Predicting Machine Damage Accumulation using a Data driven Surrogate model

CSD402: Internet and Web Systems
Project Report

Aditya Jain, Manav Prabhakar, Sanjeeth Rajeshwaran

December 8, 2021

Abstract

Data-driven models are one of the most popular types of surrogates due to their versatility, low evaluation cost, easy handling, and a large number of modeling techniques with ready-to-use implementations. In this work, we propose a pipe-lined architecture that implements Support Vector Regressors, Linear Regressors and LSTM based models for predicting damage accumulation based on given sensor values. Further, an ARIMA based architecture was used for forecasting individual sensor values (temperature) which can then be used by our data-driven models for predicting damage accumulation, thus enabling predictive maintenance. We also propose a novel scheduling algorithm - SAM (Student Activity Management) Scheduler for slot scheduling at laboratories in an academic setup that exploits predictive model for maximum utilization without causing maintenance induced interruptions.

Contents

1 Introduction

These days, we have manufacturing machines loaded with sensors that provide critical usage data. These sensors record a variety of values such as temperature, velocity, and acceleration in regular time intervals. The sensors also provide a computed damage accumulation measure, which is helpful for predictive maintenance.

Damage accumulation in materials is a consequence of micro-defect evolution, which may lead to degradation of mechanical properties up to failure. It may also be defined with the help of Miner's rule that operates on the hypothesis that the portion of useful fatigue life used up by a number of repeated stress cycles at a particular stress is proportional to the total number of cycles in the fatigue life, if that were the only stress level applied to the part. The theory of cumulative damage assumes that a stress cycle with an alternating stress above the endurance limit inflicts a measurable permanent damage.

The objective of this problem is to create a data-driven surrogate model that makes it possible to rapidly compute the damage accumulation values based on the other measurements.

2 Background Work

2.1 Machine Learning for predictive maintenance of industrial machines using IoT sensor data (2017)

The paper [1] discusses:
Data collection was done from a slitting machine. Data generated from the Slitting machine was collected using sensors and pushed to the cloud. This data was sampled per second and

collected for a period of one month. The data is stored as CSV in the system, consisting of 5 columns namely, Time Stamp, Tension, Pressure, Width and Diameter. Sliding Mean was used to replace the missing values in the data collected. The sensors send the data when the machine shows a change in state which usually is sampled per second. Hence, the data received consisted of many Null values. The first step towards preprocessing included replacing the Null values with the sliding mean values. The failure points in the data were less as compared to the data points representing good production cycles, the authors tried outlier detection using clustering and studied the outliers to gain insights. The authors used the following models:
Naive Bayes, Support Vector Machines, CART, Deep Neural Network and ARIMA: Autoregressive integrated moving average (ARIMA) model was used to predict future points [4] in the data series. Since the data showed non-stationarity, The authors have used ARIMA. The model is fitted in the dataset and further points are predicted to find if the future states which might lead to a failure.

2.2 Predictive maintenance applications for machine learning (2017)

The key contributions of this paper [2] are listed below: The authors looked at a dataset containing a subset of maintenance inspections performed by the customer across a period of 19 years, starting in 1997 and ending in 2015. For the purpose of proof of concept engagement, to provide the most business value, the authors had agreed with the customer to focus on the asset that was most serviced, a particular type of sophisticated swivel. This swivel is used in two different settings: fracking or cementing, with different observed failure risk profiles. A number of data inconsistencies had been observed in the dataset. Working with the customer's subject matter experts, the authors dealt with these issues either by removing assets where a reconciliation could not occur, or by correcting the data where possible. The authors began building a number of Predictive Models to identify assets that are going to fail within a year of the maintenance inspection. Models used included Linear regression, Logistic Regression, Neural Networks, Decision Trees, Random Forests, Gradient Boosting Machines. When evaluating the models, the authors typically randomly split the dataset into a training set and a held-out validation set. They used the model built from the training set to predict the examples in the held out validation set and compute the metrics described above. However, with small datasets, exposure to the possibly large variance in the random selection of the validation set is inevitable. For this reason, the authors employed a method called 10 fold Cross-Validation that is estimating the performance of the final model built on a full dataset by splitting the dataset into 10 folds, then building a model on 9 folds and predicting the 10th. then rotating the folds in such a way that we end up with 10 submodels where each example in the original set appeared in exactly one validation set for a submodel. At this point the authors had collected one prediction for each example, which will ensure that their computation of the performance metrics has lower variance and is more robust than one based on a random sample of the data alone. The Champion Model was an Artificial Neural Network with 3 neuron layers, which looked at all variables in the data frame. This improvement in performance is expected as Neural Networks have the ability to uncover very complex relationships between variables.

2.3 Data-Driven methods for Predictive Maintenance of Industrial Equipment: A Survey

The contributions of this paper [3] are listed below. First, a comprehensive review of the PdM is conducted from four aspects, including the definition of the PdM, significance of the data-driven PdM, specific implementation methods of the data-driven PdM, challenges in the implementation process. Second, the PdM applications of the six algorithms are compared from ML and DL perspectives, respectively, to provide graduate students, companies, and institutions with the preliminary understanding of the existing works recently published.

LR Model: LR is a well-known classification model in ML with the lowest algorithm complexity. It belongs to supervised learning; therefore, the collected data must have corresponding labels to be fed into the model. In addition, the LR model takes a linear combination of features as its input and applies a nonlinear function to conduct mapping, so that each output will fall within the range of (0, 1) and a probabilistic interpretation can be obtained.

SVM: Typically, the SVM model is used to tackle the tasks for binary classification. In the PdM of industrial equipment, SVMs have been widely applied for identifying a specific status based on the acquired signal. Moreover, because of the diversity of fault types and the ability of mapping low dimension features to hyperplanes, the SVM model can be utilized to solve multiclass tasks.

DT and RF: DT classifiers have achieved great success in various fields such as character recognition, medical diagnosis, and speech recognition. Most significantly, a DT model has the ability to decompose a complex decision-making process into a collection of simpler decisions by recursively partitioning the covariate space into subspaces, thus providing a solution that is prone to interpretation. Furthermore, RF is an ensemble learning algorithm composed of multiple DT classifiers, and the category of its output is determined jointly by these individual trees. The RF is provided with many significant advantages. For instance, it can handle high dimensional data without feature selection; trees are independent of each other during training process, and the implementation is relatively simple; in addition, the training speed is usually fast, and at the same time the generalization ability is strong enough.

ANN: Inspired by the biological neural network, ANN is designed to address nonlinear problems. It is a massively parallel computing system consisting of an extremely large number of simple processors with many interconnections. The ability of ANN models to automatically learn from examples makes them attractive and extensively applied. Moreover, compared with traditional data-driven methods, NNs have obvious advantages in addressing fuzzy data, random data, and nonlinear data. They are especially suitable for systems with a large scale, a complex structure, and unclear information.

DNN: Compared with traditional data-driven methods, DNNs can self-adaptively extract fault features to effectively represent crucial information and realize intelligent diagnosis; they can also improve identification accuracy and are extremely effective in reducing defects in manual design features.

AE: In the PdM of industrial equipment, labeled raw sensor signals are generally difficult to obtain, requiring specific and detailed experimental settings. Obviously, unsupervised feature learning is especially suitable for handling unlabeled data; it can provide a feasible solution for fault identification and RUL estimation. In addition, the AE model belongs to the type of unsupervised learning that only requires unlabeled measurement data, and it has received extensive attention and application. Third, on the basis of the above comparison, accuracy, which is the most widely used evaluation metric, is analyzed in detail, and the corresponding conclusions are drawn. In addition, some potential research directions are provided consequently. Both ML and DL can remarkably complete the PdM task. The average prediction accuracy of the reviewed literature can reach 95.06%, and the highest accuracy is 100%. With the great improvement of computational power and the rapid growth of data volume, AI algorithms and their variant models have increasingly demonstrated a superior advantage of performance. With the continuous innovation of algorithms, researchers will continue to focus on the data-driven methods in PdM applications.

3 Technical Details

3.1 Machine Maintenance Improving Machine Health

Machine maintenance is the process of performing upkeep on machinery to ensure continued working order. It may include maintenance that occurs as a part of a regular routine or prior to any sort of breakage or damage to things, like cleaning surfaces, lubricating gears, and checking for wear and tear on parts like belts. Poor maintenance strategies can reduce an organization's overall productive capacity. It can prove difficult to determine how often a machine should be taken offline to be serviced taking into account the weight borne by the risks of lost production time due to maintenance against those of a potential breakdown. Maintenance work, such as replacing and repairing parts and servicing equipment, can impose a large financial burden on the organization.

3.2 Maintenance Strategies

There are multiple strategies possible for maintenance.

- **Reactive Maintenance:** As the name implies, the machines are allowed to run to failure and any maintenance is carried post their failure. It is appealing because it offers the maximum utilization and in turn maximum production output, of the asset by utilising it to its limits.

- **Preventive Maintenance:** This involves preventing problems before they arise. This is also referred to as Planned Maintenance. It consists of maintenance tasks performed while the equipment is under normal operation to avoid unexpected breakdowns and the associated downtime and costs.

- **Predictive Maintenance:** This involves predicting problems to increase asset reliability. It directly monitors asset performance during normal operations to anticipate and predict failures.

3.3 Predictive Maintenance

Predictive maintenance has been enabled by advances in technology that connect the asset/machine to data analytics through sensor data. These days, we have manufacturing machines loaded with sensors that provide critical usage data. These sensors record a variety of values such as temperature velocity and acceleration in regular time intervals These sensors may also provide a computed damage accumulation measure, which is helpful for predictive maintenance. It directly monitors asset performance during normal operation to anticipate and predict failures. Predictive maintenance analyzes the data gathered from sensors connected to assets. This data can be used to predict when an asset will fail and in turn, allows maintenance teams to correct the issue before a failure occurs. Rather than running a part to failure, or replacing it when it still has life because of protocols, predictive maintenance helps organizations optimize their strategies by conducting maintenance only when completely necessary.

3.4 Barriers to Predictive Maintenance

- **Detection Modalities and associated datasets:** Predictive maintenance hinges on processing multiple datasets, notably those associated with different sensors and maintenance detection modalities such as vibration analysis, oil analysis, thermal imaging, acoustics and more.
- **Data Fragmentation:** Even when data from multiple sensors are available, they tend to be isolated from each other and are often represented in different formats, making their integration challenging.
- **Lack of proper analytics algorithms and tools:** The predictive analysis of large datasets requires advanced algorithms and tools beyond baseline machine learning and statistical models.
- **Complexity:** It is complex to establish than a preventive measure strategy. If the proper technology is not already established, it requires time to set up the infrastructure which involves developing equipment, technology and user adoption methods.
- **Data Collection:** Collecting the right data to enable the organization to accurately predict failures and malfunctions is crucial.

4 Proposed Architecture

There are multiple entities that we require for our model of predictive maintenance to work efficiently. The entities represent different components of the structure, which will manage all the work and data flow. The majority can be alluded to as User, Admin, Analyst, Sensors, Server, Database. The **User** of the machine will be able to place requests for the slots in which different machines can be used. Based on the availability of the machine, it will be assigned to the respective slot. If the user faces any problem then, **Admin** can help him/her to troubleshoot the problems faced. **Sensors** attached to different machines will be sending information about the machine in continuous uniform time-separated intervals. All the predictive maintenance work will be done with the help of the data sent through by the sensors, on **Servers**. All this information, whether computed or general information would be stored in a **Database**, where it would possible to access that data at any time required. **Analyst** will be the one responsible for building and enhancing the damage accumulation prediction model by retrieving the data directly from the database.The Time Series data collected so far would then be used to build and enhance damage accumulation prediction model by retrieving data directly from the database. With this predictive data, alerts are sent when maintenance is required. The Admin can then act on these alerts to halt machines and allow maintenance activities.

4.1 Prediction Approach

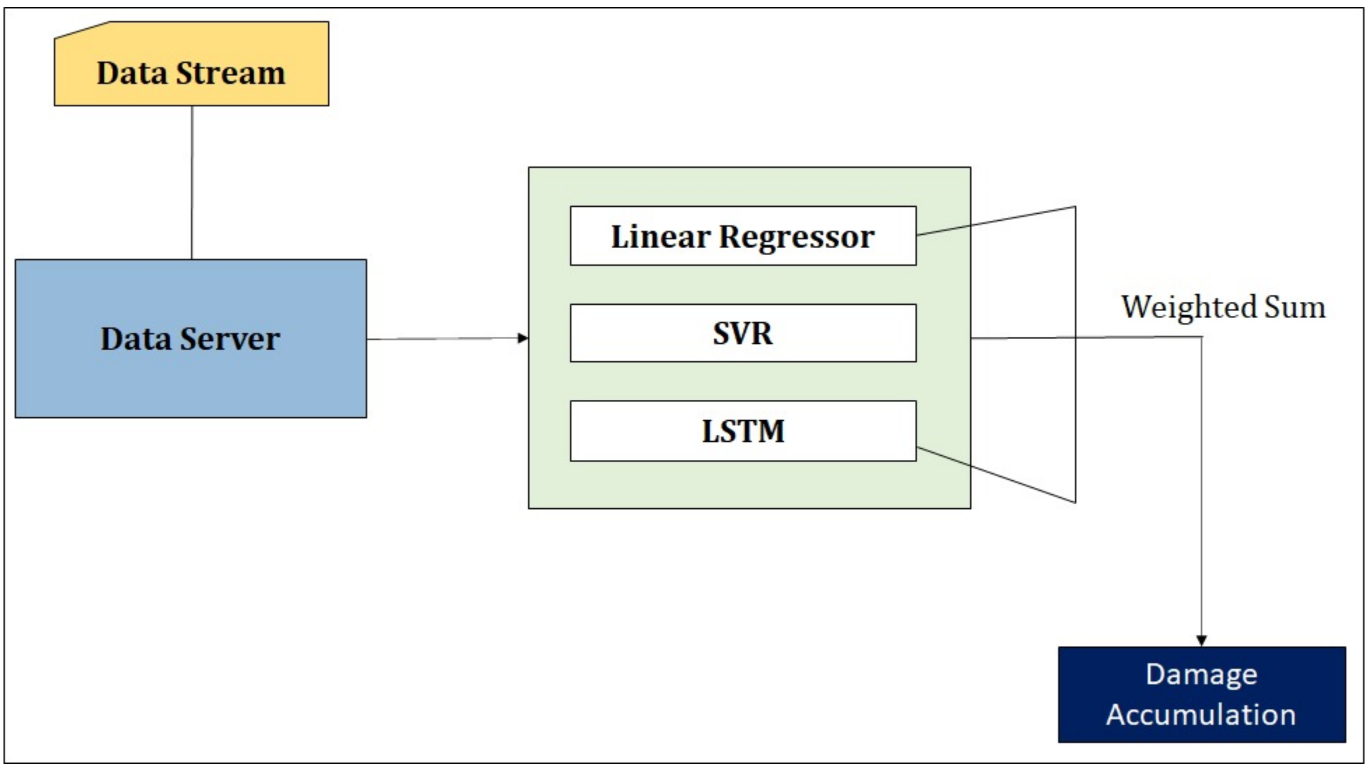

Figure 1: Proposed Model

The proposed architecture takes into account the weighted average of the Damage Accumulation DA_{model} values determined from the three models. We compute the weighted average based on the $R2$ score for Linear Regressor and SVR with the LSTM value. A higher $R2$ score implies a higher accuracy pertaining to which it's weight increases, eventually having a more significant impact on the final damage accumulation value ($DA_{predicted}$).

$$DA_{LR\,Weighted} = R2_{score} \times (DamageAccumulation_{LR}) \quad (1)$$

$$DA_{SVR\,Weighted} = R2_{score} \times (DamageAccumulation_{SVR}) \tag{2}$$

$$DA_{predicted} = \frac{DA_{LR\,Weighted} + DA_{SVR\,Weighted} + DA_{LSTM}}{3} \tag{3}$$

4.2 Approaches for effective Predictive Maintenance

4.2.1 Slot-Wise Prediction

One of the methods is to analyze and predict damage accumulation before every slot is allotted, and then if the machine is in working condition then allot the slot. This approach has some disadvantages as predicting when the user has asked for a slot, has no use, as that can be figured out by the Admin only.

4.2.2 2-Day Prediction

Analyze and predict the damage accumulation for finite number of days, so that the maintenance can be scheduled accordingly so as to minimize the reduction of working hours of the machine. This approach may also have a bottleneck when the schedule is tight for the week, and there is no time for its maintenance.

4.2.3 Chosen Approach: Weekly Prediction

Analyze and predict damage accumulation for a week in advance, which would allow us to have increased buffer time for machine repair including out of office times and weekends. There are some benefits for this approach as we can have Day-Wise analysis, Weekly average, Month-wise analysis.

4.3 SAM - Scheduling Algorithm

Given the data about different tasks required to be completed by different users(students/staff) on machines which have predictive maintenance model enabled. The data presumably consists of the duration of each task requires, and the deadlines for each of them, suggesting the urgency of it. We propose an algorithm which takes into account the slots available for the machine over the week, and the data given to it by the users, to assign slots to the respective projects. The basic implementation of the program is to first sort the data according to the deadlines, then assign higher priority to the deadlines. It might be possible that multiple of them may have the same deadline, when a weighted value is taken to give more priority to the tasks requiring more time to complete than others. It also checks whether the duration and deadlines specified are feasible or not, comparing the total hours of the tasks to the total slots available, and also comparing the total hours of similar deadline tasks to the total slots available for them.

- **High - Level Feasibility Check:** The high-level feasibility check is a boundary condition for determining if the slotting is going to be successful at all. If the total available slots is less than the total duration required for the work to be finished, a False feasibility flag is raised which terminates the execution.

$$Total\ Project\ Hours <= Available\ Slots\ Per\ Day \times Total\ Days \tag{4}$$

- **Low - Level Feasibility Check:** In the low - level feasibility check, we check for every project individually if they can meet their deadlines or not. For instance, a project which requires 30 hours of work and has a deadline within 2 days while the lab is available only for 10 hours a day will result in a false feasibility flag contrary to a project with the same duration of work requirement but a more relaxed deadline. For all projects with the same deadline, we introduce the low-level check as

$$Total\ Project\ Hours <= Available\ Slots\ Per\ Day \times Total\ Deadline\ Days \tag{5}$$

Pseudo Code: The Basic implementation of the algorithm 6

5 Experiments

5.1 Dataset Description

The dataset consists of information provided by the sensors attached to different machine collected from Bernard M. Gordon Learning Factory (Penn State). The readings from a heterogeneous set of sensors used to report metrics continuously for many different machines.

5.2 Preprocessing and Experimentation

The task at hand is to predict damage accumulation for predictive maintenance of machines, to enable a higher performance, and reduced off time. We divided the dataset into 90-10 for train and test datasets, From the 90 percent split, another 10 percent was used for validation set.

5.2.1 Support Vector Regression (SVR)

We had trained the model on varying ratios of training and test data before finding most suitable ratio. SVR involves hyper-parameters and their tuning is key to rendering a good accuracy. We have employed auto-tuning to find the best suited hyper-parameters.

5.2.2 LSTM

For the LSTM based models we used a deep and a shallow LSTM with different units in each layer. We trained all LSTM architectures for 20 epochs.

5.2.3 SAM Scheduler

We took a sample set for experimenting and testing our SAM Scheduler. We assumed 15 slots a day for one machine for five days a week. 6 projects were considered with the following duration (in hrs) for each project - $[11, 14, 10, 8, 15, 12]$ The deadlines for the projects (in days) was recorded as $[5, 3, 3, 4, 5, 2]$. Using the SAM Scheduler, we prepared a schedule (see Figure 2 with the given values so as to enable completion of the projects in the stipulated time.

	Monday	Tuesday	Wednesday	Thursday	Friday
Slot 1	P6	P6	P2	P4	P1
Slot 2	P6	P6	P2	P4	P1
Slot 3	P6	P6	P2	P4	P1
Slot 4	P6	P6	P2	P1	P5
Slot 5	P6	P6	P3	P1	P5
Slot 6	P6	P6	P3	P1	P5
Slot 7	P2	P2	P3	P1	P5
Slot 8	P2	P2	P3	P5	P5
Slot 9	P2	P2	P4	P5	P5
Slot 10	P2	P2	P4	P5	P5
Slot 11	P2	P2	P4	P5	P5
Slot 12	P3	P3	P1	P5	Empty
Slot 13	P3	P3	P1	P5	Empty
Slot 14	P3	P3	P1	P5	Empty
Slot 15	P4	P4	P1	Empty	Empty

Figure 2: Results of SAM Scheduling Algorithm

6 Results

Both the training and validation loss for the LSTM based model converge as shown in Figure 3, showing effective learning. The accuracy has not been plotted as the target variables here are float values which may give incorrect insight on the model performance. Alternatively, a rounded-off approach for determining the accuracy may be employed.

Based on the metric scores, Linear Regression provides a better result than the other two models. This, however, may change with an increase in the volume of data as in most predictive maintenance use cases, the data varies a lot upon usage and is highly non-linear.

Figure 5 depicts a high correlation between peak velocity and peak acceleration which is trivial from the laws of physics. Battery voltage shows a weak correlation with all other values and may thus be omitted while for training and predictions.

Figure 4 shows comparison for the three employed approaches Linear Regression (LR), Support Vector Regression (SVR) and LSTM with the ground truth values. The LR and SVR are more accurate in predicting the ground truth variables while the LSTM fails to predict the peak values for the samples in range (100-150) and (200-250). One possible reason for this could be the small volume of points with such high values. Correspondingly, the LSTM considers these vales to be outliers. This may be solved when we train on a larger data with more data points. If the volume for such high values increases, the LSTM architecture shall incorporate them.

Metric Scores			
Metrics	Linear Regression	SVR	LSTM
Training_Loss	-	-	0.0147
Validation_Loss	-	-	0.0031
Mean Squared Error	0.0217	0.0139	-
$R2_{score}$	0.8305	0.8912	-

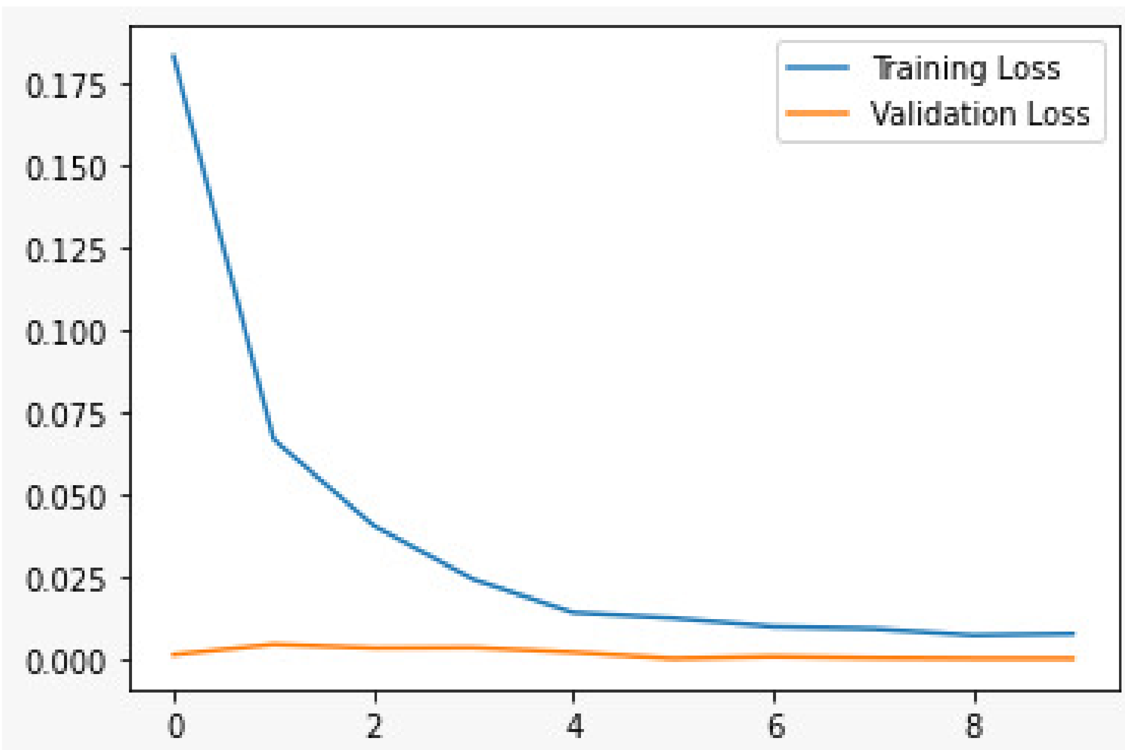

Figure 3: Training Loss and Validation Loss for LSTM

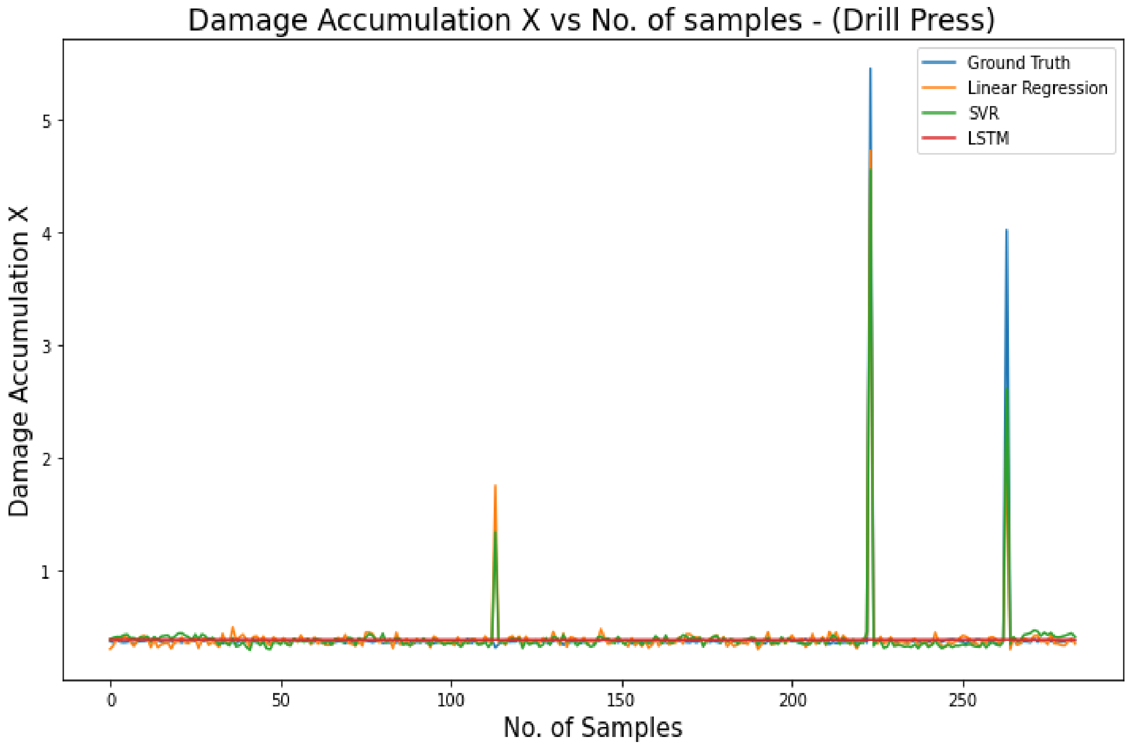

Figure 4: Prediction comparison of different Models

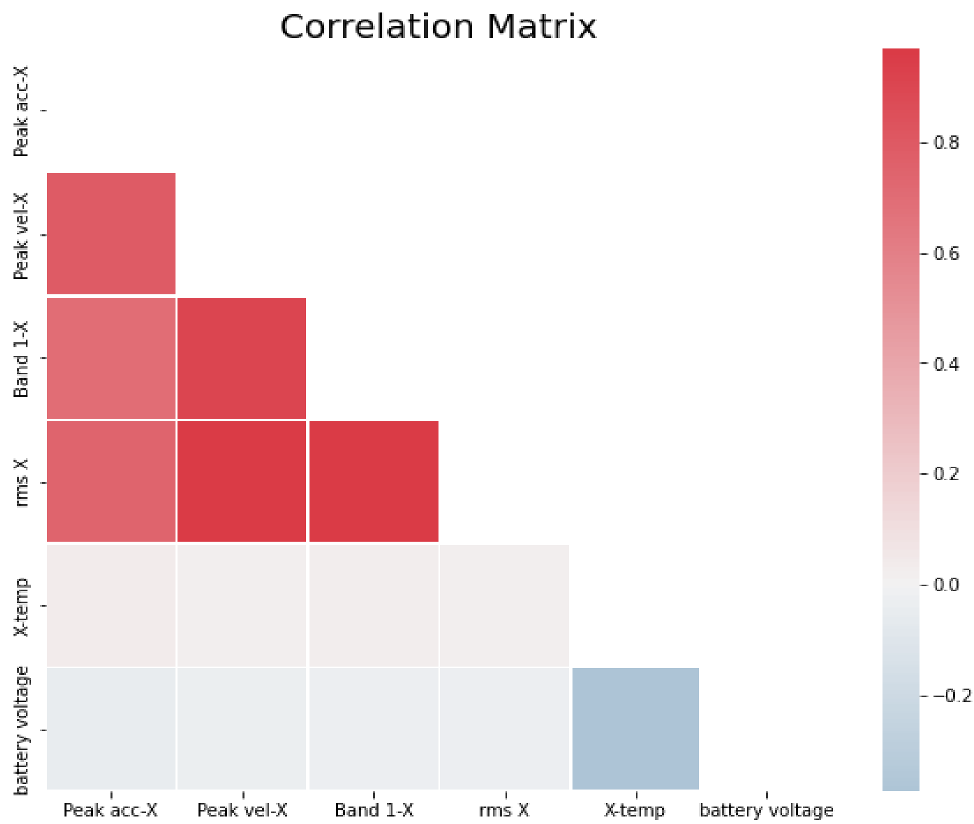

Figure 5: Correlation Matrix

7 Conclusion and Future Scope

The results are encouraging and show that the proposed architecture can be useful in the predictive maintenance use case that we have targeted. Damage accumulation value and its relation with machine breakdown is an active research problem as it varies for every machine based on it's mechanical parts and their corresponding usage. For instance, a machine with a high speed motor may break down more frequently than a low speed one for most cases. However, this may also vary depending on the material and the weight of the components for the motor. So, in such cases, it is recommended to experimentally determine the relationship between damage accumulation and machine breakdown, following which the relationship can be used to effective implementation of predictive maintenance.

The SAM - scheduler is a promising algorithm and has the potential for successful implementation in manufacturing laboratories in an academic setup. The scheduler also takes into account a buffer time to incorporate cases wherein the user may end up consuming more time for the given task or the machine breaks down during the operational hours. It is though noteworthy that the latter can occur only in the case of a Predictive Maintenance failure when the model incorrectly predicts the possible time to breakdown.

We have conducted the experiments currently on a previously collected dataset. However, in practical usage, the data would be continuously streamed via a server for regular intervals.

Algorithm 1 SAM Scheduler

```
procedure AllotSlots(params)
    Post Feasibility Checks
    Slots ← Number and type of Daily slots available
    week_slots ← Weekly Slots to be filled
    project ← Array of the form[duration, deadlines, name]
    i, start ← 0
    while i < len(project) do
        c ← project[i][1]
        Group ← grouping projects on deadlines
        hrs ← Total Hours
        tday ← 0
        while hrs > 0 do
            division ← Duration Ratio
            s ← 0
            for g = 1, 2 . . . len(group) do
                if g! = len(group) − 1 then
                    calc = ceil(project[group[g][0]]/hrs ∗ division
                    s ← s + calc, Append (group[g], calc) to temp
                else Append (group[g], division − s) to temp
                end if
            end for
            flag ← True
            for t in temp do
                x ← available − Slots[tday], e ← x
                while e < availableande < x + t[1] do
                    weeks_lots[tday][e] ← project[t[0]][2], e ← e + 1
                end while
                if x + t[1] <= available then
                    Slots[tday] ← Slots[tday] − t[1]
                    hrs ← hrs − t[1]
                    project[t[0]][0] ← project[t[0]][0] − t[1]
                else
                    Slots[tday] ← 0
                    hrs ← hrs − (available − x)
                    project[t[0]][0] ← project[t[0]][0] − (available − x)
                end if
                project[t[0][1] ← project[t[0]][1] − 1
            end for
            c ← c − 1, tday ← tday + 1
        end while
        i ← i + 1
    end while
end procedure
```

We also noted in the experimented dataset that the sensors often miss out on sending values resulting in null values. This can be corrected dynamically as and when the data is received. For this purpose, a k-rolling filter 6 can be used that uses a rolling average with a window size of k previous values to fill the missing streamed values. This window size can be experimented with for producing more accurate non null values. We may also experiment with other approaches while filling these null values. for instance, taking the average of the values for a couple of intervals on the same day, previous week. An accurate estimate of these missing values will directly affect the performance for the predictive architectures.

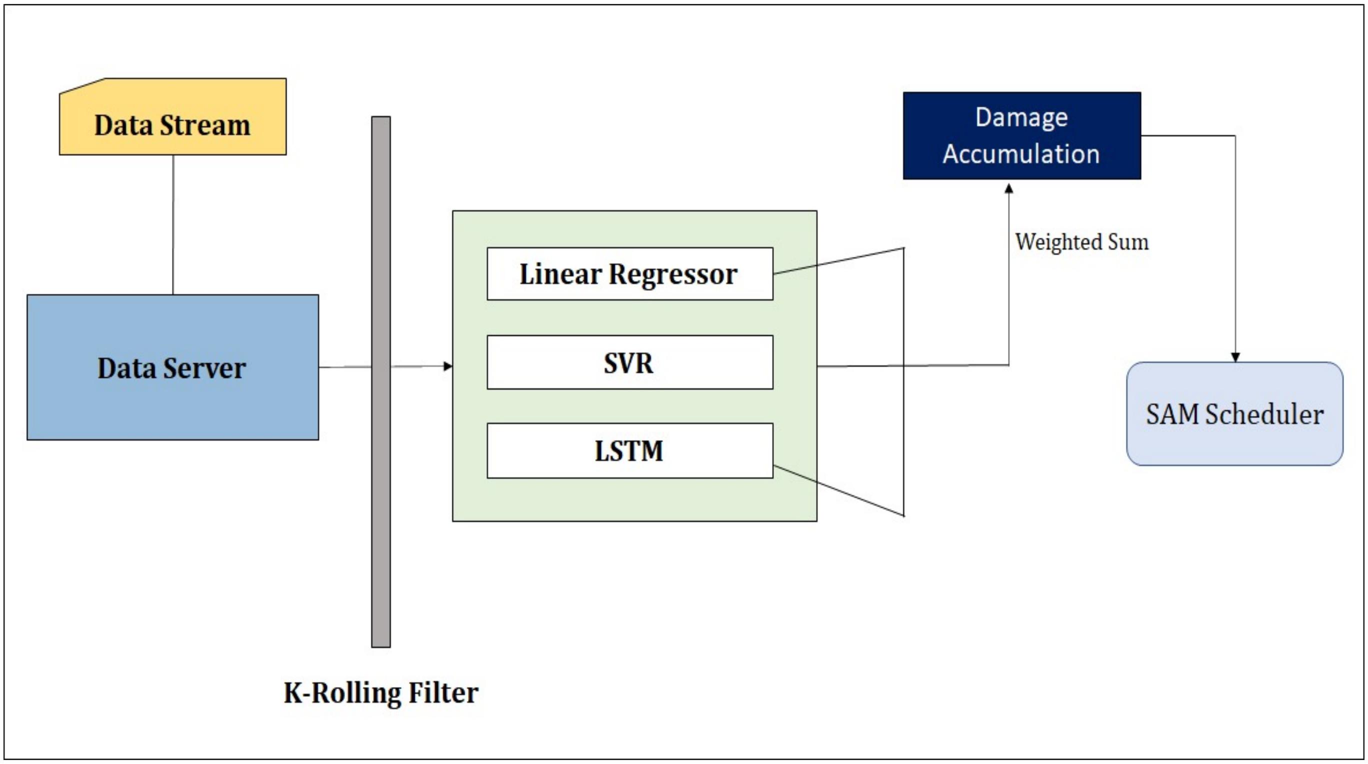

Figure 6: Possible Future Works

References

[1] A. Kanawaday and A. Sane, "Machine learning for predictive maintenance of industrial machines using IoT sensor data," 2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS), 2017, pp. 87 90, doi : 10.1109/ICSESS.

[2] B. Cline, R. S. Niculescu, D. Huffman and B.Deckel , "Predictive maintenance applications for machine learning," 2017 Annual Reliability and Maintainability Symposium (RAMS), 2017, pp. 1 7, doi : 10.1109/RAM.

[3] W. Zhang, D. Yang and H. Wang, "Data Driven Methods for Predictive Maintenance of Industrial Equipment: A Survey," in IEEE Systems Journal, vol. 13, no. 3, pp. 2213 2227, Sept. 2019, doi : 10.1109/JSYST.

[4] Z. Chen, Z. Gao, R. Yu, M. Wang and P. Sun, ”Macro-level accident fatality prediction using a combined model based on ARIMA and multivariable linear regression”, 2016 International Conference on Progress in Informatics and Computing (PIC), pp. 133-137, 2016.

[5] A. K. S. Jardine, D. Lin, and D. Banjevic, “A review on machinery diagnostics and prognostics implementing condition-based maintenance,” Mech. Syst. Signal Process., vol. 20, no. 7, pp. 1483–1510, 2006

[6] J. Lee, “Industrial Big Data, China: Mechanical Industry Press, Beijing, 2015

[7] J. Wan et al., “A manufacturing big data solution for active preventive maintenance,” IEEE Trans. Ind. Inform., vol. 13, no. 4, pp. 2039–2047, Aug. 2017

[8] P. Zhao, M. Kurihara, J. Tanaka, T. Noda, S. Chikuma and T. Suzuki, ”Advanced correlation-based anomaly detection method for predictive maintenance,” 2017 IEEE International Conference on Prognostics and Health Management (ICPHM), Dallas, TX, USA, 2017, pp. 78-83.

[9] R Zhao et al ..,,“Deep learning and its applications to machine health monitoring,” Mech Syst Signal Process vol 115 pp 213 237 2019

[10] Y Jiang and S Yin, “Recursive total principle component regression based fault detection and its application to vehicular cyber physical systems,” IEEE Trans Ind Inform vol 14 no 4 pp 1415 1423 Apr 2018

[11] H Li, K Ota, and M Dong,“Learning IoT in edge Deep learning for the Internet of Things with edge computing,” IEEE Netw vol 32 no 1 pp 96 101 Jan ../Feb 2018

[12] Vijay Rathod, Om Prakash Yadav, Ajay Rathore, Rakesh Jain,“Probabilistic Modeling of Fatigue Damage Accumulation for Reliability Prediction”, Hindawi Publishing Corporation, International Journal of Quality, Statistics, and Reliability Volume 2011 Article ID 718901 10 pages, doi: 10 1155 2011 718901

[13] Zhu, Shun Peng Huang, Hong Zhong Li, Yanfeng Liu, Yu Yang, Yuan Jian 2013 Probabilistic modeling of damage accumulation for time dependent fatigue reliability analysis of railway axle steels Proceedings of the Institution of Mechanical Engineers, Part F Journal of Rail and Rapid Transit 229.23-33.10.1177/0954409713496772

[14] Yang, Chun Hao Zidek, James Wong, Samuel 2017 Bayesian Analysis of Accumulated Damage Models in Lumber Reliability Technometrics 61.10.1080/00401706.2018.1512900

[15] J. Shimada and S. Sakajo, ”A statistical approach to reduce failure facilities based on predictive maintenance,” 2016 International Joint Conference on Neural Networks (IJCNN), Vancouver, BC, 2016, pp.5156-5160.