Using LSTM Networks to Predict Engine Condition on Large Scale Data Processing Framework

Olgun Aydin, Seren Guldamlasioglu
Big Data Research and Development Group
STM Defense Technologies Engineering and Trade Inc.
Ankara, Turkey
e-mail: olgun.aydin@stm.com.tr, sguldamlasioglu@stm.com.tr

Abstract—**As the Internet of Things technology is developing rapidly, companies have an ability to observe the health of engine components and constructed systems through collecting signals from sensors. According to output of IoT sensors, companies can build systems to predict the conditions of components. Practically the components are required to be maintained or replaced before the end of life in performing their assigned task. Predicting the life condition of a component is so crucial for industries that have intent to grow in a fast paced technological environment. Recent studies on predictive maintenance help industries to create an alert before the components are corrupted. Thanks to prediction of component failures, companies have a chance to sustain their operations efficiently while reducing their maintenance cost by repairing components in advance. Since maintenance affects production capacity and the service quality directly, optimized maintenance is the key factor for organizations to have more revenue and stay competitive in developing industrialized world. With the aid of well-designed prediction system for understanding current situation of an engine, components could be taken out of active service before malfunction occurs. With the help of inspection, effective maintenance extends component life, improves equipment availability and keeps components in a proper condition while reducing costs. Real time data collected from sensors is a great source to model component deteriorations. Markov Chain models, Survival Analysis, Optimization algorithms and several machine learning approaches have been implemented in order to model predictive maintenance. In this paper Long Short Term Memory (LSTM) networks has been performed to predict the current situation of an engine. LSTM model deals with a sequential input data. Training process of LSTM networks has been performed on large-scale data processing engine with high performance. Since huge amount of data is flowing into the predictive model, Apache Spark which is offering a distributed clustering environment has been used. The output of the LSTM network is deciding the current life condition of components and offering the alerts for components before the end of their life. The proposed model also trained and tested on an open source data that is about an engine degradation simulation provided by the Prognostics CoE at NASA Ames.**

Keywords-ANN; predictive maintenance; LSTM; apache spark; big data

I. INTRODUCTION

The system reliability is one of the most critical points for engineering operations. Failure of some parts of system could affect all of the operation. Turbine engines, power supplies and batteries are typical instances that could cause operation failure.

To avoid break down condition, some or all parts of the system should be well maintained.

In common maintenance strategies, part of a system is repaired when failure is observed [1].

To predict current condition of any system units, condition based maintenance (CBM) has been proposed. According to Jardine et al. CBM recommends actions based on the information collected from system. Main aims of CBM are avoiding unnecessary maintenance actions and recommending maintenance actions if anomaly is detected [2]. Estimation remaining useful lifetime (RUL) with high accuracy is crucial to develop effective CBM strategy. RUL could be predicted through collecting signals with sensors located on related units of system.

In 2014 Amit et al. developed artificial neural networks (ANN) to predict RUL under unknown initial wear [3]. Amit et al. proposed ANN based approach for more accurate RUL prediction of high speed milling cutters. The proposed model was constructed on time based statistical features. Sateedh et.al proposed a novel approach for RUL estimation called Meta-cognitive Regression Neural Network (McRNN) for function approximation. McRNN employs Extended Kalman Filter (EKF) to find optimal network parameters training [4].

Porotsky developed a new model to control parameter optimization based on cross validation procedure for solution of the question in the IEEE PHM 2012 Conference Challenge Competition and with their solution have been awarded “Winner from Industry”[5]. In 2013 Rodney et al proposed a time-frequency feature extraction based method for estimating RUL. The method extracts measures that quantify the complexity of time-frequency surfaces [6]. Felix et al. developed data-driven algorithm to predict RUL [7].

In the past few years large-scale data analysis have been center of attention, with data volumes in both industry and research continue to grow the processing speed of individual machines faster. Google’s MapReduce model and Hadoop pioneered an ecosystem for parallel data analysis large clusters, such as Apache’s Hive and Pig engines for SQL processing [8]. The main advantage in Apache Spark is resilient distributed dataset (RDD), users can explicitly cache an RDD and it is available to reuse in multiple MapReduce-like parallel [9].

Apache Spark, which is a fast engine for large-scale in memory data processing, is open source cluster computing framework. Apache Spark reads data items distributed over a cluster of machines and has been built for streaming, machine learning and graph processing.

Instead of traditional machine learning based models, this paper has been focused on new generation algorithms developing on rising generation technology platform for predicting the life condition of components. Deep learning algorithms are preferred as a bleeding edge and applicable approach for several areas. This research has been focused on LSTM networks that is a type of Recurrent Neural Network and running the network on Apache Spark [10].

LSTM has been applied to predict current condition of an engine by using large-scale data processing engine Spark. Hidden layers of LSTM memorize long data sequence using current input, previous input and network memory state. LSTM has four gate layers that are forget gate layer, input gate layer, candidate gate layer, output gate layer to preserve the information [11]. In this study the Python library Keras which uses Tensorflow backend and the other Python library Elephas which makes Keras available on Apache Spark were used.

This study makes available prediction of current condition of an engine by using huge size of sensor data. LSTM network running on distributed open source framework Apache Spark has been proposed for deciding on the engine life cycle.

II. PROBLEM DESCRIPTION

It could be clearly said that accurate estimation of engine condition based on sensors data has many benefits and advantages for prognosis of engine's current condition. In this paper the data, is about engine degradation simulation (C-MAPSS) provided by the Prognostics CoE at NASA Ames, has been used. Data set includes different combinations of operational conditions and fault modes in multivariate time series format [12]. Data consists three operational settings and 21 sensor measurements. Sensors are collecting data related temperature, engine pressure, fuel, coolant bleed. Details could be found in Table I.

TABLE I. DESCRIPTION OF DATASET

Operational Settings	
Settings No	**Description**
1	Altitude
2	Mach number
3	Throttle resolver angle
Sensor Measurements	
Sensor No	**Description**
1	Total temperature at fan inlet (°R)
2	Total temperature at LPC outlet (°R)
3	Total temperature at HPC outlet (°R)
4	Total temperature at LPT outlet (°R)
5	Pressure at fan inlet (psia)
6	Total pressure in bypass-duct (psia)
7	Total pressure at HPC outlet(psia)
8	Physical fan speed (rpm)
9	Physical core speed (rpm)
10	Engine pressure ratio (P50/P2)
11	Engine pressure ratio (P50/P2)
12	Ratio of fuel flow to Ps30 (pps/psi)
13	Corrected fan speed (rpm)
14	Corrected core speed (rpm)
15	Bypass Ratio
16	Burner fuel-air ratio
17	Bleed Enthalpy
18	Demanded fan speed (rpm)
19	Demanded corrected fan speed (rpm)
20	HPT coolant bleed (lbm/s)
21	LPT coolant bleed (lbm/s)

Before building a model, percentage residual life of engine R_l has been calculated as mentioned in [4] by using following equation.

$$R_l = (\text{Time to failure} - \text{Current Age}) / \text{Time to failure}$$

When R_l is equal to one, it indicates that remaining life of an engine is 100%. When R_l is equal to zero, remaining life of engine is 0%, engine is failed. R_l values were divided to four classes indicating the life condition of the components. Each class points out the one of the following life cycle conditions: Healthy, Caution, Repair, Failure as shown in Table II. Healthy state indicates that the engine is at the beginning of its life time. Caution state points out the engine which has already began to work and needed to be cared. An action has to be taken before the component falls into the repair state. Fail is the undesired state which causes the break down.

TABLE II. DESCRIPTION OF CONDITIONS

R_l Range	Condition
(0.85 - 1]	Healthy
(0.7 - 0.85]	Caution
(0.5 - 0.7]	Repair
[0 - 0.5]	Fail

Our solution is aimed to have an alert before the component falls into the repair condition. By having this alert, the component would be maintained or replaced before the system breaks down. Moreover, the solution also has a goal to observe the life condition of each component at any time.

This study approaches the problem from a different viewpoint through offering a predictive maintenance solution based on distributed clustering environment served by Apache Spark.

III. Long Short Term Memory (LSTM)

Recent studies on predictive maintenance are mostly applied with Hidden Semi Markov models to predict the remaining useful life and reliability of components. Hidden Semi Markov models commonly are expressed by failure rates that are defined as the frequency of break down of a component per hour. Hidden Semi Markov models find a probability of failure transition rates [13]. Rather than Hidden Semi Markov models, Artificial Neural Networks (ANN) are preferred for training predictive maintenance models. Features feed the input layer and feed forward neural network topology is being designed mostly. The network giving the minimum validation error is selected to represent the optimum outcome. Log sigmoid transfer function is applied while constructing ANN model. Output is normalized between 0 and 1 [14].

This study has been focused on Long Short Term Memory Networks that is a specialized implementation of Recurrent Neural Networks (RNN). LSTM has been introduced by German researchers; Sepp Hochreiter and Bergen Schmidhuber in the mid 90s in order to enable model to learn long term dependencies. LSTM has been proposed for vanishing gradient problem [15]. LSTMs have a chain of repeating modules of neural network as standard RNNs. Repeating modules in standard neural networks have the simple structures like tanh and sigmoid layers; however, LSTMs have different repeating modules comparing RNN and other type of neural networks. Instead of having a single neural network layer, LSTMs have four interacting special layers. These four layers covers the tanh and sigmoid layers as seen on Figure 1. Each layer carries an entire vector from the output of layer to inputs of the next layer.

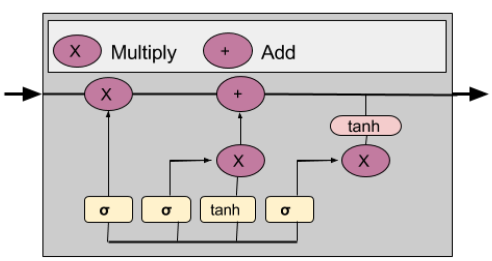

Figure 1. LSTM modules including four layers

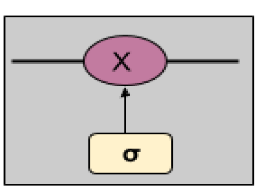

Figure 2. LSTM module including add operation

LSTMs have an ability to add information or remove information while going through gates that are a way to decide how much information to carry among the layers. Figure 2 depicts the pointwise multiplication operation which carries the information among the cells.

If all information has been carried through the gates, the value of sigmoid function takes 1. Therefore, the first step of LSTMs is deciding the amount of information carrying among the states. This decision is taken by forget gate layer which is also a sigmoid layer. If this layer takes 0 value, all information is forgotten. After deciding the forget gate layer, the next step is to decide new information will be added to next cell state. Input gate layer decides on which information will be updated on the next layer while a tanh layer creates new values will be added to the state. Moreover, the value of how much the states will be updated should be decided. Then, updated and new added candidates are combined into the state. After this process, the old state (S_{t-1}) is updated with the new state (S_t) as seen on the Figure 3. Old states are multiplied by $f_{t,}$ in order to forget things that are decided to forget. Additionally, in order to decide the amount of information that would be gained, i_t * S_t has been added to the model as seen on the Figure 3. S_t represents a tanh layer that is creating vector of new candidate values. While moving through the next states, information decided to forget is dropped from the gained information [16].

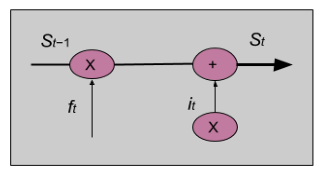

Figure 3. Updating the new state

IV. The Proposed Model

To build an efficient prediction model for learning long term dependencies, new generation one of the Deep Learning algorithms named LSTM has been applied for developing predictive maintenance model aims to predict remaining life of the engine. As a feature selection, according to calculations, Total temperature at fan inlet (°R), Pressure at fan inlet (psia), Demanded fan speed (rpm), Demanded corrected fan speed (rpm) has been dropped, because they had zero variance. Total temperature at HPC outlet (°R), Total temperature at LPT outlet (°R), Total pressure in bypass-duct (psia), Total pressure at HPC outlet(psia), Ratio of fuel flow to Ps30 (pps/psi), Bleed Enthalpy, HPT coolant bleed (lbm/s) has been dropped, because they has been highly correlated with other sensors.

Finally, Total temperature at LPC outlet (°R), Physical fan speed (rpm), Physical core speed (rpm), Engine pressure ratio (P50/P2), Engine pressure ratio (P50/P2), Corrected fan speed (rpm), Corrected core speed (rpm), Bypass Ratio, Burner fuel-air ratio, LPT coolant bleed (lbm/s) has been used as input variables for sensors.

The layer architecture shown in Figure 4 has been implemented in this study. The architecture covers 15 inputs which are operation settings 1, operation settings 2, operation settings 3, unit number, total temperature at LPC outlet (°R), physical fan speed (rpm), physical core speed (rpm), engine pressure ratio (P50/P2), engine pressure ratio (P50/P2), corrected fan speed (rpm), corrected core speed (rpm), bypass ratio, burner fuel-air ratio, LPT coolant bleed (lbm/s)) feeding into the network. The proposed model creates the classed of 4 output labels as healthy, caution, repair, fail.

Keras library of Python has been implemented on Apache Spark distributed clustering environment to build the network. Keras is one of high-level neural networks library, written in Python and capable of running on top of either TensorFlow or Theano [17]. For making all of the process distributed, another library of Python called Elephas has been used. Elephas is an extension of Keras, which allows to run distributed deep learning models at scale with Spark [18].

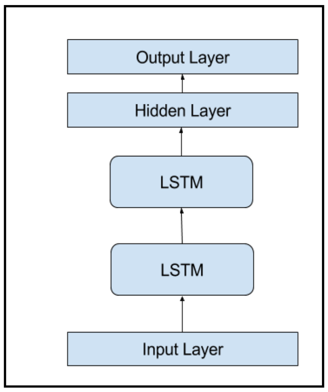

Figure 4. Proposed LSTM network layers

V. Results and Discussion

Experimental results show that current engine condition could be predicted to take necessary precautions when engine current condition is predicted as repair.

LSTM Network model has been processed by implementing Keras and Elephas libraries on distributed environment with the power of Apache Spark. In this study epoch number in training section has been decided as 200 and the batch sizes has been identified as 30. Categorical Cross Entropy has been selected as loss function and adadelta has been used to optimize the loss function.

According to Figure 5, training accuracy of RUL prediction has resulted as %85.

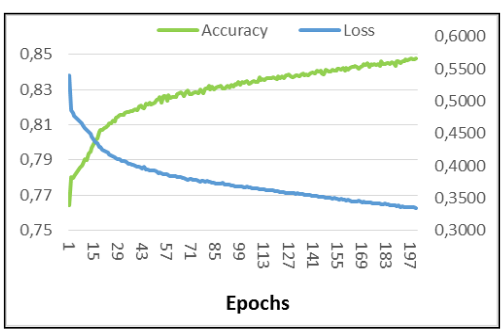

Figure 5. Training process

VI. Conclusion

This paper describes the LSTM implemented on Apache Spark offering a large-scale distributed data processing environment in order to predict the current life condition of an engine. Previous researches which aim to predict the maintenance conditions were mostly based on Hidden Markov Models or traditional Artificial Neural Networks. Instead of the common models, LSTM working on distributed environment is an edge bleeding technology to predict the current engine condition. The accuracy of the proposed model points out the reliability of the architecture which might lead industries to get an alert before a break down occurs. Therefore, companies might have a chance to reduce maintenance cost while increasing revenue and service quality.

Acknowledgment

Thanks to STM Defense Technologies Engineering and Trade Inc. for supporting us to maintain this study. STM provides system engineering, technical support, project management, technology transfer and logistics support services for TAF (Turkish Armed Forces) and SSM (Undersecretariat for Defense Industries).

References

[1] Q Zhou, J Son, S Zhou, X Mao, M Salman, Remaining “Useful life prediction of individual units subject to hard failure”, IIE Transactions Vol 46, Jun2014, pp. 1017-1030,doi: 10.1080/0740817X.2013.876126

[2] A. Jardine, D. Lin, D. Banjevic. “A review on machinery diagnostics and prognostics implementing condition-based maintenance”, Mechanical Systems and Signal Processing, vol 20, Oct 2006, pp. 1483-1510, doi: 10.1016/j.ymssp.2005.09.012

[3] A. Jain, P. Kundu, B. K. Lad, “Prediction of remaining useful life of an aircraft engine under unknown initial wear minutes”, 5th International & 26th All India Manufacturing Technology, Design and Research Conference (AIMTDR 2014), Dec 2014, pp. 494:1-494:5, doi: 10.1007/978-81-322-2352-8

[4] G. Sateesh Babu , Xiao-Li Li , S. Suresh . "Meta-cognitive Regression Neural Network for function approximation: Application to Remaining Useful Life estimation." Neural Networks (IJCNN), 2016 International Joint Conference on. IEEE, Jul 2016, pp. 4803 – 4810, doi: 10.1109/IJCNN.2016.7727831

[5] S. Porotsky. "Remaining useful life estimation for systems with non-trendability behaviour." Prognostics and Health Management (PHM), IEEE Conference on., Jun 2012, pp. 1-6, doi: 10.1109/ICPHM.2012.6299544

[6] K. Singleton, E.G. Strangas, S. Aviyente. "Time-frequency complexity based remaining useful life (RUL) estimation for bearing faults." Diagnostics for Electric Machines, Power Electronics and Drives (SDEMPED), 9th IEEE International Symposium on. IEEE, Aug 2013, pp. 600-607, doi: 10.1109/DEMPED.2013.6645776

[7] Heimes, O. Felix. "Recurrent neural networks for remaining useful life estimation." Prognostics and Health Management, PHM 2008. International Conference on. IEEE, Oct 2008, pp. 1-6, doi: 10.1109/PHM.2008.4711422

[8] M. Zaharia, M. Chowdhury, T. Das, A. Dave, J. Ma, M. Mccauley, M. Franklin, S. Shenker, I. Stoica, “Fast and interactive analytics over Hadoop data with Spark”, login, Vol 37, Aug. 2012, pp. 45-51.

[9] M. Zaharia, M. Chowdhury, M. Franklin, S. Shenker, I. Stoica, “Spark: cluster computing with working sets.”,*HotCloud*, vol.10, pp. 10-10.

[10] L. Liao , H. Ahn, “Combining Deep Learning and Survival Analysis for Asset” , 2016

[11] E. Nardo, “Distributed implementation of a LSTM on Spark and Tensorflow”, 2016

[12] A. Saxena, K. Goebel, "Turbofan Engine Degradation Simulation Data Set", NASA Ames Prognostics Data Repository (http://ti.arc.nasa.gov/project/prognostic-data-repository), NASA Ames Research Center, Moffett Field, CA, 2008

[13] M. Abbas, O. H. I. Mohammad, N. A. Omer. "Development of predictive markov-chain conditionbased tractor failure analysis algorithm." *Research Journal of Agriculture and Biological Sciences* vol. 7, pp. 52-67, 2011

[14] A. K. Mahamad, S. Saon, T.. Hiyama. "Predicting remaining useful life of rotating machinery based artificial neural network." *Computers & Mathematics with Applications,* vol.60, pp. 1078-1087, 2015

[15] K. Greff, R. Srivastava, J. Koutnik, B. R. Steunebrink, J. Schmidhuber, "LSTM: A search space odyssey." arXiv preprint arXiv:1503.04069, March 2015.

[16] D. Hristovski, B.Peterlin, J. Mitchell, M. H. Susanne, "Using literature-based discovery to identify disease candidate genes." *International journal of medical informatics*, vol 74, pp. 289-298, March 2005.

[17] Keras: Deep Learning library for Theano and TensorFlow, https://keras.io/

[18] Elephas: Distributed Deep Learning with Keras & Spark, https://github.com/maxpumperla/elephas