A Predictive Maintenance Application for A Robot Cell using LSTM Model

(article starts on next page)

Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 55-19 (2022) 115–120

A Predictive Maintenance Application for A Robot Cell using LSTM Model

Doyel Joseph*, Tilani Gallege*, Ebru Turanoglu Bekar*, Catarina Dudas, Anders Skoogh***

Department of Industrial and Materials Science, Chalmers University of Technology, 41296, Gothenburg, Sweden (e-mail: doyel@student.chalmers.se, tilani@student.chalmers.se, ebrut@chalmers.se, anders.skoogh@chalmers.se).

***Department of Advanced Analytics and AI, Volvo Cars AB, Gothenburg, Sweden (e-mail: catarina.dudas@volvocars.com)*

Abstract: Maintaining equipment is critical for increasing production capacity and decreasing production time. With the advent of digitalization, industries are able to access massive amounts of data that can be used to ensure their long-term viability and competitive advantage by implementing predictive maintenance. Therefore, this study aims to demonstrate a predictive maintenance application for a robot cell using real-world manufacturing big data coming from a company in the automotive industry. A hyperparameter tuned Long Short-Term Memory (LSTM) model is developed, and the results show that this model is capable of predicting the day of failure with good accuracy. The difficulties inherent in conducting real-world industrial initiatives are analyzed, and recommendations for improvement are presented.

Keywords: Smart Maintenance, Predictive Maintenance, Machine Learning, Long Short-Term Memory (LSTM), CRISP-DM, Industrial Robots, Manufacturing

1. INTRODUCTION

The industrial revolution's amazing growth resulted in significant advancements in domains such as automation, rapid manufacturing, and big data analytics. The fourth industrial revolution (Industry 4.0) is enabled by highly sophisticated, dynamic, and integrated information systems, as well as extremely powerful computational capabilities, (Lasi et al., 2014). Manufacturing companies generate a great deal of data because of these complex systems, but many of them are unsure how to make the right use of the data to accelerate data-driven decision-making for their competitive and sustainable future (Tsvetkova, 2017).

When it comes to change adaptation, the automotive industry has always been a pioneer. This is mainly due to the enormous amount of competition and incriminating challenges to meet customer expectations (Shayganmehr et al., 2021). The situation is similar in the utilization of data analytics to enhance the standard and quality of output. Due to this advantage, this study is conducted at a gluing robot station within a reputable Automotive company, which we refer to as company X. The current maintenance practices at company X are based on a time-based preventive maintenance strategy. One notable downside of this technology is that it wastes a considerable percentage of the machine's useful life (Löfsten, 2000).

Therefore, the main aim of this study is to estimate the failure of a gluing robot at company X's production plant using Machine Learning (ML) techniques. The aim in terms of the PdM process is to predict the occurrences of severe alarms by using the LSTM model considering the past available data. This would contribute to simplifying the maintenance schedule for this robot cell through data-driven decision-making, and thus it would enable a failure-free production in the longer term as the vision of predictive maintenance. In this study, a systematic methodology, which is called the Cross-Industry Standard for Data Mining (CRISP-DM) (Wirth and Hipp, 2000) is also utilized in order to overcome some challenges in real-world data analytics projects, i.e. lack of high-quality data and difficulties for driving actionable insights from data (Lee et al., 2014).

This paper is divided into five sections. Section 2 discusses the state-of-the-art studies that can be gleaned from comparable works. Section 3 details the formulated methodology used in this study. Section 4 summarizes the study's findings. Section 5 presents a summary of conclusions together with future research studies.

2. RELATED WORKS

Numerous research studies have been conducted on data-driven decision-making for predictive maintenance. Baptista et al. (2018) compared the various Artificial Intelligence (AI) and statistical approaches to predictive maintenance and concluded that the AI approach produces a better result than the statistical approach. They also discussed the ability of ML algorithms to handle high-dimensional multivariate data for predictive maintenance applications in various industries. Due to the nature of this study's content, a review of the literature using the LSTM model was undertaken and detailed in the following paragraphs.

Lim et al. (2021) evaluate numerous multivariate time series prediction models and discovered that the Long Short-Term

10.1016/j.ifacol.2022.09.193

Memory (LSTM) model was capable of handling missing values in time series data. The prediction findings were validated against Auto-Regressive Integrated Moving Average (ARIMA) - and Vector Autoregression (VAR) - based prediction models. As a result, the LSTM model was found to be more accurate than all the other traditional techniques. Zhang et al. (2019) applied an LSTM model for multivariate time series prediction. They observed during the training of the LSTM model that the Adam optimization algorithm's continuous weight updating and hyper-parameter adjustment to achieve the optimal number of layers and batch size were critical for good and more accurate results. In the literature, the LSTM technique has been also widely applied for different predictive maintenance applications such as early anomaly detection, constructing health indicators, and remaining useful life estimation of different industrial systems such as bearings and cutting tools (Bampoula et al., 2021; Guo et al., 2017; Ning et al., 2018). On the other hand, there have been not many research studies regarding predictive maintenance applications using AI/ML for industrial robots (Aivaliotis et al., 2021). Therefore, this study contributes to the part of this literature by demonstrating a real-world industrial application based on a structured methodology for failure prediction of a robot cell using the LSTM technique.

3. METHODOLOGY

The CRISP-DM is followed as a reference model in this study since it provides a structured methodology for planning, managing, and ensuring high-quality data in data analytics projects with six iterative phases (i.e., business understanding, data understanding, data preparation, modeling, evaluation, and deployment) (Wirth and Hipp, 2000). However, it should be noted that the CRISP-DM methodology is taken a basis in this study. Hence, it is modified with a stronger emphasis on the data collection phase, as the optimal source was determined using iteration. The modified CRISP-DM methodology is depicted in Figure 1.

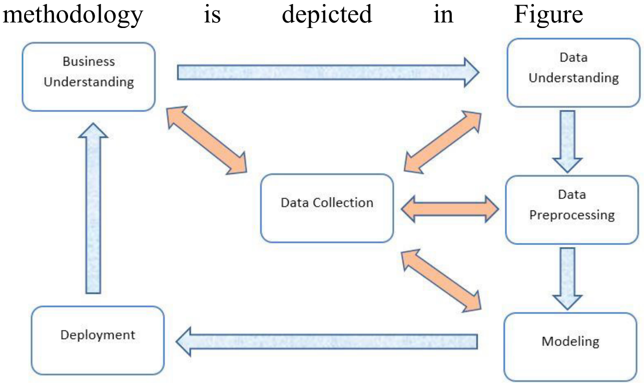

Figure 1. Enhanced Methodology - adopted from the original CRISP-DM (Wirth and Hipp, 2000)

As it is shown in Figure 1, it is an adapted version of the original CRISP-DM model. The change we have included in the adopted model is the connection with the phase of data collection in all the phases such as business understanding, data understanding and pre-processing, and modeling. The reason for this adaptation is the lack of sufficient data at the beginning of the analysis. Therefore, the orange color arrows represent the adding up test data to the already available data set and continuing the process.

3.1 Business understanding

In this study, the company wanted to develop a predictive maintenance application for a glue station in their assembly line.

The gluing stations' primary components are divided into two sections: the robot and the adhesive. Following a detailed discussion with the project's stakeholders, specifically the maintenance department, it was determined that the gluing robot system was robust, and thus the project needs to focus on the adhesive components (i.e., doser, gun, and docking station), which are most prone to failure, and therefore they require the highest rate of maintenance services. As a result, we evaluate solely data from the doser and gun will be explained detailed in the next section. Additionally, temperature, production log data, and alarm event data (alarm log) can be collected from sensors situated throughout the gluing apparatus. There are three types of alarms and warnings: 'A', 'B', and 'D'. Category 'A' alarms are those that require immediate action, category 'B' alarms are those that require critical error correction, and category 'D' alarms are those that provide warning and information. The occurrence of category 'A', 'B', or severe alarms is interpreted as indicating a proclivity for failure. Therefore, the objective of the study is to predict these severe alarms beforehand.

3.2 Data understanding

The phase of data comprehension begins with data collection in order to conduct data analysis. The primary functions are data familiarization, data quality assessment, and data visualization to get the first insights from the data. The purpose of data understanding is to uncover hidden patterns and characteristics that may affect the results; hence, data quality concerns need to be addressed effectively in this phase. Visualizations and pattern discovery were carried out using software tools such as Power BI and Python.

3.3 Data preparation

Even if the data set for this study is recorded automatically with machines, there are still missing numbers and the requirement for data transformation. Preprocessing of data is critical for the quality of prediction, and thus, in this study, the data is normalized using an appropriate normalization technique since it comes from multi-sources having varied fundamental units. For example, temperature logs from various components of the glue robot are in Celsius, whereas the doser log contains a variety of measurements including different units such as real flow data (ml/s), torque (Nm), and pressure (bar).

3.4 Modeling and Evaluation

During the modeling phase, the most appropriate models are chosen and analyzed using a variety of ML models in the open-source software Python. Selecting the optimal model is primarily done based on the produced results that satisfy the business objective. In addition to this, it should be also suitable for the existing data set in order to obtain the best outcomes. Hyperparameter tuning is performed to determine the optimal parameters by minimizing losses while providing the highest accuracy.

Validation and evaluation of ML models are critical for achieving the best results and continuously refining them (Joseph and Gallege, 2021). In this study, the total number of rows was 24284, out of which 80% was used for training (19427) and 20% was a holdout for testing (4856). There are numerous techniques for evaluating the prediction accuracy using test data; in this study, we use the absolute mean error (MAE) and the root mean squared error (RMSE). Equations 1 and 2 illustrate the MAE and RMSE, respectively.

$$MAE = \frac{1}{n}\sum_{t-1}^{n}|y - y'| \tag{1}$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{t-1}^{n}(y - y')^2} \tag{2}$$

3.5 Deployment

The study does not cover the deployment phase, but rather makes recommendations to the company. Those recommendations are about the right data collection and management in order to help them for enhancing the quality of the data in future applications of smart maintenance.

4. RESULTS AND DISCUSSIONS

This section presents the findings of the study, which is achieved by following the structured CRISP-DM methodology. Additionally, we will seek to glean relevant insights from the results and investigate any gaps in order to formulate improvement recommendations for the company.

4.1 Business understanding

The objective of the project is concluded which is to predict the severe alarm occurrences with the past available data. And discussions were carried out understand the relevant sources to collect more data to have a more reliable prediction.

4.2 Data understanding

The data is gathered from a variety of sources, namely PLC, Teamster logs, Maximo, Axxos, and a SQL database. The temperature, doser, and alarm logs were extracted from the Teamster system. Temperature logs include data from the chamber, gun, hose, dock valve, plate, and body sensors, with a timestamp. The doser log contains information about the doser volume, the servomotor moment, the flow reference, and the pressure. The alarm log can be queried for alarm event data. The alarm log CSV file contains the alarm category, alarm ID, timestamp, and necessary description (Joseph and Gallege, 2021). Different problems related to the quality of the data were discovered during the implementation of the study. Therefore, numerous data preparation approaches were required to prepare the data before fitting it into an ML model. The majority of data sources were automatically captured and so possessed a high degree of reliability. When it comes to data relevance, the temperature and doser data were found to be relevant for predictive maintenance using ML. However, one of the biggest challenges was missing labels in the data required extra attention during the data preparation phase. This was due to the less-than-accurate data from the maintenance log (Maximo data). All data were available in CSV format and were easily accessible via software such as Python and Power BI. For several data sources, the files lacked headings, which were added during the pre-processing stage. However, the general structure and readability of the data were excellent.

4.3 Exploratory Data Analysis

In this section, we make use of data visualization techniques to analyze the quintessential patterns concealed inside the data collection. The graphic in Figure 2 depicts the relationship between temperature and triggered alarms over time. From this result, it can be deduced that the normal working temperature of the gluing machine is between 40^0C to 50^0C. It can also be visualized that temperature logging occurs even when the machine is not in use. On the other hand, as illustrated in Figure 3, the doser parameters are only registered when the machine is operational. Following each gluing cycle, the doser volume is returned to its initial value, while the doser volume continuously decreases during the gluing process. Negative values were reported for both real and reference flow, indicating that glue is flowing in the opposite direction (Joseph and Gallege, 2021).

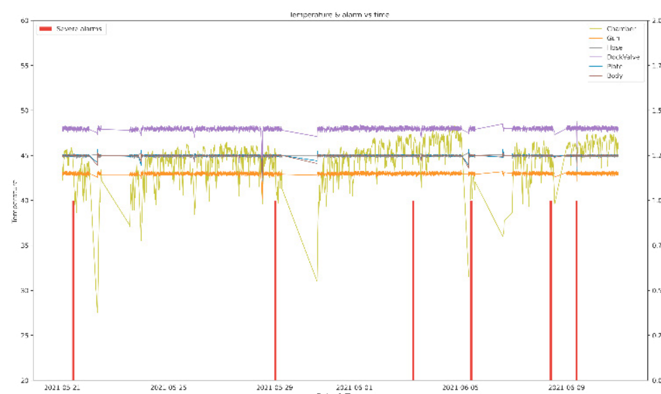

Figure 2. Temperature alarm plot

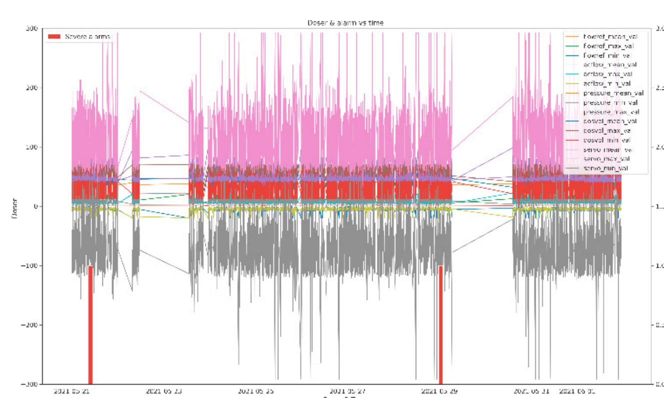

Figure 3. Doser alarm plot

From Figures 2 and 3, it is observed that a failure or breakdown occurrence is somewhat related to the severe alarm occurrence. This was a crucial discovery from the exploratory analysis because it assisted us in approximating the occurrence of critical alarms as machine failures. With this assumption, severe alerts are labeled as failures, and the ML model is expected to forecast their occurrences.

4.4 Data preparation

The initial step in data preparation is to merge all temperature logs into a single file, as well as all doser and alarm files. The doser log was received without headers, which were inserted later when the files were read into Python via the Pandas library. Additionally, several of the titles in the temperature and alarm dataset were modified to improve readability. The timestamp in object format was converted into timestamp format. The severe alarms were isolated by generating a new data frame that contained only the rows with the priority 'A' and 'B'.

The temperature data was merged into the doser data by filling out the temperate value in the neighborhood of the doser time stamp. However, the fundamental disadvantage of this method was overfitting. Bilbao and Bilbao (2017) explained that overfitted models produce excellent outcomes with train data but bad results with test data. Thus, to minimize overfitting, the data points were shrunk from microsecond to minute scale. This had a positive impact on merging the different time scales as well as reducing the overfitting. However, the primary obstacle to data compression is the risk of losing vital information. The most effective way to accomplish this was to employ several strategies, such as calculating the mean, minimum, and maximum statistics of all values within a minute time frame. It was a trial-and-error process, but the best results were produced by utilizing mean, minimum, and maximum numbers. Each property in the doser data has been replaced with the mean, median, and mode of all values within a one-minute time period.

Following this stage, the temperature, processed doser, and severe alert data frames were combined into a single data frame. The doser and temperature data frames were combined up to the timestamp column with a tolerance of 10 minutes. This new data frame is then combined with the severe alarm data frame up to a tolerance of 30 minutes and a timestamp. From 6th May 2021, continuous data were available from all data sources, and so many rows with a timestamp prior to this date were excluded from the merged data frame.

Due to the fact that there are only a few severe alarms in the entire data frame, all other values were designated as null values. Zeros were substituted for these null values. All severe alarms, that is, those with a priority of 'A' or 'B', have been transformed to the float value 1.0. Thus, the binary digit, i.e. ‘1’ indicates the appearance of a serious warning, while ‘0’ indicates the gluing machine is operating normally. The fundamental units of the entities are different in the final data set, for example, pressure measurements are expressed in bars, temperatures in degrees Celsius, and flows in ml/s. The Sklearn standard scaler from the pre-processing package was used for the normalization of the final dataset. The aim is to reduce the different measurements to a neutral (i.e. standard) scale. If the sample is referred to as x, the mean as u, and the standard deviation as s, then the standard score is calculated using Equation 3. After statistical computation on the training dataset, scaling is applied to individual attributes. The data is transformed using the stored mean and standard deviation.

$$z = (x - u)/s \tag{3}$$

And the entire dataset was split into train and test classes with the ratio of 80% of train data (19427) and 20% of test data (4856).

4.5 Modeling and Evaluation

This section presents ML modeling and the parameters utilized in the models, as well as the obtained findings. Following a thorough review of the literature, it was determined that LSTM was the best model for time series prediction to implement with the mean squared error (MSE) as the loss function. Furthermore, modeling is a critical stage in getting the desired outcome, and as noted in the methodology section, this approach was iterative.

Preparing data in an appropriate format is a critical step in implementing the LSTM model. This data preparation should be performed on both the training and test datasets. To train LSTM networks, data must be reshaped in the format of N samples x time steps. Preprocessing is performed in such a way that, if n rows are used for training, the $n + 1^{st}$ row is predicted, and this sliding window is shifted by one row. For instance, if the first to fourteenth values are used for training in the first sliding window, the fifteenth value is used for prediction, and the sliding window then advances one row forward, using the second to fifteenth rows for training and the sixteenth row for prediction. All properties are reshaped similarly in two lists (trainX and trainY) using a loop. For more theoretical background and application details of the LSTM, we refer to the Master thesis study done by (Joseph and Gallege, 2021). There is only one attribute for train Y because the prediction is only to be done on the alarm column, as a value of 0 or 1, where 0 indicates that the gluing machine is operating normally and 1 indicates the occurrence of a severe alert. On the test dataset, the same data pre-processing was performed.

The next critical step was to create the model architecture for the LSTM model. If the number of modules is less than the optimal number, there is a risk of underfitting; if the number of modules is greater than the optimal number, there is a significant risk of overfitting. To avoid these concerns, hyper-parameter tuning is critical to determining the optimal number of modules and layers (Merity et al., 2017). This study utilized Keras Tuner to perform hyper-parameter tuning with a random search technique. The input parameter for each LSTM layer was in the range of 16 to 256 LSTM modules, and the search method used a step value of 16 modules. All permutations and combinations within this range were run for a total of five layers. After the two LSTM layers, a drop-out layer was added to help minimize overfitting by randomly removing data from the neural network (Joseph and Gallege, 2021). The remaining inputs were scaled up by 1/(1-rate) to maintain the same overall inputs as proposed in a study done by (Chollet et al., 2015). Adam optimizer was used for first-order gradient-based optimization (Merity et al., 2017). Following hyper-parameter adjustment, the optimal model was determined to be a two-layer LSTM model, which is indicated LSTM 1 and LSTM 2 with 24 modules in the first layer and 32 modules in the second layer. Typically, the parameters of a neural network are the connection weights. These parameters are learned during the training stage. In this case, the algorithm (together with the input data) adjusts these parameters (Sung, 1998). The best model architecture is presented in Table 1.

Table 1: The architecture of the LSTM model

Layer (Type)	Output Shape	No. of parameters
LSTM 1	(none, 14, 24)	4512
LSTM 2	(none, 32)	7296
Dropout	(none, 32)	0
Dense	(none, 1)	33

To minimize processing costs and run time during hyper-parameter adjustment, only ten epochs were employed for training in the beginning. The optimal number of epochs was obtained by the trial-and-error method. From training loss Figures 4a, 4b, 4c, 4d it was obvious that when the number of epochs is increased, test loss increases. This indicates that the model becomes overfitted as the number of epochs increases. Thus, in this study, 25 epochs for training were chosen since it provides the greatest outcomes with the least amount of running time and processing expense.

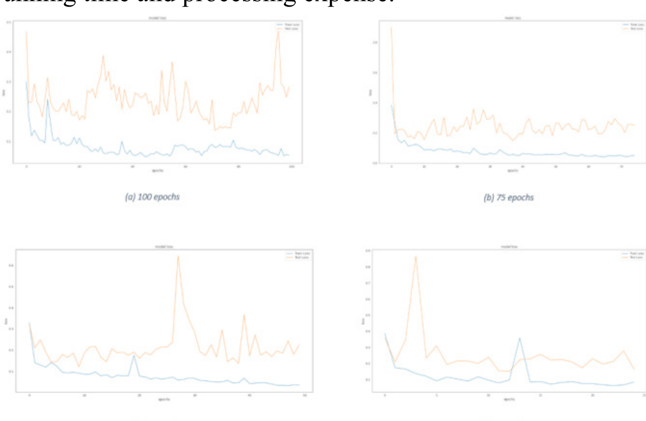

Figure 4. Training and test loss

After developing the LSTM model with the set parameters, cross-validation for forecasting future alarms was performed without providing any dependent variables from the test data, as the model's future dependent variables are unknown in real-time. The output of this prediction was probability values in the range between zero and one and was then classified manually to alarm occurring and not occurring (depicted in Figure 5). Since the initial predictions of the LSTM model were continuous, MAE and RSME matrices were used for the evaluation and their calculated values were 0.2518 and 1.5539, respectively. The x-axis of the graph in Figure 5 represents the future time stamp, while the red lines represent the predicted alerts, and the blue lines represent the actual alarms. As shown in Figure 5, the LSTM model is capable of predicting the day on which the severe warning will occur. It should be also noted that the predictions contain some false alarms as well. Additionally, the false alarms might be caused due to the fact that a worker can open the station's door or there might be another reason unrelated to the used data. The positive aspect is that during cross-validation, none of the genuine alarms were missed. As a result, it can be inferred that the LSTM model is capable of meeting business objectives by anticipating severe alerts with high accuracy and consequently machine failure, thereby this model can assist the maintenance department in performing predictive maintenance on the glue robot.

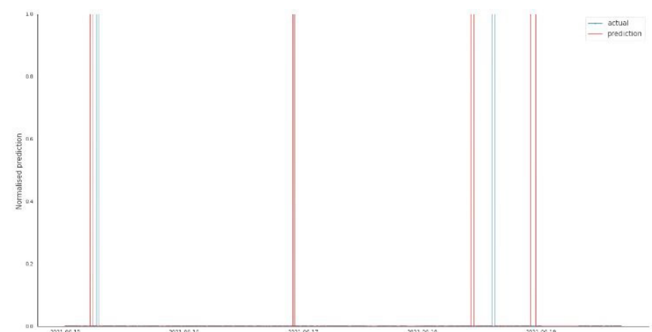

Figure 5. Prediction of future alarms after inputting dependent variables

For the same investigation and comparison, an ARIMA model was built (Gilbert, 2005), and it was discovered that the ARIMA model was only capable of forecasting continuous variables such as pressure, temperature, and so on. Prediction using the ARIMA method had an MAE of 4.3023 and an RSME of 4.9151. The lower the value of these terms, the more accurate the model, therefore it demonstrates that the LSTM model was capable of performing time series classification in this study.

5. DISCUSSIONS

One of the main contributions of this study was to make insightful recommendations to the company to assist them in achieving their goal of smart maintenance using data-driven decision-making. Throughout the exploratory data analysis, data were scarce, making it difficult to conduct modeling properly and thus required a lot of time for data preparation. Thus, by saving pertinent data in a database, these unnecessary time losses can be avoided (Baptista et al., 2018). In addition to this, because the gathered data was coming from multiple sources, it has also required lots of attention in order to merge them in a reasonable manner. This issue has been also highlighted as a common challenge in implementing predictive maintenance, especially in real-world industrial environments (Kurrewar et al., 2021). Thus, in this study, it is vital to combine carefully the data from a variety of sources to estimate the failures using ML. There is another observation during the implementation of this study, some data which can create useful information for data labeling cannot be used due to non-automated (manually entering) data collection. This observation is also one of the important lessons learned for the company and recommended to improve the system in a more automated way. Although we used severe alarms to categorize the data in this study, the model's accuracy would have been much improved if we had used breakdown records instead of alarms as explored in the study done by Johnson et al., 2010). Additionally, the company might explore collaborating with

Original Equipment Manufacturer' suppliers, which would facilitate knowledge transfer from domain specialists to the company's data scientists for the purpose of creating additional data sources for the efficient implementation of predictive maintenance. Finally, this study was focusing on a single gluing robot for a short period of time. Better predictions might be produced if the model had observed the gluing robot in a variety of operating conditions. One of the most effective ways to accomplish this is to incorporate data from similar gluing robots found in the same production plant; this diversifies the data set.

5. CONCLUSION

This study demonstrates a predictive maintenance application for a robot cell by estimating the day of failure based on the LSTM model and an enhanced version of the CRISP-DM methodology. From an academic point of view, this study can be a good reference for future research and studies dealing with data-driven approaches for real word predictive maintenance applications in the industry. The results demonstrate that the developed LSTM model is capable of forecasting the day of the breakdown based on the severity of the alarms. Therefore, this study can assist maintenance personnel in scheduling and organizing maintenance activities and provide a good establishment to the company towards robust implementation of predictive maintenance as a core concept of smart maintenance.

ACKNOWLEDGEMENT

This paper was produced from the MSc thesis of the first and second authors at the Chalmers University of Technology. The authors would like to thank the Maintenance department of the company for providing guidance and support with real-world data.

REFERENCES

Aivaliotis, P., Arkouli, Z., Georgoulias, K., & Makris, S. (2021). Degradation curves integration in physics-based models: Towards the predictive maintenance of industrial robots. *Robotics and Computer-Integrated Manufacturing*, *71*, 102177.

Bampoula, X., Siaterlis, G., Nikolakis, N., & Alexopoulos, K. (2021). A Deep Learning Model for Predictive Maintenance in Cyber-Physical Production Systems Using LSTM Autoencoders. *Sensors*, *21*(3), 972.

Baptista, M., Sankararaman, S., de Medeiros, I. P., Nascimento Jr, C., Prendinger, H., & Henriques, E. M. (2018). Forecasting fault events for predictive maintenance using data-driven techniques and ARMA modeling. *Computers & Industrial Engineering*, *115*, 41-53.

Bilbao, I., & Bilbao, J. (2017, December). Overfitting problem and the over-training in the era of data: Particularly for Artificial Neural Networks. In *2017 eighth international conference on intelligent computing and information systems (ICICIS)* (pp. 173-177). IEEE.

Chollet, F. et al. (2015). Keras. URL https://github.com/fchollet/keras.

Gilbert, K. (2005). An ARIMA supply chain model. *Management Science*, *51*(2), 305-310.

Guo, L., Li, N., Jia, F., Lei, Y., & Lin, J. (2017). A recurrent neural network based health indicator for remaining useful life prediction of bearings. *Neurocomputing*, *240*, 98-109.

Johnson, C. R., Montanari, M., & Campbell, R. H. (2010). Automatic management of logging infrastructure. In *CAE Workshop on Insider Threat*.

Joseph, D., & Gallege, T. N. (2021). A study on remaining useful life estimation for predictive maintenance of a robot cell.

Kurrewar, H., Bekar, E. T., Skoogh, A., & Nyqvist, P. (2021, September). A Machine Learning Based Health Indicator Construction in Implementing Predictive Maintenance: A Real World Industrial Application from Manufacturing. In *IFIP International Conference on Advances in Production Management Systems* (pp. 599-608). Springer, Cham.

Lasi, H., Fettke, P., Kemper, H. G., Feld, T., & Hoffmann, M. (2014). Industry 4.0. *Business & information systems engineering*, *6*(4), 239-242.

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

Lim, S., Kim, S. J., Park, Y., & Kwon, N. (2021). A deep learning-based time series model with missing value handling techniques to predict various types of liquid cargo traffic. *Expert Systems with Applications*, *184*, 115532.

Löfsten, H. (2000). Measuring maintenance performance–in search for a maintenance productivity index. *International Journal of Production Economics*, *63*(1), 47-58.

Merity, S., Keskar, N. S., & Socher, R. (2017). Regularizing and optimizing LSTM language models. *arXiv preprint arXiv:1708.02182*.

Ning, Y., Wang, G., Yu, J., & Jiang, H. (2018, September). A feature selection algorithm based on variable correlation and time correlation for predicting remaining useful life of equipment using rnn. In *2018 Condition Monitoring and Diagnosis (CMD)* (pp. 1-6). IEEE.

Shayganmehr, M., Kumar, A., Garza-Reyes, J. A., & Moktadir, M. A. (2021). Industry 4.0 enablers for a cleaner production and circular economy within the context of business ethics: a study in a developing country. *Journal of Cleaner Production*, *281*, 125280.

Sung, A. H. (1998). Ranking importance of input parameters of neural networks. *Expert systems with Applications*, *15*(3-4), 405-411.

Tsvetkova, R. (2017). What does Industry 4.0 mean for sustainable development?. *Industry 4.0*, *2*(6), 294-297.

Wirth, R., & Hipp, J. (2000, April). CRISP-DM: Towards a standard process model for data mining. In *Proceedings of the 4th international conference on the practical applications of knowledge discovery and data mining* (Vol. 1, pp. 29-39). London, UK: Springer-Verlag.

Zhang, T., Song, S., Li, S., Ma, L., Pan, S., & Han, L. (2019). Research on gas concentration prediction models based on LSTM multidimensional time series. *Energies*, *12*(1), 161.