gather information about the scenes and the behavior of the players based on the photographs.

Changing the arrangement of layers and connections is something that is mentioned in the study titled "Simulation of Maintenance Activities for Micro-Manufacturing Systems." This is due of the expectations that have been placed on real-time processing procedures. Computer systems that are fully linked, in which every node of one layer is connected to every node of the layer below it, can solve difficult problems, but they also need a significant amount of computational power. The approach of dropping out any connections that do not significantly affect the outcome is a method that may be used to lessen the complexity of a deep learning network and, therefore, its computational demand, without compromising accuracy in a meaningful way. In addition to dropout, you could also discuss max-pooling layers, batch normalization, and transfer learning as other options for performance enhancement.

The argument presented in the study titled "An Analysis of Deep Neural Network Models for Practical Applications" is that many of the deep learning models that have been detailed in the literature are simply not appropriate for application in actual situations. For instance, this is because their processing time is lengthy or because they use an excessive amount of electricity. In his work, he makes the argument that performance concerns should get a greater amount of attention since they are important variables in practical applications of deep learning. In this study, fourteen distinct deep learning projects, such as Alex Net and Google Net, are compared with one another in terms of their accuracy, memory footprint, parameters, operations count, inference time, and power consumption. The research presented in this article demonstrates that a very little improvement in accuracy may result in a significant increase in both the amount of processing power and the amount of time required for calculation. It is strongly suggested that maximum energy usage be established for each DL project and that the accuracy be adjusted by that limit.

IV. CONCLUSION

A narrative assessment of chosen literature that applies deep learning methods to the area of industrial internet of things (IoT) to create rapid forecasts of maintenance concerns was presented in this work. According to the papers, the use of DL in Internet of Things and PdM is an important subject in the business world. Currently, a wide variety of applications are being used in practice, and these applications are continuously being created and enhanced.

Combining several deep learning models to combine their respective benefits and capabilities in a single application is a common practice that has been described. Additionally, the need for real-time processing of complicated data and data streams has been established in several application situations.

The applications for predictions that fall within this category include PdM in particular. Concepts of parallel deep learning networks that make use of a final aggregate layer or intermediate layers to simplify the system are widely used to enhance the real-time capacity. Even if there are a lot of activities that can be seen in the field of real-time processing of deep learning models, there are also voices that are critical of the absolute concentration on accuracy. These voices are advocating for a larger focus on performance and lighter applications that are suited for practical usage. Most papers agree that a significant amount of research is still required in this field.

SUMMARY OF REVIEW PAPERS

Table 1 Summary of Review Papers

Reference	ML Methods	Characteristics	Typical Applications
(Mohammadi, Al-Fuqaha, Sorour, & Guizani, 2018)	AE, CNN, DBN, GAN, LSTM, RBM, RNN, VAN, Ladder Net.	To do picture identification, feature extraction, and dimensionality reduction of Internet of Things data using AE and DBN CNN is required, but a huge training set is required. a ladder, a GAN, and a VAE It is a classification layer for recurrent neural networks (RNN) that enables unsupervised learning and is appropriate for unsupervised learning. The LSTM algorithm offers a high level of performance for data that has long-term dependencies. RBM to solve issues including feature extraction,	The identification and forecasting of faults Internet of Things surroundings Processing in real time and in streams using a variety of recurrent neural networks.

		dimensionality reduction, and classification a specialized RNN designed for time-series data.	
(Song, Kanasugi, & Shibasaki, 2016)	LSTM, RNN	Data streams that include time series and Internet of Things data; LMST for data that has long-term dependencies In conjunction with RNN, LSTM provides labeling and prediction capabilities. RNN is effective in situations where sequential data and data streams are present.	IoT, Transport, Mobility
(Xie, Wu, Liu, & Li, 2022)	LMST, RNN	IoT data streams and time series data streams are both suited for LMST and RNN.	Predictions made possible by long-term dependencies in data RNN for short-term Internet of Things applications such as condition monitoring.
(Mocanu, Nguyen, Gibescu, & Kling, 2016)	RBM, CRBM, FCRBM.	Feature extraction, dimensionality reduction, and classification will be accomplished via RBM. To provide long-term forecasts, CRBM is an extension of RBM that incorporates a conditional history layer. The number of potential compositions of each output layer in a stacked (C)BRM is reduced thanks to FCRBM, which results in a performance improvement.	Predictive Internet of Things applications such as estimates of electricity generation.
(Gensler, 2021)	DBN, Auto LSTM	DBN is effective when it comes to making predictions on time-series data. Auto-LSTM is a mix of AE and LSTM that is used to make predictions on time series database data.	Predictive Internet of Things applications such as estimates of electricity generation.

The publications that were evaluated are summarized in Table 1, which also includes a discussion of the DL-Methods. An overview of the features (or strengths and shortcomings) of the DL methods described in the related publication is provided for each study. Additionally, the suggested application areas (such as predictions) are also included in this summary. Table 1 does not include any conclusions or assertions about the validity of the data in a quantitative manner. It is only by qualitative means that the various DL models are categorized into their respective categories. This is only when specific measurable values are specified across all of the publications that were assessed. The only assertions that are provided by the other papers are qualitative ones. There is an issue that remains unanswered about how to quantify and evaluate the validity and quality of the findings obtained from various DL approaches. To this day, there have been very few methods created for measuring, assessing, and benchmarking each other. In addition, such methods are often not verifiable within the context of universal validity. For instance, when it comes to classifications, the use of accuracy estimation procedures, such as the "holdout method" or "n-fold cross-validation," may be utilized to accomplish the evaluation of performance, predictive ability, and model correctness. Considering this, the strategies that have been presented partition a training set into data regions for learning and validation using a variety of methods. There is currently no notion of measuring, assessing, or benchmarking that has been specified for most models. In general, the assessment is carried out here based on the views of specialists. It is pointed out in the study titled "Data Stream Classification and big data analytics" that there is a necessity for new techniques of measuring and benchmarking. To evaluate deep learning models, it is necessary to have measuring techniques that are effective in producing representative benchmarks.

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, February 28). Deep Learning for solar power forecasting — An approach using AutoEncoder and LSTM Neural Networks | IEEE Conference Publication | IEEE Xplore. Retrieved February 28, 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

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAR984

[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 | IEEE Xplore. Retrieved February 28, 2024, from ieeexplore.ieee.org website: https://ieeexplore.ieee.org/document/8342870/metrics#metrics

[8]. Song, X., Kanasugi, H., & Shibasaki, R. (2016). *DeepTransport: 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). *IoT Data Analytics Using Deep Learning*. Retrieved from https://arxiv.org/pdf/1708.03854