Smart Science

ISSN: (Print) 2308-0477 (Online) Journal homepage: https://www.tandfonline.com/loi/tsma20

Edge-Based RNN Anomaly Detection Platform in Machine Tools

Chia-Yu Lin, Chih-Ping Weng, Li-Chun Wang, Hong-Han Shuai & Wen-Peng Tseng

To cite this article: Chia-Yu Lin, Chih-Ping Weng, Li-Chun Wang, Hong-Han Shuai & Wen-Peng Tseng (2019): Edge-Based RNN Anomaly Detection Platform in Machine Tools, Smart Science, DOI: 10.1080/23080477.2019.1578921

To link to this article: https://doi.org/10.1080/23080477.2019.1578921

Published online: 20 Feb 2019.

SMART SCIENCE
https://doi.org/10.1080/23080477.2019.1578921

Taylor & Francis
Taylor & Francis Group

ARTICLE

Edge-Based RNN Anomaly Detection Platform in Machine Tools

Chia-Yu Lin[a], Chih-Ping Weng[b], Li-Chun Wang[a], Hong-Han Shuai[b] and Wen-Peng Tseng[c]

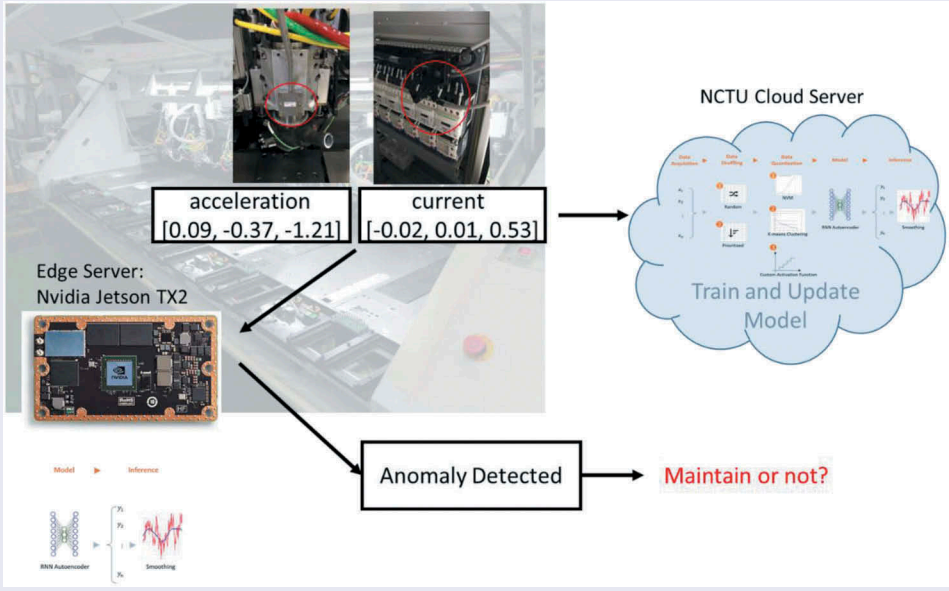
[a]Department of Electrical and Computer Engineering, National Chiao Tung University, Hsinchu City, Taiwan; [b]Department of Computer Science, National Chiao Tung University, Hsinchu City, Taiwan; [c]Tongtai Machine & Tool Co. Ltd, Kaohsiung City, Taiwan

ABSTRACT

With the rapid advances in machine learning algorithms and sensing technologies, machine prognostics and health management (PHM) via data-driven approaches has become a trend in sophisticated machine tool industry. The run-to-failure data are necessary for data-driven approaches. However, the average life of the machine is two to three years, the time of collecting data is extended. It is a big challenge to collect run-to-failure data and build a PHM model. Therefore, we propose an Edge-based RNN Anomaly Detection Platform (ERADP). ERADP builds the model based on healthy data and notify anomalies two hours in advance. The true alarm rate is up to 100%. Besides, ERADP can accelerate the training time almost 120 times faster than the traditional model.

ARTICLE HISTORY
Received 8 November 2018
Accepted 2 February 2019

KEYWORDS
Prognostics and health management (PHM); anomaly detection (AD); recurrent neural network (RNN)



We propose an Edge-based RNN Anomaly Detection Platform (ERADP) to solve the data-imbalance issue and demonstrate detect anomalies in real time for machinery industry. ERADP can make the true alarm rate up to 100% and speed up model training almost 120 times faster. Besides, we cooperate with TongTai, which is the biggest machine tool company in Taiwan. Equipped ERADP with machine tools, the cost of repairing and failure products can be intensively decreased. The price of machine tools can increase by 6%. The revenue of the machinery industry can increase by about 0.27 billion US dollars. ERADP can really make a significant impact on the machinery industry.

1. Introduction

With the rapid development of computer technologies in recent years, the industry has undergone a revolution in productivity, business models, and innovation. The first three industrial revolutions focused on the improvement of material, machine, methods, measurement, and maintenance, which is called 5M. Industry 4.0 proposed "Modeling," which is the sixth M.

The modeling technology of Industry 4.0 can predict anomalies of the machines in advance. Workers no longer need to monitor the machines. The human resources can be brought into product strategy planning and productivity. Besides, the mean time to repair (MTTR) of the machines is about one month. The longer

CONTACT Li-Chun Wang lichun@g2.nctu.edu.tw Department of Electrical and Computer Engineering, National Chiao Tung University, Taiwan

the machine tool is repaired, the more expensive it will cost. Furthermore, the deficient products will be dumped. More seriously, a rolling component failure may cause industry safety issues, such as the engine failure of a helicopter. If the machine failure can be predicted in advance, the order of repairing material and production can be rescheduled. The repairing cost of machines can be intensively decreased.

Anomaly detection (AD) is proposed to detect or even predict whether a machine will be in an abnormal state. When a machine is in an abnormal state, a set of representative single points can be captured. However, most of the anomaly detection models only consider the case that an anomaly occurs individually or separately [1–3], which cause high false alarm rate. Anomaly detection models should have the ability to remember the previous data, and to represent the relationship between the previous events and the current event [4]. Recurrent neural network (RNN)is widely used for analyzing the time-series data, thereby achieving the goal of the AD for machines.

Run-to-failure data is necessary for developing AD model. Since the average life of the machine is two to three years, the collection time of data is lengthened. It is a big challenge to collect run-to-failure data and build the prediction model for PHM. In this paper, we adopt the autoencoder of RNN to build the healthy model of machines. The new data are compared with the healthy model. If they are not matched, an anomaly is going to occur. The lack of run-to-failure data can be solved. Besides, to detect anomalies in real time, we build RNN anomaly detection model in NVDIA Jetson TX2 as an edge server. From the experiments, the proposed model can notify anomaly two hours in advance, and the true alarm rate is up to 100%. The training time of the proposed model is almost 120 times faster than the traditional model. The main contribution of this paper is demonstrating an Edge-based RNN anomaly detection platform (ERADP) to detect anomalies in real time for machinery industry.

We cooperate with TongTai Inc., which is the biggest machine tool company in Taiwan. Equipped ERADP with machine tools, the cost of repairing and failure products can be intensively decreased. The price of machine tools can increase by 6%. ERADP is not only for TongTai, but also for the manufacturing industry. The annual production of machine tools in Taiwan is about 0.4 million units. If there are 10% machine tools equipped with ERADP, the revenue of the machinery industry can increase by about 0.27 billion US dollars. ERADP can really make a significant impact on the machinery industry.

The rest of this paper is organized as follows. Section 2 introduces anomaly detection-based prognostics techniques. Section 3 presents the proposed Edge-based RNN anomaly detection platform. The experiments and numerical results are demonstrated in Section 4. Finally, we give our concluding remarks in Section 5.

2. Related Work

Prognostics and health management (PHM) have gained much attention in recent years. The goal of PHM is to maintain the operation of assets and maximize the utilization with minimal cost. PHM is so complicated that we cannot only rely on human experience. Data-driven approaches are more promising than human experience. As presented in [5], data-driven PHM can be applied in industrial domains such as (1) anomaly detection for aircraft engines and (2) RUL-driven ranking of locomotives in a fleet. Two popular data-driven PHM approaches are: (1) predicting the remaining useful life (RUL) of a machine and (2) applying anomaly detection (AD) on a machine. RUL-based prognostics is to use collected data like vibration signals to predict the RUL of a machine. When the predicted RUL is below a predefined maintenance threshold, the machine is regarded as in failure state. To accurately predict RUL of a machine is very difficult. Moreover, the maintenance threshold may be different depending on different working condition of a machine. For example [6], proposed a complicated HI-based RUL prediction algorithm to improve the reliability and availability of a machine. Although the prediction performance is improved, the maintenance threshold still depends on manual configuration [7]. Solved the range of features and the failure threshold determination issues in the prediction process of RUL. In addition, run-to-failure data is necessary to train RUL-based models. During the process of collecting run-to-failure data, a machine may face the situation of abnormality, such as impulse signals, before it fails. Therefore, AD-based approaches, which aim to detect anomalies in real time, are more suitable for machine tool industry compared to RUL approaches.

In the machine tool industry, most data are labeled as the normal state and only a few data are labeled as the abnormal state. This is called data imbalance issue. In [8], a soft-ensemble and threshold-moving method was proposed to solve data imbalance issue in cost-sensitive neural networks. Cost-sensitive learning was a popular method to solve class imbalance issue in classification problem. Cost-sensitive learning modified the cost functions to consider misclassification. However, in machinery industry, the new sensor data are continuously generated, the cost function has to be updated based on the new data. The cost of updating model is too high.

Generally, machine fault diagnosis approaches are classified into five groups [9], namely, probabilistic novelty detection, distance- based novelty detection, reconstruction- based novelty detection, domain-based novelty detection, and information- theoretic novelty detection. Among them, reconstruction-based novelty detection takes the imbalanced data issue into account [10]. Compared two reconstruction-based AD techniques, one is the simple curve fitting approach and the other is the NN approach [11]. Compared three machine learning approaches for anomaly detection (AD), including support vector machine regression (SVMR), multilayer artificial neural network (ANN) model, and Gaussian process regression (GPR). The result found that NN approaches can define more complex models and NN has the lowest error on average. Besides, NN is more useful to model data points that are not correlated. Among different types of NNs, RNN is widely used for time-series data such as vibration signals. On the contrary, the long short term memory anomaly detection (LSTM-AD) approach proposed by [12] and the encoder-decoder anomaly detecion (EncDec-AD) scheme proposed by [13] have already shown that RNN is a vi- able option to model time series behaviors. Although both LSTM-AD and EncDec-AD yielded promising results on four small volume of datasets, the performance of training model is not discussed. However, in the scenario of anomaly detection, real-time detection is necessary. Therefore, we propose a Edge-based RNN anomaly detection platform to solve data imbalance issue, increase the accuracy of anomaly detection and decrease the training time.

3. Edge-Based RNN Anomaly Detection Platform

Figure 1 is the system architecture of Edge-based RNN anomaly detection platform (ERADP). Sensors, RNN anomaly detection model on cloud servers and inference on edge servers are three critical elements in the platform. Sensors are installed on machine tools to collect the status of machine tools. Sensor data are sent to cloud servers. We train the model on the cloud servers based on the sensor data. After the model is built, the model is deployed to edge servers. New data are sent to edge servers. Edge servers can detect whether there is an anomaly or not.

3.1. Sensors

To collect the operation status of machine tools, TongTai Inc. install accelerometers, cur- rent sensors and temperature sensors on the machine tools. In many cases of PHM for machine tools, we find that vibration signals can fully represent the behaviors of machine tools [6,7,14–19]. Therefore, we choose vibration signals from accelerometers to decrease the volume of collected data and identify the characteristics of the machine tools. Raw vibration data are collected, cleaned, extracted, and piped into the cloud servers. We obtain the features of raw data from Fast Fourier Transform (FFT) to make model performance better, as shown in Figure 2. Furthermore, min-max-normalization is applied to normalize the value of features within zero to one. In the normalization process, the training process speeds up and the reconstruction errors can be measured under a uniform scale.

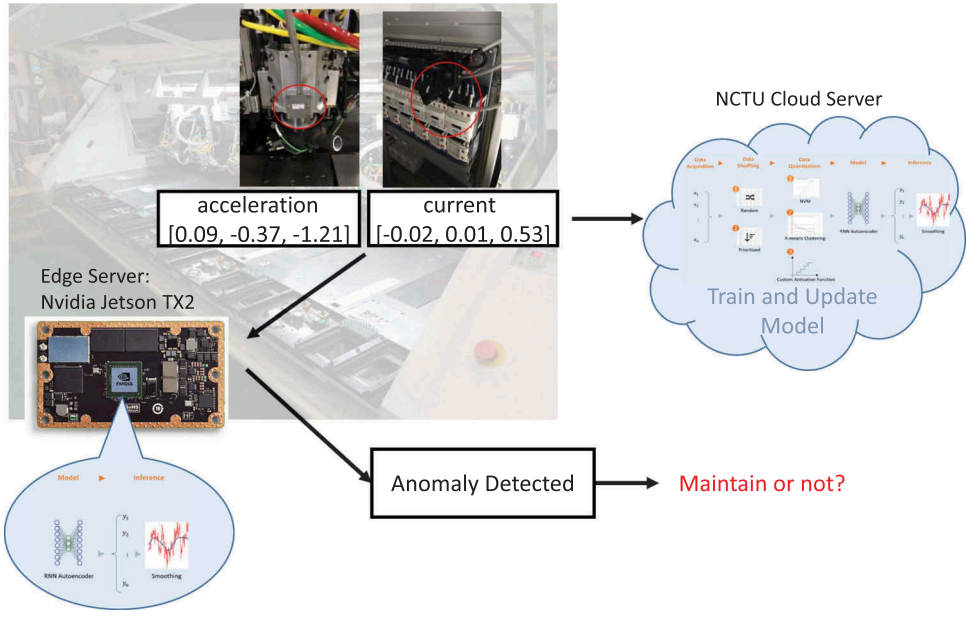


Figure 1. The system architecture of edge-based RNN anomaly detection platform.

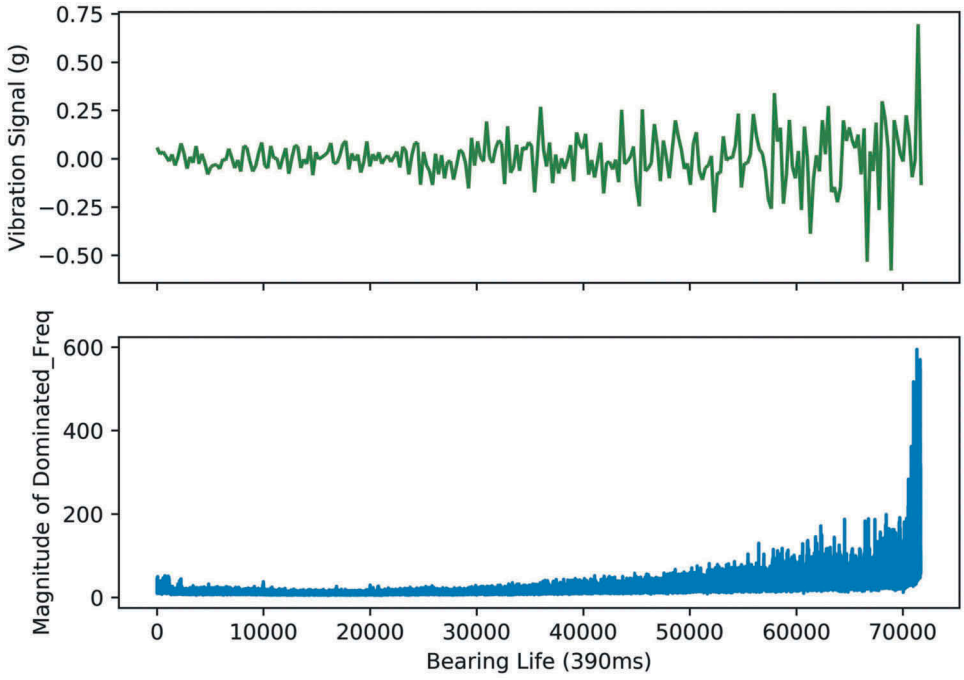


Figure 2. The raw vibration data transform to frequency domain.

3.2. RNN Anomaly Detection Model on Cloud Servers

We train a RNN anomaly detection model based on the sensor data in the cloud servers. Since lack of run-to-failure data problem, we build reconstruction-based novelty detection model by RNN. In other words, we create a healthy model of machine tools. The new data are compared with the healthy model. If they are not matched, the anomaly is going to occur.

RNN is widely used for time-series data such as temperatures, currents, or vibrations. As mentioned in [12,13], the autoencoder architecture of RNN is suitable for anomaly detection and can overcome the data imbalance issue.

An autoencoder is a type of neural network, which is composed of an encoder network and a decoder network. In the encoder network, an input vector $\boldsymbol{i}$ of length m is passed into several encoder layers Enc_i and encoded into a feature vector $\boldsymbol{f}$. In the decoder network, the feature vector $\boldsymbol{f}$ is passed through several decoder layers Dec_i and decoded into an output vector $\boldsymbol{o}$ of length n, as follows:

$$f = Enc_p(Enc_{p-1}(\ldots Enc_2(Enc_1(i)))) \tag{1}$$

$$o = Dec_q(Dec_{q-1}(\ldots Dec_2(Dec_1(f)))) \tag{2}$$

An autoencoder is shown in Figure 3, consisting of two layers of the encoders, two layers of decoders, and two-dimension latent vectors. Based on [12] and [13],

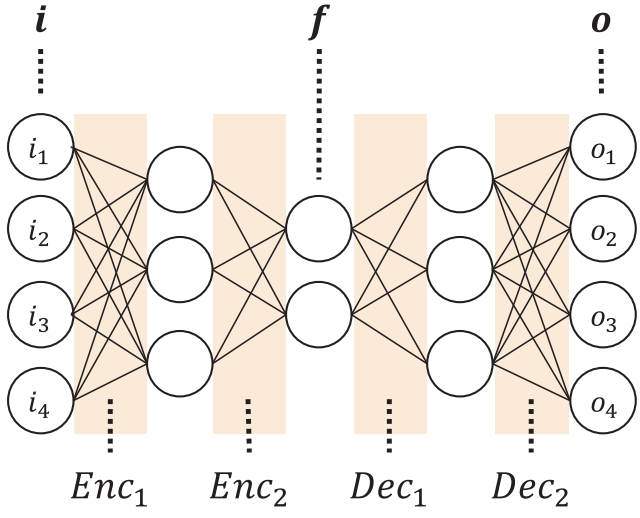


Figure 3. An illustration of autoencoder.

we train a health model by RNN based on the normal data. The training target is to minimize the Euclidean distance e between $\boldsymbol{i}$ and $\boldsymbol{o}$, where the vector length m and n are the same. During the training, the model will learn how to reconstruct the input data.

$$e = \sqrt{\frac{1}{n}\sum_{j=1}^{n}(o_j - i_j)^2} \tag{3}$$

We adopt the sequence to sequence type [20] to build the autoencoder network. Both encoder and decoder networks are composed of two-layered LSTM

cells with dropout rate 10% and time step size 32 [21]. The output size of all cells is 64. The size of the embedding vector for each symbol is 128. Models are trained with a fixed learning rate of 0.001 in 500 epochs.

3.3. Inference on Edge Servers

After the model is trained, we deploy the model to edge servers to decide whether there is an anomaly in real time. We adopt NVDIA Jetson TX2 as an edge server, as shown in Figure 4 New data are fed into a sufficiently trained autoencoder model on edge servers, and the reconstruction error e will be evaluated. To infer whether a machine tool is broken, new data are fed into a sufficiently trained autoencoder model, and the reconstruction error e will be evaluated. If the reconstruction error is smaller than a predefined anomaly threshold T, the new data will be categorized as usual. Otherwise, it will be categorized as anomalous.

The anomaly threshold T is derived from the following process. During the training, reconstruction errors $e_n^{(t)}$ and $e_a^{(t)}$ are evaluated at epoch t, where $e_n^{(t)}$ is from normal data and $e_a^{(t)}$ is from anomalous data. Then, the specific epoch t' is picked when $e_a^{(t'')}$is minimal. Finally, the reconstruction error of normal data at epoch t' is regarded as T.

$$t' = arg_t \min e_a^{(t)} \tag{4}$$

$$\mathrm{T} = e_a^{(t')} \tag{5}$$

4. Experiment

In this section, we give some numerical results to demonstrate the inference accuracy and the training time of the proposed model.

Figure 4. NVIDIA Jetson TX2.

4.1. Data Preparation

Since the dataset of TongTai Inc. is confidential, we adopt the dataset of IEEE Prognostics and Health Management (PHM) Data Challenge in 2012 [22] to evaluate ERADP. The dataset include six training sets and 11 testing sets. Six training sets are adopted to train RNN anomaly detection model on cloud servers. After the model training is trained, the well trained model is moved to edge server to detect the anomaly in real time. Horizontal and vertical vibration signals and temperature values are collected from sensors deployed on rolling bearings in these dataset. The vibration signals are collected with frequency 25.6 kHz, and temperature data are sampled with frequency 10 Hz.

As mentioned before, since vibration signals can represent machine behaviors [6,7,14–19], we only take vibration signals into account to decrease the volume of data and increase the efficiency of model training. To capture more data feature, the raw vibration signals are transformed into frequency domain by FFT and the dominated frequencies are selected as training feature. Min-max normalization is applied to training and testing data for improving training speed and providing a uniform measurement scale for inference.

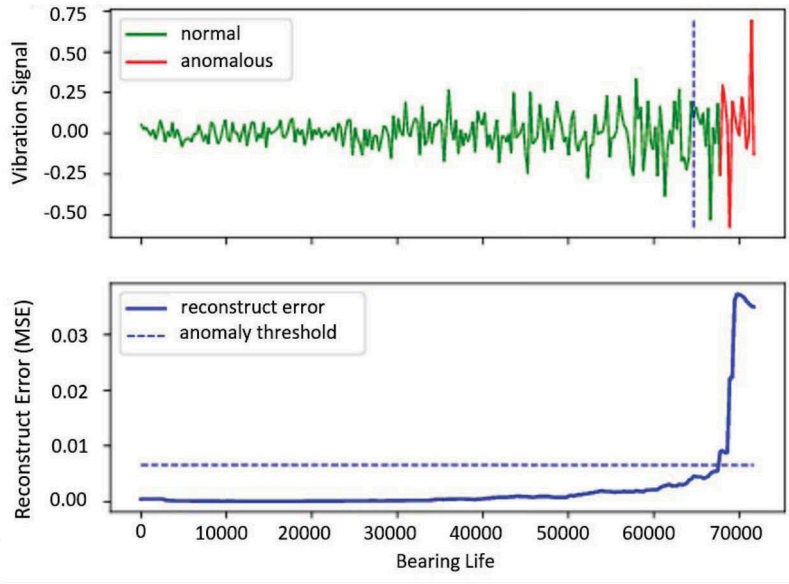
Labeling data is necessary for training AD-based models by supervised learning. We label the top 90% data of the whole machine life as normal data and the remaining 10% data as anomalous data. The normal data are the input of designed RNN autoencoder models.

4.2. Numerical Results and Observation

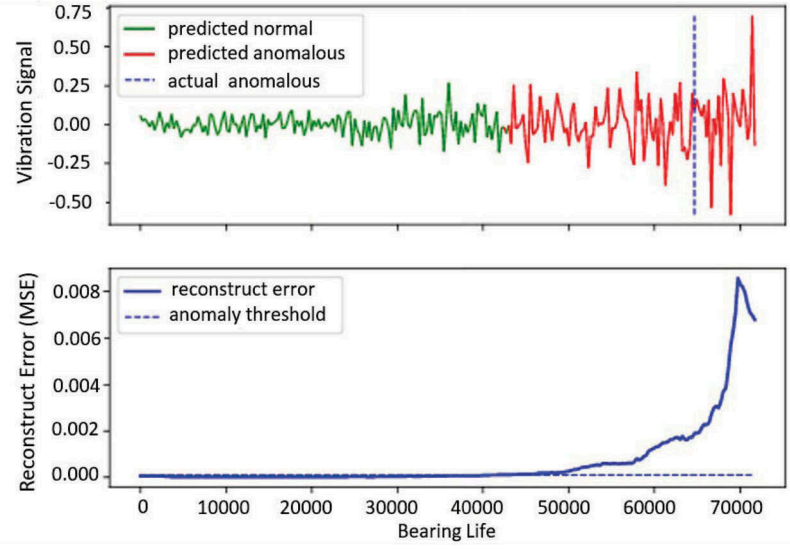
4.2.1. Inference Result

The 11 testing sets of the dataset is adopted to evaluate the performance of inference. Mean squared error (MSE) in Eq. (3) is used as performance metrics. Figure 5 is the inference result of one of the testing set. In Figure 5, the upper part shows the raw vibration signals and the lower part shows the trend of reconstruction error to demonstrate when ERADP can detect anomaly. Both parts share the same timeline and the time unit is 390 ms. The raw vibration signals are marked in red color when reconstruction error is higher than the anomaly threshold. The vertical dotted line represents the time that real anomaly happened. The reconstruction errors are averaged with window size 15 to smooth unstable peaks and improve the inference accuracy. We repeat the experiment 40 times to get the average.

The inference result of traditional RNN regression model is shown in Figure 5(a), and the result of the



(a) Traditional model.



(b) The proposed model.

Figure 5. Comparison of inference results.

proposed model is shown in Figure 5(b). We can observe that the proposed model can achieve 100% true alarm rate and the anomalies are predicted between 148.64 min and 152.72 min before anomalies happen. The average time of detecting anomaly is 150.68 min. The standard error is 1.02 min. Therefore, the confidence of the proposed model is 95%. On the contrary, the reconstruction error of the traditional model increases steeply and the anomalies are detected after they have already happened. It implies that the proposed model can identify and predict anomalies as well, which is of importance in the machine tool industry.

4.2.2. Training Performance

As shown in Figure 6, EncDec-AD takes 185 epochs about 17,880 s to train a model that can identify anomalies. Otherwise, the proposed model takes only one epoch about 150 s to finish model training. The proposed model speeds up model training about 120 times faster.

4.2.3. Inference Benchmark

The proposed model is not only lightweight but also can identify the status of machine tools in real time. The storage size of the proposed model is only 18.1 MB. It is suitable to be deployed on an embedded system or the edge server of machine tools. The

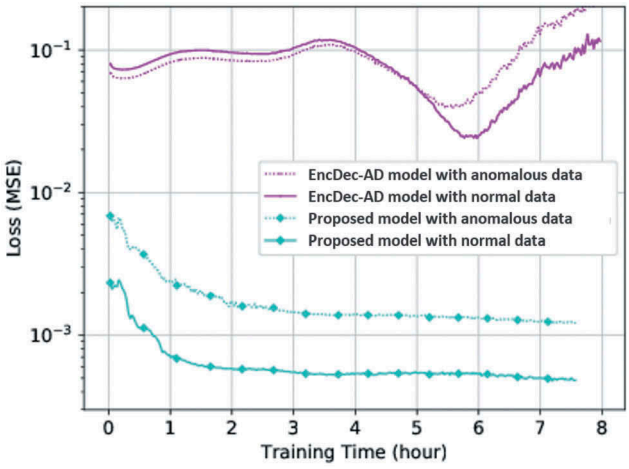


Figure 6. Comparison of training performance.

inference time of different devices are measured, as shown in Table 1. Two PC-level CPUs, two general-purpose GPUs and one embedded GPU from NVIDIA Jetson TX2 module are tested. The Jetson TX2 is tested under Max-N mode, which is of full performance. In this benchmark experiment, dominated frequencies of vibration signals with batch size 128 are tested.

The inference process are divided into three phases. In phase I, the trained model is loaded from disk to memory and constructed according to the stored meta-data such as model weightings and hyperparameters. Once the model is built, it is ready for inference. In phase II, a batch of sequential sensing data are fed into the model for initial interference. Phase III is similar to phase II, while some parameters are cached. Therefore, the inference process in phase III takes less time than that in phase II.

From Table 1, Phase I seems to be a bottleneck. However, the computational time of phase I and phase II are one-time overhead. After the initial inference, all the remaining inferences are in phase III and will not take unreasonably long time. Unlike the expectation in most cases, the results of phase III show that CPUs take less time than GPUs in a significant manner. This is because the recurrent structure of RNN cannot be computed in parallel. Although ERADP does not benefit from the powerful parallel computing capability of GPU, ERADP only takes about 1 s to detect anomaly. The detection result can be replied to user in real time.

Table 1. Inference benchmark of ERADP on different devices.

Device	Load model	Inference (first batch)	Inference (remaining single batch)
i7-7700K CPU	33.22 s	0.98 s	0.15 s
i7-8700 CPU	9.40 s	1.03 s	0.15 s
GTX 1060 GPU	35.50 s	1.32 s	0.23 s
GTX 1080Ti GPU	33.54 s	1.27 s	0.25 s
Jetson GPU	170.63 s	5.53 s	1.03 s

5. Conclusions

In this work, we propose an Edge-based RNN anomaly detection platform (ERADP). We adopt the reconstruct-based method to construct a healthy model of machines and solve the lack of run-to-failure data issue. From the experiments, the proposed model can achieve 100% true alarm rate in anomaly detection and accelerate model training up to 120 times. Besides, ERADP can help TongTai Inc. and manufacturing companies easily build maintenance models. The repairing cost can be saved and the production can be increased. The price of machine tools can grow about 6% after ERADP is equipped with machine tools. The revenue of the machinery industry can increase by about 0.27 billion US dollars. The competitiveness of the manufacturing industry can also be intensively increased.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

This work is supported by Minstry of Science and Technology, Taiwan, under the project 106-2634-F-009-002-CC2.

ORCID

Chia-Yu Lin http://orcid.org/0000-0002-5106-7286
Li-Chun Wang http://orcid.org/0000-0002-7883-6217
Hong-Han Shuai http://orcid.org/0000-0003-2216-077X

References

[1] Chmielewski A, Wierzchon ST. V-detector algorithm with tree-based structures. Proceedings of the International Multiconference on Computer Science and Information Technology, Wisła (Poland); 2006. p. 9–14.
[2] Hawkins S, He H, Williams G, et al. Outlier detection using replicator neural networks. International Conference on Data Warehousing and Knowledge Discovery. Springer; 2002. p.170-180.
[3] Salama MA, Eid HF, Ramadan RA, et al. Hybrid intelligent intrusion detection scheme. In: Soft computing in industrial applications. New York, NY, USA: Springer; 2011. vol.96, p. 293–303.

[4] Bontemps L, McDermott J, Le-Khac NA, et al. Collective anomaly detection based on long short-term memory recurrent neural networks. International Conference on Future Data and Security Engineering. Can Tho City, Vietnam, Springer; 2016. p.141-152.
[5] Bonissone PP. Machine learning applications. In: Springer handbook of computational intelligence. Springer; 2015. p. 783–821.
[6] Yang F, Habibullah MS, Zhang T, et al. Health index-based prognostics for remaining useful life predictions in electrical machines. IEEE Trans Ind Electron. 2016;63(4):2633–2644.
[7] Guo L, Li N, Jia F, et al. A recurrent neural network based health indicator for remaining useful life prediction of bearings. Elsevier Neurocomput. 2017;240:98–109.
[8] Zhou ZH, Liu XY. Training cost-sensitive neural networks with methods addressing the class imbalance problem. IEEE Trans Knowledge Data Eng. 2006;18 (1):63–77.
[9] Pimentel MA, Clifton DA, Clifton L, et al. A review of novelty detection. Elsevier Signal Process. 2014;99:215–249.
[10] Schlechtingen M, Santos IF. Comparative analysis of neural network and regression based condition monitoring approaches for wind turbine fault detection. Elsevier Mech Syst Signal Process. 2011;25 (5):1849–1875.
[11] Elforjani M, Shanbr S. Prognosis of bearing acoustic emission signals using supervised machine learning. IEEE Trans Ind Electron. 2018;65(7):5864–5871.
[12] Malhotra P, Vig L, Shroff G, et al. Long short term memory networks for anomaly detection in time series. In: Proceedings. European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN). Bruges, Belium; 2015. p. 89.
[13] Malhotra P, Ramakrishnan A, Anand G, et al. LSTM-based encoder-decoder for multi- sensor anomaly detection. arXiv preprint arXiv:160700148. 2016.
[14] Sardana D, Bhatnagar R, Pavel R, et al. Data driven predictive analytics for a spindle's health. In: IEEE International Conference on Big Data. Santa Clara, CA, USA; 2015. p. 1378–1387.
[15] Si XS, Wang W, Hu CH, et al. Remaining useful life estimation–a review on the statistical data driven approaches. Eur J Oper Res. 2011;213(1):1–14.
[16] Jin X, Sun Y, Que Z, et al. Anomaly detection and fault prognosis for bearings. IEEE Trans. on Instrum Meas. 2016;65(9):2046–2054.
[17] Dong L, Shulin L, Zhang H. A method of anomaly detection and fault diagnosis with online adaptive learning under small training samples. Elsevier Pattern Recogn. 2017;64:374–385.
[18] Sakthivel N, Nair BB, Elangovan M, et al. Comparison of dimensionality reduction tech- niques for the fault diagnosis of mono block centrifugal pump using vibration signals. Elsevier Eng Sci Technol Int J. 2014;17 (1):30–38.
[19] Zhang W, Peng G, Li C. Bearings fault diagnosis based on convolutional neural networks with 2-D representation of vibration signals as input. International Conference on Mechatronics and Mechanical Engineering. Elazığ, Turkey.; 2016.
[20] Cho K, Van Merri¨Enboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:14061078. 2014.
[21] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput. 1997;9(8):1735–1780.
[22] FEMTO-ST. IEEE PHM 2012 data challenge. Online website; 2012. http://www.femto-st.fr/en/Research-departments/AS2M/Research-groups/PHM/IEEE-PHM-2012-Data-challenge.php