Expert Systems With Applications 173 (2021) 114598

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Predictive maintenance system for production lines in manufacturing: A machine learning approach using IoT data in real-time

Serkan Ayvaz [a,*], Koray Alpay [b]

[a] Department of Artificial Intelligence Engineering, Bahcesehir University, Besiktas, Istanbul, Turkey
[b] Department of Gebze Plant Digital Team, Gebze Development Center, Procter & Gamble, Kocaeli, Turkey

ARTICLE INFO

Keywords:
Predictive maintenance
Internet of things
Manufacturing systems
Artificial intelligence
Machine learning
Big data

ABSTRACT

In this study, a data driven predictive maintenance system was developed for production lines in manufacturing. By utilizing the data generated from IoT sensors in real-time, the system aims to detect signals for potential failures before they occur by using machine learning methods. Consequently, it helps address the issues by notifying operators early such that preventive actions can be taken prior to a production stop. In current study, the effectiveness of the system was also assessed using real-world manufacturing system IoT data. The evaluation results indicated that the predictive maintenance system was successful in identifying the indicators of potential failures and it can help prevent some production stops from happening. The findings of comparative evaluations of machine learning algorithms indicated that models of Random Forest, a bagging ensemble algorithm, and XGBoost, a boosting method, appeared to outperform the individual algorithms in the assessment. The best performing machine learning models in this study have been integrated into the production system in the factory.

1. Introduction

With the digital transformation experienced today, applications of Big Data and Artificial Intelligence (AI) shape our daily lives and increase the efficiency of working processes in all areas. Unprecedented technological and scientific developments are particularly taking place in the fields of AI and Internet of Things (IoT), which are at the heart of the fourth industrial revolution (Lee, Davari, Singh, & Pandhare, 2018). In this era, AI and big data analytics play a vital role in innovation independently of the industry by providing solutions to complex problems. Due to global challenges and intense competitiveness in the markets, the firms constantly face a pressing need for increasing efficiency, reducing costs and delivering better solutions nowadays (Furman & Seamans, 2019). Institutions and countries that successfully carry out the processes of understanding, analyzing big data, producing fast results, developing AI driven new and useful tools, and adapting data centered decision-making mechanisms are one step ahead in the competitive world (Bolton, Machová, Kovacova, & Valaskova, 2018; Wamba-Taguimdje, Wamba, Kamdjoug, & Wanko, 2020).

1.1. Data collection and connectivity through IoT

Technological advancements have transformed how individuals access to information in recent decades. Now, data are being collected from variety of resources. Mobile phones, wearable technologies, and other smart devices have become essential part of daily life. While data is being collected more rapidly, it has a more complex structure now. The key challenge is analyzing such a big data collected from various environments and resources in a timely manner, and utilizing it in making better decisions in important issues such as management of real-time operations, processing and evaluation of data produced by devices, and instant detection of risks and faults (Wang, Wan, Zhang, Li, & Zhang, 2016).

Thanks to rapid developments in data protocols, smart devices and technologies, the IoT has been transforming the manufacturing processes by integrating systems and connecting devices, systems and people (Ansari, Erol, & Sihn, 2018). At the center of industry 4.0 revolution, the IoT facilitates greater productivity in manufacturing by enabling connectivity and data exchange between production systems (Lu, 2017). Modern manufacturing processes are typically composed of very complex autonomous robotic machinery, which are embedded with sophisticated software systems and surrounded by the IoT devices. With

* Corresponding author at: Yildiz mh. Ciragan cd. No: 4-6, Besiktas, 34349 Istanbul, Turkey.
E-mail addresses: serkan.ayvaz@eng.bau.edu.tr (S. Ayvaz), alpay.k@pg.com (K. Alpay).

https://doi.org/10.1016/j.eswa.2021.114598
Received 10 September 2020; Received in revised form 14 November 2020; Accepted 8 January 2021
Available online 16 January 2021
0957-4174/

integration of robotic and autonomous equipment, the efficiency of manufacturing systems has increased tremendously around the world in recent years (Stock & Seliger, 2016).

1.2. Production stops and preventive maintenance

The production environment rely on equipment to work properly. A fault in a component or sub-system may cause a stop in entire production line. The production stops are associated with huge costs in many folds including but not limited to the loss of production and time in downtime, losses of efforts in identification of the cause of the failure and repair, wastes of products produced right after bringing back the system until normal operations due to low quality, costs of repairs and deterioration of equipment (Froger, Gendreau, Mendoza, Pinson, & Rousseau, 2016).

The stops may occur as a consequence of various types of issues such as malfunctions of equipment, operator errors or environmental factors. Although it may not be possible to eliminate all system failures, preventable faults make of a large portion of the production stops (He, Gu, Chen, & Han, 2017). If predicted early, the equipment or sub-system leading to potential faults can be addressed without a production stop, which would result in significant saving in associated costs (Yu, Dillon, Mostafa, Rahayu, & Liu, 2019).

Predictive maintenance (PdM) has become common objective in the industry to reduce maintenance costs and ensure sustainable operational management (Stock & Seliger, 2016). The essence of the PdM is to predict the next error in a manner that preventive maintenance can be performed before the failure takes place. PdM also has the potential to promote sustainable practices in production by maximizing the useful lives of production (Lee, Kao, & Yang, 2014).

For PdM, data flexibility is seen as a critical issue that can compromise algorithm performance for modeling based on data (Wang, Zhang, Duan, & Gao, 2017). Processing raw data from sensors presents another challenge since sensor data are not labeled. In addition, data sparseness may affect the algorithm performance in existing approaches when addressing the problem of data tagging. Moreover, it is often very difficult to clearly identify the anomalies and patterns in high dimensional IoT data (Erfani, Rajasegarar, Karunasekera, & Leckie, 2016). When considering big data environments, it is even more challenging due to noise in device generated data (Yu et al., 2019).

1.3. AI applications in manufacturing

Nevertheless, data driven AI applications utilizing data collected from IoT devices can help in predictive maintenance (Kanawaday & Sane, 2017; Yu et al., 2019; Wang et al., 2017). The aim of this study is to develop a predictive maintenance system that produces realistic predictions of potential failures for production lines in manufacturing before occurring using machine learning methods. In order to obtain the most suitable model to address this problem, multiple algorithms were explored in details and compared using a real-world dataset.

The paper is organized as follows: Section 2 reviews existing solutions and the related work in the area of predictive maintenance. In Section 3, the methods and materials used in the study are explained in detail. Section 4 presents a real-world application of the proposed system in a manufacturing plant as a use case. Section 5 illustrates the results of the evaluations assessing the efficiency of the proposed system. Section 6 provides an interpretation of the results obtained and discusses the significance of the study. Finally, it is followed by conclusion.

2. Related work

With continuing development in big data, AI and IoT technologies that are driving forces of industry 4.0, the field of predictive maintenance has attracted many researchers from research community recently (Rieger, Regier, Stengel, & Clarke, 2019; Li, Wang, & Wang, 2017; Zhang, Wang, Yan, & Gao, 2018; Traini, Bruno, D'Antonio, & Lombardi, 2019; Yu et al., 2019; Erfani et al., 2016; Fernandes, Canito, Corchado, & Marreiros, 2019). Some studies have treated PdM from diverse perspectives such as supervised classification (Traini et al., 2019; Edward, Dariusz, Żabiński, Prucnal, & Se, 2020, anomaly detection Rabatel, Bringay, & Poncelet, 2011; Yu et al., 2019; Rivera, Scholz, Bühl, Krauss, & Schilling, 2019), regression in high dimensional data (Zhang, Wang, Yan, & Gao, 2018; Chen et al., 2020; Zhang et al., 2018), reinforcement learning for system modeling (Huang, Chang, & Arinez, 2020) and unsupervised learning problem (Erfani et al., 2016; Zhao et al., 2017).

2.1. Statistical approaches to predictive maintenance

Based on current developments in production and technology, estimating remaining useful life of machinery has begun to play an important role for ensuring machine condition monitoring, productivity, reliability and safety. Different theoretical and practical applications have been proposed. Frequently reported are combinations of different deep learning models that combine multiple aspects in one application. In addition, the need for real-time processing of complex data and data streams has been demonstrated in some application scenarios. Similarly, Autoregressive Integrated Moving Average (ARIMA) models have also been used for error detection and predictive maintenance forecasts in time series (Ho, Xie, & Goh, 2002; Ramos, Oliveira, & Silva, 2014; Kanawaday & Sane, 2017; Fernandes et al., 2019; Francis & Mohan, 2019). However, when the accuracy of prediction performance is compared, it is known that ARIMA methods are not as successful and flexible as the machine learning counterparts. Hence, machine learning methods were considered in our approach only.

2.2. Data-driven AI applications to failure predictions

Zhang et al. investigated developing models using temporally dependent sensor data for Nasa's aircraft engine performance decline monitoring and engine useful life expectancy (Zhang et al., 2018). Due to their wide use in most aircraft engines, 6 built-in sensors were selected and focused on a Long Short Term Memory (LSTM) based model to characterize system failure behavior and estimate the useful life remaining. In addition, the performance of LSTM's ability to capture temporal dependence was compared with various machine learning techniques. Because the recurrent neural network (RNN) algorithms do not store long-term memory, the LSTM cell was chosen for this problem. In the experiments, 200 synthetic cycles were created to test LSTM performance. It has been shown that the trained model can effectively predict the actual behavior of the test set and the accuracy of the prediction increases as the length of the prediction decreases. Compared to various machine learning prediction models, LSTM performed better in useful life estimation in both datasets than alternative machine learning methods including Support Vector Regression (SVR), Multilayer Sensor (MLP), and Deep Evolution Neural Network (DCNN) (Zhang et al., 2018).

In a similar study, Lee et al. worked on data-driven artificial intelligence based modeling approach to predict spindle motor, cutting machine wear and malfunctions (Lee et al., 2019). In their experiments, SVR and artificial neural network models have yielded successful results in predicting the condition and remaining life of the systems. In another related work, Traini and others proposed a modeling framework to estimate maintenance on grinder machines (Traini et al., 2019). They evaluated several algorithms for regression and classification purposes.

Rivera et al. examined the errors in production systems from the point of view of anomaly detection (Rivera et al., 2019). Accordingly, they provided a statistical method based on kernel density estimation and the evolution of distribution over time, which could lead to the possibility of predicting maintenance. However, when analyzed from an expert point of view, it was noticed that there were not many outstanding findings in the data. On the other hand, the problem of reinterpretability of the results has emerged. Because it was not clear why the algorithm marked the particular cycle as abnormal, they had difficulty deciding whether it was correct to throw the cycle as an anomaly for further analysis.

In Rivera et al. (2019), the authors pointed out the difficulties of using raw sensor data and stated that without using expert knowledge, the reference data may deviate too much and may cause the proposed system to make false predictions. The most important finding that they obtained was the importance of data quality and the use of expert knowledge.

Li et al. also proposed a data mining framework for maintenance and error prediction in production machines (Li et al., 2017). However, they stated that there were still many challenges in the application of the framework that they proposed for fault diagnosis in machine centers. The main benefits of the Artificial Neural Networks (ANN) algorithm that they used in the application were fault tolerance, generalization and adaptability, but besides, the limitation was that it is not possible to determine the effect of various factors and the disadvantages of not having an explanation function.

2.3. Deep learning applications in predictive maintenance

From deep learning perspective, Song et al. and Xie et al. have also confirmed that RNN and LSTM algorithms perform well for sequential data, time series with long-term dependencies, and IoT flow data (Song, Kanasugi, & Shibasaki, 2016; Xie, Wu, Liu, & Li, 2017). The authors in Xie et al. (2017), suggested that a combination of LSTM and Naïve Bayes model may capture trends and make forecasts successfully. Naïve Bayes served as an anomaly detector in the output of the LSTM model.

Contrarily, Mocanu and others stated that the Factored Conditional Restricted Boltzmann Machine (FCRBM) method performs slightly better than other algorithms in comparing deep learning methods in their applications involving the prediction of building energy consumption in smart cities (Mocanu, Nguyen, Gibescu, & Kling, 2016). In addition, when Gensler and others examined deep learning methods in an IoT application that includes solar panels prediction, in addition to LSTM, Deep Belief Network (DBN) method also showed good results in time series data (Gensler, Henze, Sick, & Raabe, 2016).

Rieger et al. reviewed and categorized the deep learning methods used in predictive maintenance in IoT devices (Rieger et al., 2019). Temporal delay in failure prediction in industrial production processes is a problem. Therefore, it is treated as a real-time process. In such estimates, RNNs have been proposed due to sequential data development in spatial–temporal data types, because of their success in these types. However, for these data types with long-term dependencies, RNNs were not considered a good choice as they did not learn the previous situation and results. For such problems, the combination of RNN with LSTM is preferred because LSTM adds labeling and prediction functionality.

In a research study, a new approach called Cox proportional hazard deep learning (CoxPHDL) was proposed to address data flexibility and data censorship, which are commonly mentioned in the analysis of operational maintenance data (Chen et al., 2020). The main idea was to offer an integrated solution using deep learning and reliability analysis. To begin with, an automatic encoder was adopted to convert nominal data into a solid representation. Secondly, a Cox proportional hazard model (Cox PHM) was investigated to estimate Time-Between-Failure from censored data. A long-term memory (LSTM) network had been created to train the Time-to-Failure prediction model based on preprocessed maintenance data. The experimental studies using a large real-world maintenance dataset demonstrated the advantages of their proposed approach, in which algorithm performance was based on the LSTM network (Chen et al., 2020).

Mohammadi et al. also compiled a large number of deep learning algorithms used in the processing of flowing IoT data in his review article. It is mentioned that Restricted Boltzmann Machine (RBM) method is useful in feature extraction and dimensional reduction in case of multidimensionality in IoT data. It has been stated that RNN algorithms are successful for time series data and LSTM algorithm, which is a version of RNN for data with long-term dependencies, provides good performance (Mohammadi, Al-Fuqaha, Sorour, & Guizani, 2018). Different from these aforementioned RNN based approaches, we deal with a problem that the exact time to a failure is not known at the time of prediction in production setting. Thus, the RNN algorithms are not applicable to our case because these approaches necessitate availability of actual data at previous time steps due to sequence-to-sequence analysis of data.

While many applications were developed for real-time processing of deep learning based predictive maintenance models, there are also critical voices that criticize the absolute focus on predictive success and recommend more focus on practical use and robust applications (Li et al., 2017).

Numerous deep learning approaches described in the literature are not suitable for practical use. This is due to long processing times, computational complexity and excessive power consumption. Canziani and others propose to pay more attention to performance problems in their articles because performance problems are key in practical deep learning practices (Canziani, Paszke, & Culurciello, 2016). Due to performance implications of deep learning methods in real-time practical applications, we have decided to use data-driven machine learning algorithms in our predictive maintenance system.

2.4. Predictive maintenance in sustainability of cyber-manufacturing

Predictive maintenance has also attracted attention from the perspective of sustainability in Cyber-Manufacturing Systems (Jasiulewicz-Kaczmarek, Legutko, & Kluk, 2020; Song & Moon, 2017; Jena, Mishra, & Moharana, 2020; Machado, Winroth, & Ribeiro da Silva, 2020). Recent technological advancements and digitalization have begun to transform manufacturing towards Industry 4.0 (Jeschke, Brecher, Meisen, Özdemir, & Eschert, 2017) by making systems and processes intelligent connected through IoT to such an extent that physical manufacturing systems can be represented by a digital twin (Lu et al., 2020) in cloud, which enables monitoring and intelligent decision making in real-time (He & Bai, 2020).

Applying data-driven predictive maintenance as a function of smart manufacturing offers sustainability benefits (Song & Moon, 2017) in many ways, including increasing reliability and lifetime of equipment, reducing downtime and energy consumption due to stopping, and lowering cost of materials and inventory due to production wastes (Jasiulewicz-Kaczmarek et al., 2020). Consequently, overall productivity and quality of products can be increased (Song & Moon, 2018).

3. Proposed predictive maintenance system

Making machines, devices and systems connected is the most fundamental step for an IoT application implementation. An efficient and scalable data processing infrastructure is required to use the data streaming continuously and instantly from the sensors placed in the factory in decision making processes. The connectivity of devices in existing production environments can be established by adding sensors to the machines, controllers and installing gateway devices for communication.

The IoT system must have a framework that can be integrated with external systems. With an integrated data streaming prediction model, a PdM system may be able to detect the failures, create alarms by applying certain rules, execute commands on the production systems, and send warning messages to the authorized officials in real-time.

In this study, the infrastructure was first improved by making machines connected through sensors and gateways, establishing the Internet of Things platform, making the data streaming from the production lines constantly accessible and integrating system components prior to developing the predictive maintenance system.

3.1. Overview of proposed system architecture

The system comprises an IoT platform to enable communication between machines, collect data from devices, monitor live data, and manage historical data. The IoT platform monitors the status and sensor data on the machines and periodically record them in the historical database and create the basis data for the failure prediction model.

Since data are provided from a large number of IoT devices at the factory, no data loss must occur, the data collected from all sensors must be processed instantly and evaluated in decision processes. The system contains a database for connected systems, which allows real-time querying of streaming data that is used for machine learning (ML) models. After collecting the data from the sensors and making the necessary data conversions automatically during the data preprocessing process, a fault detection analysis is performed by ML algorithms.

Fig. 1 shows an architectural overview of proposed predictive maintenance system components. The data from the sensors are collected on the private cloud system in a database that allows recording data in distributed systems by gathering them through the Internet of Things (IoT) and converting them into a single data type using the Message Queuing Telemetry Transport (MQTT) protocol, which is a standard messaging protocol designed by OASIS for IoT. MQTT is chosen for connecting the IoT sensors in our case because it is a lightweight, easy to implement and open messaging protocol (Stanford-Clark & Hunkeler, 1999; Locke, 2010).

Using the data collected in real-time from sensors, the system aims to automatically detect the signals for potential errors using machine learning algorithms independently from the human interventions and to guide operational preventive actions. The machine learning models in the system calculate the optimum values according to the changes in the input parameters resulting from instantaneous measurements automatically. Algorithms were trained to optimize the weights of the input parameters, and capture the data patterns in on the stream data successfully. The best performing ML models then were integrated into the production system in the factory.

Application interfaces were developed to dynamically present the relevant content and outputs of the prediction model to the end users as warnings, visual notifications depending on the user roles and authorizations. The end-user interfaces for monitoring and error reporting operations are web applications running on web browsers. The web interfaces were customarily built and integrated into the IoT platform.

In our system, all the computational tasks are executed on-premise servers due to data security and privacy concerns. We also followed the best practices and tuning in order to make efficient use of the architectural setup.

4. Implementing proposed system in a consumer goods manufacturing plant

To assess the effectiveness of the proposed system, we have implemented the system in a real-world manufacturing plant in Turkey as a use case in an additive layered manufacturing process. The plant produces personal care goods such as baby care, feminine hygiene and home care products for a world leader consumer goods production company. The following subsections present the application of the proposed system and the underlying machine learning modeling for describing our general approach in a concrete example.

4.1. Data preprocessing

4.1.1. Data collection

The dataset was obtained from assembly lines producing baby diapers in a real-world production plant. The variables reflect device generated data values that change over time. The dataset consisted data readings from various types of built-in IoT sensors that are primarily detecting properties such as motion, speed, weight, temperature, electrical current, vacuum and air pressure on different equipment in the manufacturing lines. The data points were collected from IoT sensors monitoring the production system in every 3 to 6 s periods. In total, the dataset contained 101 features and 8,389,515 rows.

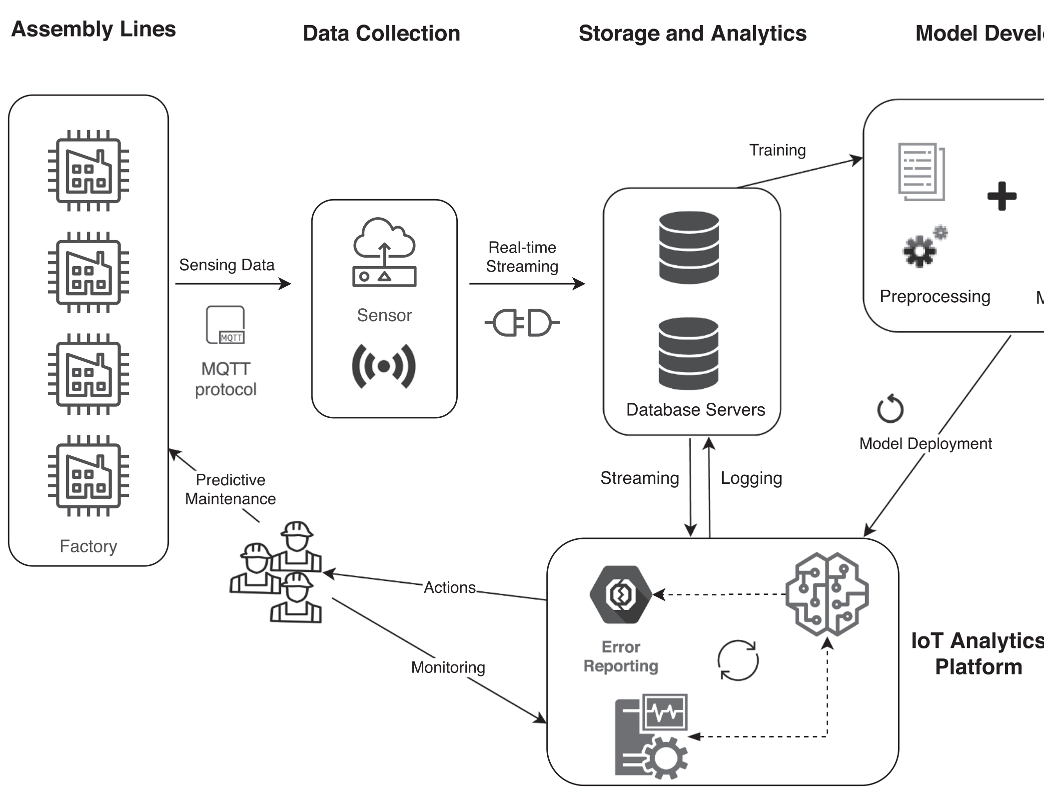

Fig. 1. Architectural overview of the predictive maintenance system.

Most of the columns in the dataset were filled. There were only a few columns with a small ratio of missing values, which consisted approximately one percent of data. The missing values might have occurred due to errors in the process of logging the sensor reading in database. Since the proportion of the missing values was very low, those were automatically filled with median values of the respective columns. All columns except timestamps were numerical. This provided great convenience in the analysis methods.

However, the ranges of the sensor readings were significantly different, in some cases an order of magnitude difference. Normalizing features that are not in the same scale plays an important role in eliminating potential optimization problems during data modeling. The normalization process makes modeling less sensitive to the scale of features and improves overall quality of the analysis. Thus, the data standardization was applied to the features during pre-processing phase of the study.

4.1.2. Handling imbalanced data

In general, machine learning algorithms can capture patterns in data better from a balanced data distribution. When class distribution is imbalanced, the ML classifiers are prone to have bias towards large class. For handling imbalanced datasets, some data sampling strategies have been developed such as applying random under-sampling and over-sampling, syntatic over-sampling (Chawla, Bowyer, Hall, & Kegelmeyer, 2002), bagging and boosting.

Due to the nature of data distribution in predictive maintenance, the data distribution is almost always going to be highly imbalanced. Because the frequency of the failure is often expected to be an order of magnitude less than of the normal (No-failure) operation cases. Moreover, the distribution of classes may be imbalanced among the failure types as it was the case in our situation. Some of the failure types were very rare with a few examples only while other failure types occur more commonly. To build a meaningful predictive model, we omitted rare failure types in the dataset and focused on two of failure types only, namely, Weight low 1 and Weight low 2 errors that originate from adjacent zones in the assembly line and make up the majority of the cases. These errors are called "Weight low" due to the fact that failures originally arise from insufficient raw material feeding into injection system.

In our case, we have applied bagging and boosting methods as balancing strategies.

4.1.3. Correlation analysis

One of the most important outputs of the study is data discovery. The descriptive correlation analysis demonstrates inherent associations among different features. In this research, feature discovery and correlation analyzes have been explored. We examined the relationships between the data points that were collected from the production lines and the remaining useful lifetime before the two Weight low failure types. The features that were descriptively important in preventive maintenance have been investigated for decision making processes and taken into account in modeling task.

Fig. 2 illustrates the results of correlation levels between all features in the analysis. It is clear from the figure that some features have high degree of correlation, e.g., X5 through X12 and X13 through X17, and might lead to Multicollinearity. However, this is not surprising because

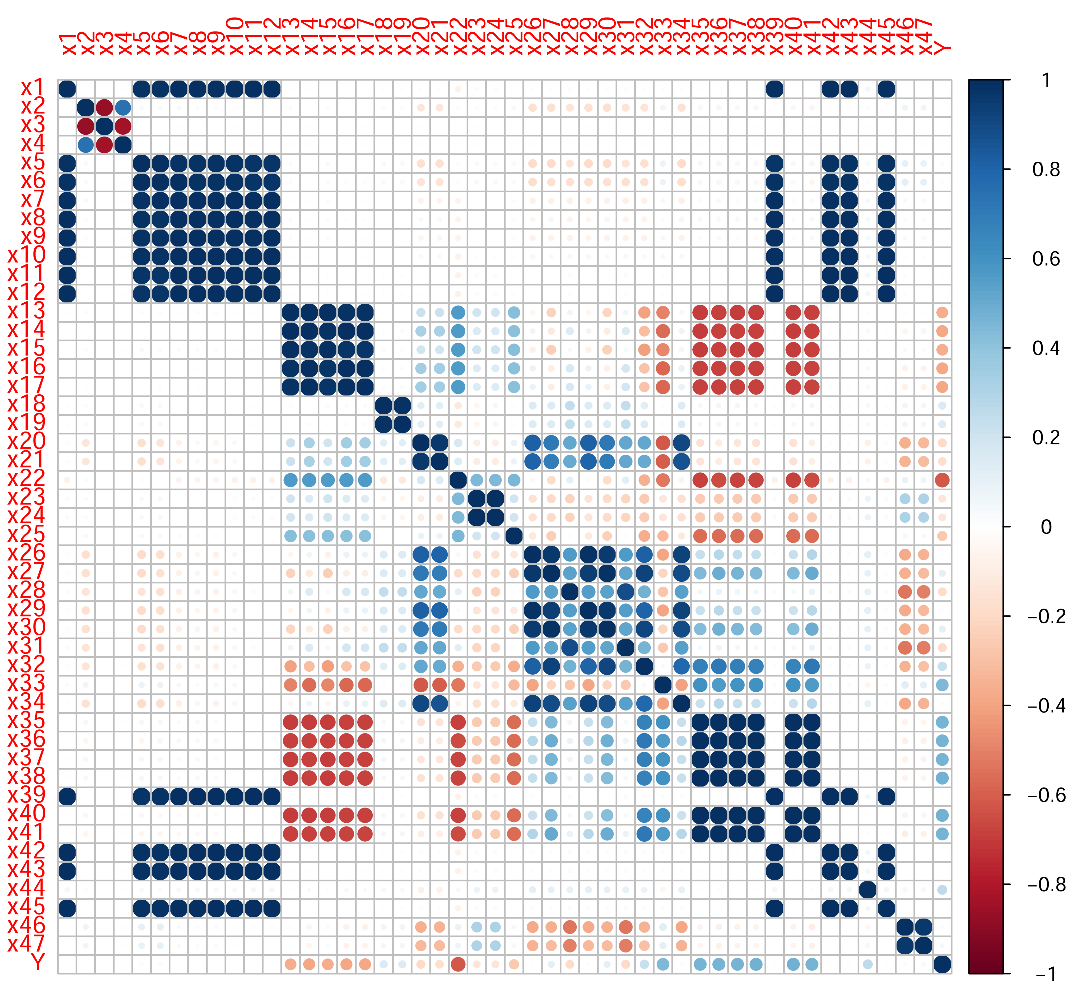

Fig. 2. Correlation of variables.

the highly correlated variables were derived from different sensors on the same components and represented the corresponding subsystem of production unit. Since very highly correlated features could be represented by a smaller form, the correlation results were considered for dimensionality reduction in the feature selection phase.

4.1.4. Feature selection

In machine learning, feature selection is an essential task for modeling high dimensional big data successfully. High dimensionality problem, also known as curse of dimensionality (Bellman, 1966; Powell, 2007), can muddle the ML models since it often causes numerous challenging issues such as overfitting, multicollinearity, exponential computational complexity, data sparseness, and reduced interpretability (Altman & Krzywinski, 2018). By carefully applying dimensionality reduction strategies, feature selection process can help decrease potential ML model prediction errors in high dimensional data.

In the study, the dataset contained 94 sensor generated variables about the production line and 7 data points associated with the failure type, a total of 101 features. The variables related to the two failure types included data points such as timestamp, error type, uptime, downtime, team, work shift at the time of stops. Since each sensor reading was accompanied with a timestamp variable showing the last time the sensor recorded a value, the dataset contained many duplicated timestamp values. To eliminate the duplicate features, we mapped 47 timestamps to a single unique timestamp. After the mapping and removal of redundant timestamps, the dataset was reduced to 58 features.

Principal component analysis (PCA) is a dimension reduction technique that projects a set of variables into a smaller set of linearly orthogonal new artificial dimensions, called principle components that capture the largest variances among original observations (Wall, Rechtsteiner, & Rocha, 2003). The original variables can be constructed by using the weight vectors of the PCs. PCA is especially useful when dealing with high dimensional space because the dataset can be represented by a much smaller subspace without having to loose a lot of variation in original data samples (Shaw, 2009; Al-Kandari & Jolliffe, 2005).

The dataset in our case was high dimensional and required applying feature selection. As a dimension reduction technique, PCA was also applied during feature selection phase. The results of PCA indicated that more than 95% of the variance in the dataset can be explained by 17 principle components only. Fig. 3 illustrates PCA variance contributions of the features. It demonstrates how each variable contributes to principal components 1 and 2.

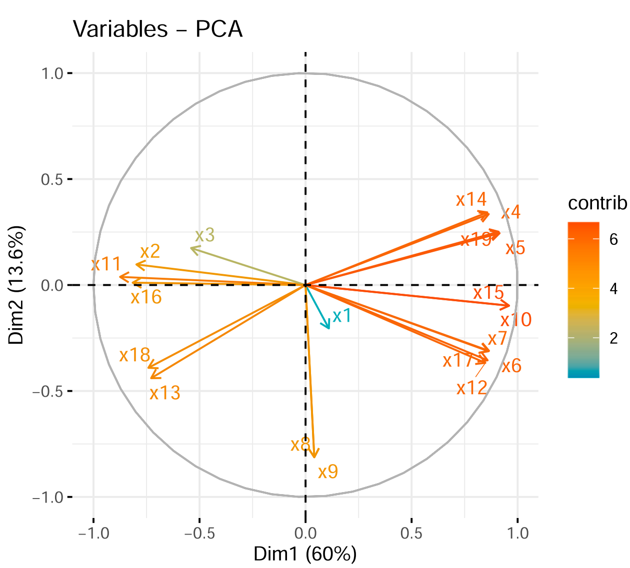

Fig. 3. PCA variable variance contributions.

Moreover, the variable importancies were calculated based on the predictive power of features in Random Forest algorithm (Svetnik et al., 2003) using Gini impurity measure in the study. Gini-based importance of a variable is computed based on the sum of squared errors (SSE) difference if the variable is excluded (Strobl, Boulesteix, Kneib, Augustin, & Zeileis, 2008; Menze et al., 2009). Among all variables, top 10 important features were identified as shown in Fig. 4.

The findings of PCA variable contributions were mostly in line with the variable importancies. We have also considered the expert knowledge in identifying relevant sensor variables mostly associated with common error types. After taking into account the factor analysis and domain expertise, the modeling dataset was finally reduced to 19 features; 18 independent variables and the dependent variable, which is the remaining useful lifetime before the next stop associated with the two error types.

Before proceeding to the modeling phase, the process of splitting the dataset into training and test set has been applied with appropriately. It is the key process for identifying potential modeling issues such as overfitting and underfitting. After the split, 80% of the dataset was allocated to training dataset and the rest held out to the test set. Moreover, 5-fold cross validation technique has been applied in training data to avoid the overfitting due to sampling issues. To prevent the risk of involuntarily leaking information from the test set to the training and indirectly provoking overfitting, the test set was hidden from hyperparameter tuning process. For performing hyperparameter tuning, we used random search with 5-fold cross validation technique, in which 1-fold of training data was held out as a validation set and the remaining folds were used for model training.

4.2. Selecting and training ML models

In predictive maintenance, predicting the class of an instance as a failure or normal operation only is not sufficient. It is more important to correctly estimate the remaining time well before the potential failure to a certain extent that the maintenance or other preventive actions can be taken ahead of time. Hence, the problem can be considered as a regression task rather than a classification problem. In other words, the objective of the model was to predict remaining useful time before the failure in a data-driven approach.

Although the prediction problem in this case may appear like a time series problem, it is not a suitable approach to consider it as one because the model can't verify actual time to failure values at the run time until the next failure happen. Since it is not possible to calculate the prediction errors from input time steps for propagation through time, the predictions will be based on the predictions of previous input time-steps.

To fully understand a particular predictive modeling problem, various configurations, which are also called hyperparameters in

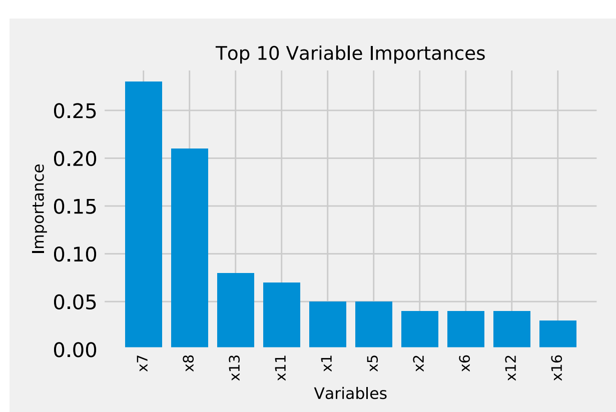

Fig. 4. Top 10 important variables.

machine learning, must be explored and compared against a benchmark. In the analysis, the selection of the algorithms and corresponding hyperparameters determines the success of the model. In this study, multiple ML algorithms and hyperparameters were examined for the purpose of finding the most suitable prediction models. As some equipment and data points varied in production lines, different ML models were developed for each production line. For hyperparameter tuning, random search with 5-fold cross validation technique was used. The random search was chosen over grid search method due to extreme computation cost and inefficiencies of the latter.

For the evaluations, six different algorithms were used: four ensemble and two constituent machine learning algorithms. The ensemble methods, namely were Random Forest, which is a bagging ensemble method, and XGBoost, Gradient Boosting and AdaBoost, which are boosting ensemble algorithms. Moreover, Multilayer Perceptron (MLP) Regressor, a Neural Network model and Support Vector Regression (SVR) algorithm were assessed.

Furthermore, we have explored the three other widely known linear regression algorithms for modeling purposes: Multiple Regression, Lasso Regression and Ridge Regression. However, these algorithms performed poorly due to non-linear nature of our problem. These algorithms were unable to capture any variance in the data. In other words, the models were not able to perform any better than predicting the mean RUL of the sample dataset. Therefore, we omitted the linear models from the evaluations in the study.

Ultimately, the algorithms used in the study were selected primarily for the following reasons: (1) since prediction of RUL is treated as regression problem, the chosen ML algorithms must be applicable to regression; (2) they can be applied to non-linear data as it is in our case; (3) the algorithms need to have high prediction power according to standard performance metrics in literature; (4) the models need to be robust and scalable in high-dimensional data.

After evaluating several different options within the hyperparameter space for the models, the following parameters were found optimal under the specified conditions. For the XGBoost Regressor model, we noticed that the hyperparemeters of maximum depth of 5, the learning rate of 0.3 and the number of estimators of 100 were the best options. Likewise, for training the Random Forest Regressor model, various alternatives were tested to optimize the number of tree estimators. The model parameter with 51 tree estimators was found to be performing well.

Similarly, based on the hyperparameter optimizations in the MLP Regressor model training, it was observed that the MLP Regressor model with relu activation function, 50 hidden layer nodes, epsilon of 1e-08 and adam optimizer appeared to fit well to the dataset.

When fitting the Support Vector Regression (SVR) model to the dataset, the most important SVR parameter is kernel type. Since the dataset was not linearly separable, the linear SVR performed poorly. Upon explorations, we observed that SVR with Radial Basis Function (RBF) kernel appeared to fit relatively well in our case.

For Gradient Boosting model, the hyperparemeters of maximum depth of 2, the learning rate of 0.05, and the number of estimators of 1000 performed the best. On the other hand, Adaboost model was unable to fit the data adequately. Among various tuning options, the hyperparemeters of maximum depth of 2, the learning rate of 0.05, and the number of estimators of 1000 resulted in its best performance.

In the study, Python programming language, Keras and Sci-kit learn machine learning libraries were used for the model development and evaluations. During optimization of methods, learning speed and momentum values were kept as default values. The machine learning models were deployed into production using Flask web-development framework. Flask API exposes the deployed ML models over REST web services interface with the intention that the models can be accessed from the IoT platform securely via HTTP protocol in real-time.

5. Results

In the evaluations, the prediction performances of the algorithms were also assessed using R^2 (R-squared), MAE (Mean Absolute Error), MAPE (Mean Absolute Percentage Error) and RMSE (Root Mean Squared Error) metrics as illustrated in Table 1.

R^2 is the proportion of the total variability in the dependent variable that the model was able to reduce. In other words, it measures the model quality by the fraction of overall uncertainty that the model has helped explain. A higher R^2 value indicates a better fit for the model in general (Draper & Smith, 1998). As shown in Eq. 1, R^2 is defined as:

$$R^2 = 1 - \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{\sum_{i=1}^{n} (y_i - \overline{y})^2} \tag{1}$$

MAE measures the average difference between the model predictions from the actual failures the test set in terms of absolute values (Willmott & Matsuura, 2005). MAE is calculated as shown in Eq. 2:

$$MAE = \frac{\sum_{i=1}^{n} |y_i - \hat{y}_i|}{n} \tag{2}$$

In the formula, $\hat{y}_i$ refers to the predicted remaining useful time value by the algorithms and y_i denotes the actual value in the test dataset. If the MAE value is smaller, it indicates that the algorithm provides better predictions for potential failures. MAPE is very similar to MAE. The main difference is that MAPE measures the variation in percentage as opposed to the absolute values (Chai & Draxler, 2014).

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}} \tag{3}$$

RMSE is another commonly used metric that measures the standard deviation of the prediction errors. Eq. 3 demonstrates the formula used for calculating the RMSE measure (Hyndman & Koehler, 2006; Chai & Draxler, 2014). Similarly, a smaller RMSE value is preferred as it indicates that the model prediction is more accurate.

As a result of evaluations and comparisons, the best results were obtained using Random Forest with a 0.982 R^2 value in the testing dataset. Although the XGBoost method had a very close R^2 value with 0.979, it lagged slightly behind Random Forest. On the other hand, Neural Network model MLP Regressor with had performed worse when compared to the other three ensemble learning methods and shown to be insufficient for this prediction process. The worst performance was observed when using Adaboost and SVR algorithms with 0.338 and 0.347 R^2 values, respectively.

Fig. 5 illustrates a comparison of algorithm results in terms of prediction performance based on the test dataset. An excerpt of records from the evaluations were demonstrated in the figure only for visualization purposes. As shown in the figure, Random Forest and XGBoost algorithms clearly achieved better prediction results than the Gradient

Table 1
Evaluation results for accuracy of ML algorithms on test dataset.

Algorithm	R^2	MAE	MAPE	RSME
Random forest	0.982	51.97	3.27	147.19
XGBoost	0.979	82.09	5.16	157.28
Gradient boosting	0.776	394.52	24.79	523.91
MLP regressor	0.675	466.32	29.30	632.69
SVR	0.347	682.43	42.88	896.07
AdaBoost	0.338	752.95	47.32	902.85

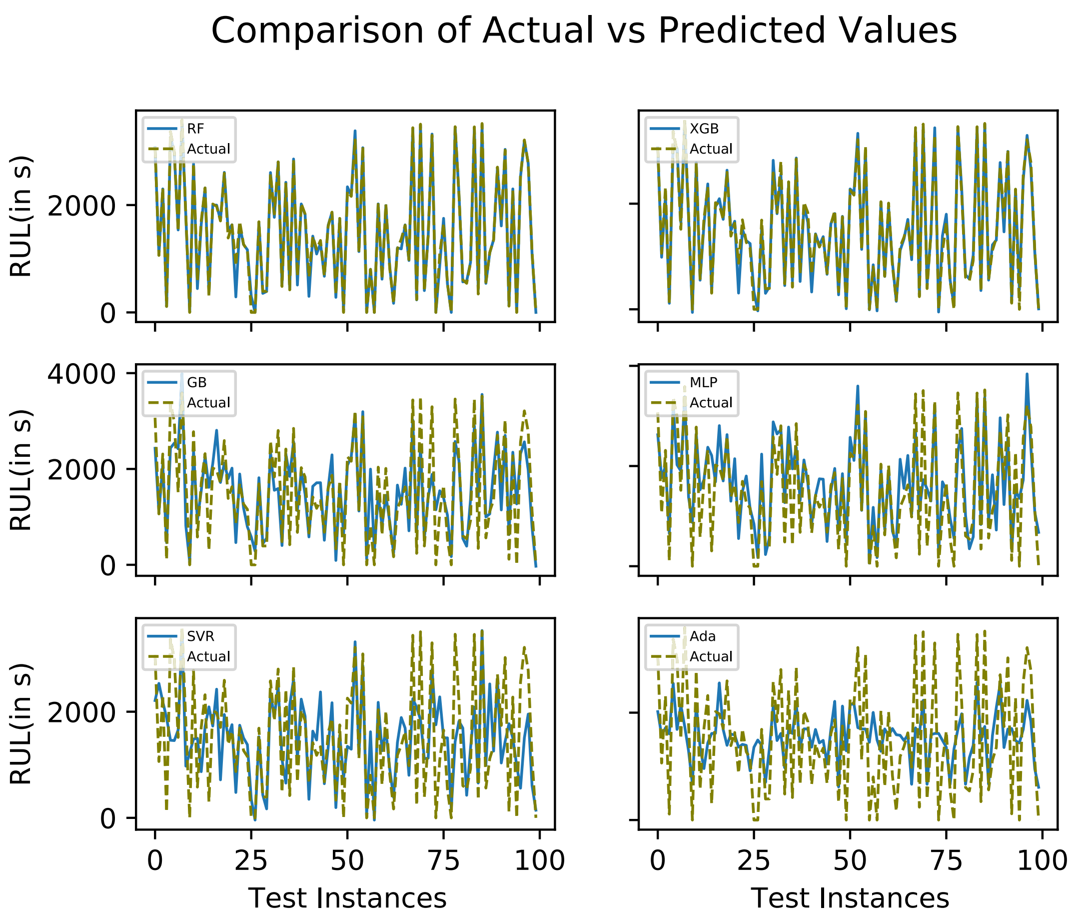

Fig. 5. Comparison of algorithms results: actual vs predicted.

Boosting, AdaBoost, MLP and SVR models.

6. Discussion

In manufacturing, traditional maintenance strategies of production lines typically involve replacement of equipment occurring after a failure or at predetermined time periods. These maintenance plans often lead to extra costs due to early replacement of equipment. The case of late replacement causes unplanned stops in the production lines that result in heavy loss of capacity and products. Although some firms plan maintenance based on lifetime of individual equipment recommended by the vendors, there still exists a high risk of late equipment maintenance situation due to the fact that this maintenance strategy does not take into account the entire system.

As opposed to traditional methods, the predictive maintenance system developed in the study considers entire production line based on data collected from IoT sensors in real-time. The machine learning models evaluate changing conditions and estimate the best time for maintenance of the equipment. Consequently, many potential unplanned line stops, discarded products and capacity losses can be prevented.

While there are many approaches to predictive maintenance in literature, it is an active research area and there is no one fits all solution available. In that context, we think that our approach contributes to the literature in the following ways. Firstly, we provide an end-to-end ML based predictive maintenance approach for manufacturing that integrates all components in a real-life factory environment from IoT sensors to development of ML models and creating alerts. Secondly, to the best of our knowledge, this is the first AI based predictive maintenance system implemented in a real factory in this specific sector. Moreover, we have shown that the system is effective and scalable to high-dimensional fast data in real-time. This is important as many approaches in the literature are developed based on syntactic data and are not suitable for practical use in factory settings. Many approaches fail due to unforeseen challenges associated with the manufacturing environments.

Additionally, the proposed predictive maintenance system has contributed a secondary benefit by facilitating the digital transformation in the production lines. By implementing IoT sensors on the equipment, the production process has become digitized such that the collected data can be used in data analytics for improving the performance of production lines in the future.

For testing the effectiveness and the predictive power of the system, we evaluated multiple machine learning algorithms and investigated various configurations. The results indicated that the models based on three ensemble learning algortims, namely Random Forest, XGBoost and Gradient Boosting, outperformed individual MLP Regressor and SVR algorithms alone. Our findings were in line with the related studies as stated in literature that ensemble learning techniques are known to perform better in general in terms of prediction power than other individual machine learning algorithms (Polikar, 2006; Rokach, 2010). The AdaBoost algorithm was an exceptional case as it has performed poorly in fitting the data. We investigated various hyperparameter tuning options. Yet, the models were unable to achieve better results than constituent counterparts.

A potential limitation of the current study is that the dataset was collected from one factory only. Thus, the machine learning models were based on the production lines from the same factory and may not be generalized to all cases in manufacturing. Nevertheless, the equipment is standardized and commonly available in the industry. The proposed approach is scalable and generalizable. With minor configurations in the system, the approach can be applied to similar production environments. That being said, the ML models cannot be applied directly to new production lines as is because the models must be retrained with new dataset when different IoT devices are used or different failure types are targeted.

On the other hand, we think that similar approach or intelligent function of the proposed approach can also be applied to product quality prediction with integration of additional sensors and image processing functionality for monitoring the final products as the proposed framework lay the foundation for model extensions.

Due to rare occurrences some failure categories in the dataset, we only addressed the most frequent failure types. However, we plan to develop ML models to consider rare failure types as well in the future.

7. Conclusion

In this study, we developed a machine learning based predictive maintenance system for manufacturing environments. We evaluated the effectiveness of the system using real-world manufacturing system IoT data. The output of the model represents the estimated useful time remaining before failure. The evaluations results show that our proposed predictive maintenance system is effective in capturing the signals of machinery failure using real-time sensor data and it can help prevent potential production stops by taking preventive actions suggested by the system. Our findings indicate that boosting and bagging ensemble models perform well in general. For the future work, we plan to apply the system to other types of production lines in different settings.

CRediT authorship contribution statement

Serkan Ayvaz: Conceptualization, Data curation, Methodology, Software, Writing - original draft, Writing - review & editing, Visualization, Funding acquisition, Formal analysis, Investigation, Resources, Validation. **Koray Alpay:** Conceptualization, Data curation, Project administration, Writing - review & editing, Resources, Funding acquisition, Validation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) Grant in TEYDEB-1501 TUBITAK Industrial R&D Projects under Grant No: 3181393 and the Procter & Gamble Turkey. Special thanks to Mesut Saban and Ammar Omar from the Procter & Gamble Turkey, and Orhan Dengiz from DVM Technology for their support and constructive feedback throughout the project.

References

Al-Kandari, N. M., & Jolliffe, I. T. (2005). Variable selection and interpretation in correlation principal components. *Environmetrics: The Official Journal of the International Environmetrics Society, 16*(6), 659–672.

Altman, N., & Krzywinski, M. (2018). The curse (s) of dimensionality. *Nature Methods, 15* (6), 399–400.

Ansari, F., Erol, S., & Sihn, W. (2018). Rethinking human-machine learning in industry 4.0: How does the paradigm shift treat the role of human learning? *Procedia Manufacturing, 23*, 117–122.

Bellman, R. (1966). Dynamic programming. *Science, 153*(3731), 34–37.

Bolton, C., Machová, V., Kovacova, M., & Valaskova, K. (2018). The power of human–machine collaboration: Artificial intelligence, business automation, and the smart economy. *Economics, Management, and Financial Markets, 13*(4), 51–56.

Canziani, A., Paszke, A. & Culurciello, E. (2016). An analysis of deep neural network models for practical applications. arXiv preprint arXiv:1605.07678.

Chai, T., & Draxler, R. R. (2014). Root mean square error (rmse) or mean absolute error (mae)?–arguments against avoiding rmse in the literature. *Geoscientific Model Development, 7*(3), 1247–1250.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). Smote: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research, 16*, 321–357.

Chen, C., Liu, Y., Wang, S., Sun, X., Di Cairano-Gilfedder, C., Titmus, S., & Syntetos, A. A. (2020). Predictive maintenance using cox proportional hazard deep learning. *Advanced Engineering Informatics, 44*, Article 101054.

Draper, N. R., & Smith, H. (1998). *Applied regression analysis* (Vol. 326). John Wiley & Sons.

Edward, K., Dariusz, M., Żabiński, T., Prucnal, S., Se, J. & et al. (2020). Machining sensor data management for operation-level predictive model. Expert Systems with Applications, p. 113600.

Erfani, S. M., Rajasegarar, S., Karunasekera, S., & Leckie, C. (2016). High-dimensional and large-scale anomaly detection using a linear one-class svm with deep learning. *Pattern Recognition, 58*, 121–134.

Fernandes, M., Canito, A., Corchado, J. M., & Marreiros, G. (2019). Fault detection mechanism of a predictive maintenance system based on autoregressive integrated moving average models. In *International symposium on distributed computing and artificial intelligence* (pp. 171–180). Springer.

Francis, F., & Mohan, M. (2019). Arima model based real time trend analysis for predictive maintenance. In *2019 3rd International conference on electronics, communication and aerospace technology (ICECA)* (pp. 735–739). IEEE.

Froger, A., Gendreau, M., Mendoza, J. E., Pinson, E., & Rousseau, L.-M. (2016). Maintenance scheduling in the electricity industry: A literature review. *European Journal of Operational Research, 251*(3), 695–706.

Furman, J., & Seamans, R. (2019). Ai and the economy. *Innovation Policy and the Economy, 19*(1), 161–191.

Gensler, A., Henze, J., Sick, B. & Raabe, N. (2016). Deep learning for solar power forecasting–an approach using autoencoder and lstm neural networks. In 2016 IEEE international conference on systems, man, and cybernetics (SMC) (pp. 002858–002865). IEEE.

He, B., & Bai, K.-J. (2020). Digital twin-based sustainable intelligent manufacturing: A review. *Advances in Manufacturing*, 1–21.

He, Y., Gu, C., Chen, Z., & Han, X. (2017). Integrated predictive maintenance strategy for manufacturing systems by combining quality control and mission reliability analysis. *International Journal of Production Research, 55*(19), 5841–5862.

Ho, S.-L., Xie, M., & Goh, T. N. (2002). A comparative study of neural network and box-jenkins arima modeling in time series prediction. *Computers & Industrial Engineering, 42*(2–4), 371–375.

Huang, J., Chang, Q., & Arinez, J. (2020). Deep reinforcement learning based preventive maintenance policy for serial production lines. *Expert Systems with Applications, 160*, Article 113701.

Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International Journal of Forecasting, 22*(4), 679–688.

Jasiulewicz-Kaczmarek, M., Legutko, S., & Kluk, P. (2020). Maintenance 4.0 technologies–new opportunities for sustainability driven maintenance. *Management and Production Engineering Review, 11*.

Jena, M. C., Mishra, S. K., & Moharana, H. S. (2020). Application of industry 4.0 to enhance sustainable manufacturing. *Environmental Progress & Sustainable Energy, 39*, 13360.

Jeschke, S., Brecher, C., Meisen, T., Özdemir, D., & Eschert, T. (2017). Industrial internet of things and cyber manufacturing systems. In *Industrial internet of things* (pp. 3–19). Springer.

Kanawaday, A., & Sane, A. (2017). Machine learning for predictive maintenance of industrial machines using iot sensor data. In *2017 8th IEEE international conference on software engineering and service science (ICSESS)* (pp. 87–90). IEEE.

Lee, J., Davari, H., Singh, J., & Pandhare, V. (2018). Industrial artificial intelligence for industry 4.0-based manufacturing systems. *Manufacturing Letters, 18*, 20–23.

Lee, J., Kao, H.-A., Yang, S., et al. (2014). Service innovation and smart analytics for industry 4.0 and big data environment. *Procedia Cirp, 16*(1), 3–8.

Lee, W. J., Wu, H., Yun, H., Kim, H., Jun, M. B., & Sutherland, J. W. (2019). Predictive maintenance of machine tool systems using artificial intelligence techniques applied to machine condition data. *Procedia CIRP, 80*, 506–511.

Li, Z., Wang, Y., & Wang, K.-S. (2017). Intelligent predictive maintenance for fault diagnosis and prognosis in machine centers: Industry 4.0 scenario. *Advances in Manufacturing, 5*(4), 377–387.

Locke, D. (2010). Mq telemetry transport (mqtt) v3. 1 protocol specification. *IBM DeveloperWorks Technical Library, 15*.

Lu, Y. (2017). Industry 4.0: A survey on technologies, applications and open research issues. *Journal of Industrial Information Integration, 6*, 1–10.

Lu, Y., Liu, C., Kevin, I., Wang, K., Huang, H., & Xu, X. (2020). Digital twin-driven smart manufacturing: Connotation, reference model, applications and research issues. *Robotics and Computer-Integrated Manufacturing, 61*, Article 101837.

Machado, C. G., Winroth, M. P., & Ribeiro da Silva, E. H. D. (2020). Sustainable manufacturing in industry 4.0: An emerging research agenda. *International Journal of Production Research, 58*(5), 1462–1484.

Menze, B. H., Kelm, B. M., Masuch, R., Himmelreich, U., Bachert, P., Petrich, W., & Hamprecht, F. A. (2009). A comparison of random forest and its gini importance with standard chemometric methods for the feature selection and classification of spectral data. *BMC Bioinformatics, 10*(1), 213.

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.

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.

Polikar, R. (2006). Ensemble based systems in decision making. *IEEE Circuits and Systems magazine, 6*(3), 21–45.

Powell, W. B. (2007). *Approximate dynamic programming: Solving the curses of dimensionality* (Vol. 703). John Wiley & Sons.

Rabatel, J., Bringay, S., & Poncelet, P. (2011). Anomaly detection in monitoring sensor data for preventive maintenance. *Expert Systems with Applications, 38*(6), 7003–7015.

Ramos, P., Oliveira, J. M. S., & Silva, P. (2014). Predictive maintenance of production equipment based on neural network autoregression and arima. In *21st International EurOMA conference-operations management in an innovation economy*.

Rieger, T., Regier, S., Stengel, I., & Clarke, N. L. (2019). Fast predictive maintenance in industrial internet of things (iiot) with deep learning (dl): A review. *CERC*, 69–80.

Rivera, D. L., Scholz, M. R., Bühl, C., Krauss, M., & Schilling, K. (2019). Is big data about to retire expert knowledge? A predictive maintenance study. *IFAC-PapersOnLine, 52* (24), 1–6.

Rokach, L. (2010). Ensemble-based classifiers. *Artificial Intelligence Review, 33*(1–2), 1–39.

Shaw, P. J. (2009). *Multivariate statistics for the environmental sciences*. Wiley.

Song, X., Kanasugi, H., & Shibasaki, R. (2016). Deeptransport: Prediction and simulation of human mobility and transportation mode at a citywide level. *IJCAI, 16*, 2618–2624.

Song, Z., & Moon, Y. (2017). Assessing sustainability benefits of cybermanufacturing systems. *The International Journal of Advanced Manufacturing Technology, 90*(5–8), 1365–1382.

Song, Z., & Moon, Y. (2018). Cybermanufacturing system: A solution for sustainable manufacturing. In *ASME international mechanical engineering congress and exposition* (volume 52019)American Society of Mechanical Engineers. page V002T02A068.

Stanford-Clark, A. & Hunkeler, U. (1999). Mq telemetry transport (mqtt). Online]. http://mqtt. org. Accessed September, 22:2013.

Stock, T., & Seliger, G. (2016). Opportunities of sustainable manufacturing in industry 4.0. *Procedia Cirp, 40*, 536–541.

Strobl, C., Boulesteix, A.-L., Kneib, T., Augustin, T., & Zeileis, A. (2008). Conditional variable importance for random forests. *BMC Bioinformatics, 9*(1), 307.

Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., & Feuston, B. P. (2003). Random forest: A classification and regression tool for compound classification and qsar modeling. *Journal of Chemical Information and Computer Sciences, 43*(6), 1947–1958.

Traini, E., Bruno, G., D'Antonio, G., & Lombardi, F. (2019). Machine learning framework for predictive maintenance in milling. *IFAC-PapersOnLine, 52*(13), 177–182.

Wall, M. E., Rechtsteiner, A., & Rocha, L. M. (2003). Singular value decomposition and principal component analysis. In *A practical approach to microarray data analysis* (pp. 91–109). Springer.

Wamba-Taguimdje, S.-L., Wamba, S. F., Kamdjoug, J. R. K., & Wanko, C. E. T. (2020). Influence of artificial intelligence (ai) on firm performance: The business value of ai-based transformation projects. *Business Process Management Journal*.

Wang, J., Zhang, L., Duan, L., & Gao, R. X. (2017). A new paradigm of cloud-based predictive maintenance for intelligent manufacturing. *Journal of Intelligent Manufacturing, 28*(5), 1125–1137.

Wang, S., Wan, J., Zhang, D., Li, D., & Zhang, C. (2016). Towards smart factory for industry 4.0: A self-organized multi-agent system with big data based feedback and coordination. *Computer Networks, 101*, 158–168.

Willmott, C. J., & Matsuura, K. (2005). Advantages of the mean absolute error (mae) over the root mean square error (rmse) in assessing average model performance. *Climate Research, 30*(1), 79–82.

Xie, X., Wu, D., Liu, S. & Li, R. (2017). Iot data analytics using deep learning. arXiv preprint arXiv:1708.03854.

Yu, W., Dillon, T., Mostafa, F., Rahayu, W., & Liu, Y. (2019). A global manufacturing big data ecosystem for fault detection in predictive maintenance. *IEEE Transactions on Industrial Informatics, 16*(1), 183–192.

Zhang, J., Wang, P., Yan, R., & Gao, R. X. (2018). Deep learning for improved system remaining life prediction. *Procedia CIRP, 72*, 1033–1038.

Zhang, J., Wang, P., Yan, R., & Gao, R. X. (2018). Long short-term memory for machine remaining life prediction. *Journal of Manufacturing Systems, 48*, 78–86.

Zhao, G., Liu, X., Zhang, B., Zhang, G., Niu, G., & Hu, C. (2017). Bearing health condition prediction using deep belief network. In *Proceedings of the annual conference of prognostics and health management society, Orlando, FL, USA* (pp. 2–5).