GAN-LSTM Predictor for Failure Prognostics of Rolling Element Bearings

Hao Lu
Departments of Mechanical Engineering, and Electrical and Computer Engineering
Iowa State University
Ames, IA
hlu1@iastate.edu

Vahid Barzegar
Department of Civil, Environmental and Construction Engineering
Iowa State University
Ames, IA
barzegar@iastate.edu

Venkat P. Nemani
Department of Mechanical Engineering
Iowa State University
Ames, IA
vnemani@iastate.edu

Chao Hu
Departments of Mechanical Engineering, and Electrical and Computer Engineering
Iowa State University
Ames, IA
chaohu@iastate.edu

Simon Laflamme
Department of Civil, Environmental and Construction Engineering, and Electrical and Computer Engineering
Iowa State University
Ames, IA
laflamme@iastate.edu

Andrew T. Zimmerman
Percēv LLC and Grace Technologies
Davenport, IA
andyz@gracetechnologies.com

***Abstract*—Failure prognostics is the process of predicting the remaining useful life (RUL) of machine components, which is vital for the predictive maintenance of industrial machinery. This paper presents a new deep learning approach for failure prognostics of rolling element bearings based on a Long Short-Term Memory (LSTM) predictor trained simultaneously within a Generative Adversarial Network (GAN) architecture. The LSTM predictor takes the current and past observations of a well-defined health index as an input, uses those to forecast the future degradation trajectory, and then derives the RUL. Our proposed approach has three unique features: (1) Defining the bearing failure threshold by adopting an International Organization of Standardization (ISO) standard, making the approach industry-relevant; (2) Employing a GAN-based data augmentation technique to improve the accuracy and robustness of RUL prediction in cases where the deep learning model has access to only a small amount of training data; (3) Integrating the training process of the LSTM predictor within the GAN architecture. A joint training approach is utilized to ensure that the LSTM predictor model learns both the original and artificially generated data to capture the degradation trajectories. We utilize a publicly available accelerated run-to-failure dataset of rolling element bearings to assess the performance of the proposed approach. Results of a five-fold cross-validation study show that the integration of the LSTM predictor with GAN helps to decrease the average RUL prediction error by 29% over a simple LSTM model without GAN implementation.**

Keywords—bearing, remaining useful life, prognostics and health management, generative adversarial network

I. INTRODUCTION

Predicting the remaining useful life (RUL) of mechanical components and industrial systems prior to catastrophic failure is vital for predictive maintenance. In an industrial environment, the failure of rolling element bearings is among the foremost causes of machinery failures [1]**.** The bearing failure may severely affect not only the bearing but also other connected components, leading to catastrophic machine failure [2]. Thus, the ability to accurately predict the RUL of a bearing is practically vital to ensuring the continuous and safe operation of machinery, minimizing unexpected machine downtime, and reducing maintenance costs. However, the bearing degradation is highly non-linear, making accurate prognostics a challenge.

There has been extensive research in the field of bearing prognostics. In general, existing bearing prognostics approaches can be classified into two categories: (a) model-based approaches and (b) data-driven approaches.

The model-based approaches attempt to capture the degradation process of a machine component by constructing mathematical models based on the component's failure mechanisms [3]. The commonly used models include the Paris-Erdogan model, Bailey-Norton model, and exponential degradation model [4]. These approaches require extensive comprehension of the failure mechanisms and accurate calibration of model parameters. However, in most cases, it is challenging to develop an accurate mathematical model for a specific bearing under different operating conditions. If the operating conditions of a bearing change, the prediction result of these approaches tends to be less accurate due to their poor adaptability [5].

The data-driven approaches employ machine learning techniques to capture the bearing degradation pattern without making any assumptions on the underlying damage mechanisms [6]. In recent years, deep learning techniques, such as convolutional neural networks [7-9] and recurrent neural networks [10-12], have become the mainstream techniques for machinery prognostics. Zhu et al. [7] combined wavelet

transform analysis with a convolutional neural network (CNN) to predict the RUL of bearings. In their approach, wavelet transform was first used to extract a time-frequency representation of each sample, which was then fed into a multi-scale CNN for RUL prediction. Ren et al. [8] used the short-time Fourier transform to extract the time-frequency representation of each sample, then calculated the maximum amplitudes at predefined frequency sub-bands, finally fed these amplitudes into a deep CNN for RUL prediction. Wu et al. [10] proposed a vanilla LSTM model for bearing RUL prediction. To enhance the LSTM model's cognitive ability in estimating and predicting degradation, a dynamic differential feature extraction method was utilized that enabled capturing the changes of features under different operating conditions. Guo et al. [11] constructed a health index for bearing RUL prediction by fusing multiple features using a recurrent neural network. Other variants of CNN or LSTM are also applied to predict the RUL of bearings. For example, Yang et al.[9] proposed a double-CNN model for bearing RUL prediction. Zhang et al. [12] combined a multi-layer LSTM network with an attention mechanism to improve the accuracy and robustness in RUL prediction.

Although many data-driven approaches can achieve satisfactory accuracy in bearing RUL predictions, they often face one or more of the following challenges:

1) Extensive research has shown that a bearing's degradation often does not follow a linear trend [13,14]. Prior to the formation of a bearing fault, no clear degradation trend can be revealed from collected data. Soon after the onset of a bearing fault, the degradation may start to accelerate, and the bearing may approach failure in a very short time. The non-linear degradation trend makes correlation/mapping of extracted features directly to RUL a challenge and can yield non-physical results.

2) The performance of existing data-driven approaches depends heavily on the quality and quantity of the available training data used to optimize the parameters of deep learning models [15]. However, gathering large amounts of run-to-failure training data can be very costly and time-consuming, and insufficient available training data may lead to issues such as overfitting.

To address the aforementioned challenges, this study proposes a GAN-based LSTM predictor training method for RUL prediction of rolling element bearings. The main contributions of the proposed approach are summarized as follows:

1) The root mean square (RMS) features in the velocity domain were extracted and used to determine the first prediction time (FPT). A threshold is defined based on ISO standard 10816 [16-17] to determine the end of life (EOL) for bearings, as opposed to traditional heuristic approaches of using the maximum or mean vibration amplitude in the acceleration domain.

2) A GAN-based LSTM framework for bearing degradation data augmentation is developed to enhance the model's prediction accuracy and robustness in forecasting future degradation trajectories.

3) A joint training strategy is developed by integrating the training of LSTM predictor into the training of GAN.

The proposed approach implements the following two steps in the offline process: (1) Data Preparation extracts segments of a degradation feature from the collected vibration data in the velocity domain; and (2) Model Training first pre-trains the GAN-LSTM predictor based on the original training data, then pre-trains the generator and discriminator using the proposed GAN-LSTM network, and finally jointly trains the GAN-LSTM predictor, generator, and discriminator. After the GAN-LSTM network has been fully trained, the GAN-LSTM predictor can be used for online tests to predict the RUL of test bearings. The remainder of the paper is organized as follows: Section II introduces the methodology, including the data preparation, the GAN-LSTM predictor training, and RUL prediction. The performance of the proposed approach is evaluated in Section III through a five-fold cross-validation study on a publicly available dataset. Finally, several concluding remarks are given in Section IV.

II. METHODOLOGY

Figure 1 shows a flowchart of the proposed approach. In our methodology, run-to-failure vibration data is first transformed to velocity domain $v(t)$ by numerical integration of the acceleration vibration signal $a(t)$. Then, feature extraction is conducted on the velocity data. The root mean square (RMS) in the velocity domain is calculated in the range of 0.2 times the shaft frequency, 0.2ω, to the end of frequency spectrum (which is one-half the sampling frequency sf for a single-sided fast Fourier transform spectrum) and is used as the health index of a bearing. The extracted features ($V_{0.2\omega-sf/2}^{\mathrm{RMS}}$) are analyzed to determine the FPT using the 2σ method [21]. The ISO standard-based threshold is used to determine the EOL [16]. The calculations of FPT and EOL are described in section 2.1. After determining the FPT and EOL, data segmentation is performed to generate training data through a moving window segmentation function where a moving window with a predetermined length takes segments in sequential order. A GAN-LSTM architecture is used to train the GAN-LSTM degradation pattern predictor. Through joint training, the GAN-LSTM predictor is trained using both the real training data and the generated data obtained from the generator network. The following sections present the details of each step.

A. Data Preparation

The data pre-processing is composed of two parts: (1) feature extraction and (2) determination of FPT and EOL.

Most lab-based accelerated bearing run-to-failure datasets provide vibration data in the acceleration domain [18]-[19]. However, the industry-relevant ISO standards for defining the EOL or alarm amplitudes are defined in the velocity domain [16]. This is because the amplitude of the acceleration signal increases with the shaft frequency, whereas the amplitude of the signal in the velocity domain provides a more stable representation [17]. To obtain degradation patterns reflecting the bearing damage severity, the velocity RMS in the frequency

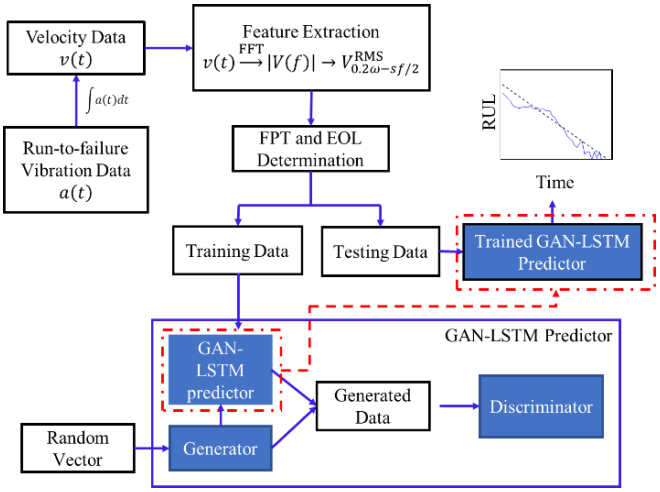

Figure 1. A flowchart of the proposed approach

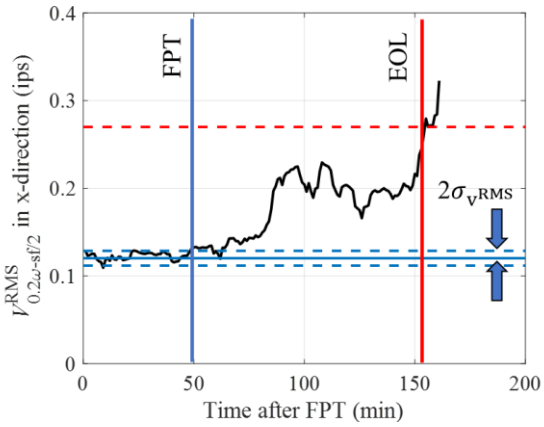

Figure 2. Evolution of $V^{RMS}_{0.2\omega-sf/2}$ and identification of FPT, EOL

range of $0.2\omega - sf/2$ Hz is used as a degrading feature for the sampling frequency $sf = 25.6$ kHz with operating shaft frequency ω. The extracted run to failure feature is a sequence of velocity-RMS values $V^{RMS}_{0.2\omega-sf/2}$ defined using the Parseval's theorem as [20]:

$$V^{RMS}_{0.2\omega-sf/2} = \sqrt{\sum_{f=0.2\omega}^{sf/2} \frac{|V(f)|^2}{2}} \quad (1)$$

where $V(f)$ is the single-sided frequency spectrum for $v(t)$. A larger value of $V^{RMS}_{0.2\omega-sf/2}$ represents a more severely damaged state. At the beginning of the run-to-failure tests, bearings are often healthy and no clear degradation trend can be revealed, resulting in an approximately flat RMS sequence. FPT and EOL were defined to obtain the degradation curve between where the bearing starts to experience damage up to its EOL. In this paper, the 2σ method, which was proposed in Ref. [21], was used to determine the FPT. The RMS values are used to employ the 2σ criterion in the velocity domain $V^{RMS}_{0.2\omega-sf/2}$. The mean $\mu_{V^{RMS}}$ and standard deviation $\sigma_{V^{RMS}}$ were calculated using the data collected at the beginning of the experiment, then the FPT was obtained when $V^{RMS}_{0.2\omega-sf/2}$ crosses the threshold of $\mu_{V^{RMS}} + 2\sigma_{V^{RMS}}$ for two consecutive observations.

Based on the ISO standard, the EOL of the bearing is defined when the velocity RMS reaches a given threshold. Here, the threshold was set at 0.27 ips reflecting the ISO standard alarm state for medium-sized motors. For bearing prognostics, the true RUL of a bearing was defined to decrease linearly with a unit slope from the FPT to EOL. The development of $V^{RMS}_{0.2\omega-sf/2}$ is depicted in Figure 2 for a sample bearing.

B. The GAN-LSTM predictor

There are two approaches to performing bearing RUL prediction using deep learning techniques: direct mapping approach and trajectory prediction approach. The direct mapping approach takes the extracted features as input and produces an RUL estimate as output [22-23]. The trajectory prediction approach takes the historical measurements of a feature as the input, forecasts the future trajectory of this feature, then calculates the RUL as the time at which the trajectory crosses a failure threshold [24]. The proposed method adopts the second approach. Specifically, a one-step ahead predictor, called GAN-LSTM predictor, is proposed to forecast the degradation of a bearing. The predictor's performance is enhanced by augmenting real training data with synthetic data generated by a GAN. To obtain a complete forecast up to the failure threshold, the GAN-LSTM predictor is repeatedly evaluated by marching in time, treating predicted values as new data points.

The GAN-LSTM predictor is composed of an LSTM layer and a fully connected layer. LSTM is a special type of recurrent neural network (RNN). It utilizes memory cells to retain useful information within both long and short periods often with no issue of vanishing gradients [22]. Each LSTM unit has three gates: input gate, forget gate, and output gate. The input gate determines what new information should be stored in the memory cell, the forget gate determines what information should be discarded from the previous cell state, and the output gate determines what the hidden state at the next time step should be based on the previous cell state, the current input, the current cell state. A more detailed introduction to LSTM can be found in Ref. [25].

Given training time-series data, the GAN-LSTM predictor takes the features at the current and previous $(k-1)$ time steps as the input, and predicts the feature value at the next time step. The LSTM layer extracts temporal degradation information, which will then be used to determine the feature value at the next step by using a fully connected layer.

The GAN-LSTM predictor is a deep learning model, and its performance highly depends on the quantity and quality of the training data. Data augmentation techniques are commonly used to cope with the lack of available training data. Some of the most common data augmentation techniques used for bearing RUL prediction include adding noise and shrinking or extending the training data [26]. Recent research has also shown the promise of performing data augmentation using GANs [27-28]. Here, a GAN-based framework for degradation data augmentation is proposed.

In its basic form, a GAN comprises two separate networks working with opposing goals: (1) a generator network and (2) a discriminator network. The principle of a generator is to create synthetic sequences with a similar probability distribution to that of the real data while the discriminator tries to distinguish between the real and synthetic sequences. The competition between the two networks enhances the quality of the generated data when the fully-trained discriminator fails to distinguish the real and the synthetic data. The training of the GAN involves alternate optimizations of the generator and the discriminator. The objective function of the discriminator consists of two parts. The first part is the probability that real data is classified as real data, and the second objective is the probability that the generated data is classified as synthetic data. The objective function of the discriminator is written as:

$$\max\ (V_D) = \frac{1}{n}\sum_{i=1}^{n}[\log D(x_i) + \log(1 - D(G(z_i)))] \quad (2)$$

where x_i is the real degradation data, z_i is the random noise, $D(x_i)$ denotes the output given by the discriminator when the input is x_i, and $G(z_i)$ represents the generated sample. The training process of the discriminator tries to maximize the objective function V_D so that the well-trained discriminator could identify data correctly.

For each training epoch, after the discriminator is trained, the generator is trained by keeping the discriminator's weights constant. The generator takes random noise as input and generates synthetic data. The object of the generator is to confuse the discriminator. The objective function of the generator can be written as:

$$\min\ (V_G) = \frac{1}{n}\sum_{i=1}^{n}\log\ (1 - D(G(z_i)) \quad (3)$$

The training of GAN involves the optimization of V_D and V_G iteratively. Conceptually, the training of GAN can correspond to a minimax two-player game [29] written as:

$$\min_{G}\ \max_{D}\ V(D,G) = \frac{1}{n}\sum_{i=1}^{n}[\log D(x_i) + \log\ (1 - D(G(z_i)))] \quad (4)$$

The ideal objective for GAN is that the discriminator cannot distinguish the generated data from the real data. The generated data from a well-trained generator will mimic the distribution of the training data. A straightforward way to perform GAN-based data augmentation is to train the GAN based on the training dataset, then combine generated data with training data to train the deep learning model. In our proposed method, we integrate the training of the GAN-LSTM predictor into the training of the GAN architecture for a more robust and direct implementation.

The architecture of the GAN-LSTM network is illustrated in Figure 3. The generator takes a random sequence of length l and outputs a vector of synthetically generated data of the same length. The LSTM predictor takes the output and predicts the feature at the next step in the sequence. The one-step ahead prediction is then appended to the generated data and is fed into the discriminator to be labeled as real or generated data. The network is trained by back-propagating error through time and the standard gradient descent optimization method [29].

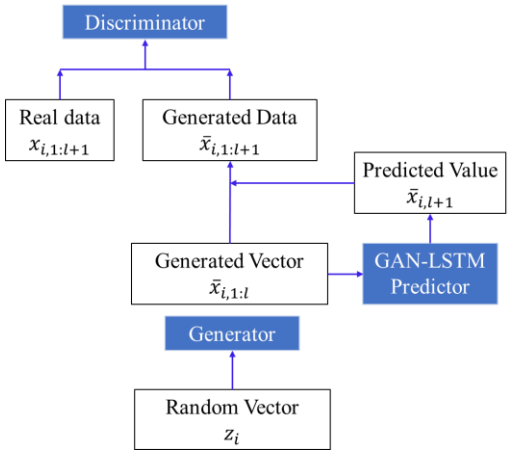

Figure 3. A flowchart depicting the essential elements of the GAN-LSTM network

Here, the GAN-LSTM predictor is pre-trained on the limited available data. The architecture shown in Figure 3 is then used to enhance the prediction accuracy by generating augmented sequences. The training of the GAN-LSTM predictor is composed of three steps:

1) Pre-training of the GAN-LSTM predictor: The GAN-LSTM predictor is pre-trained using the real data.

2) Pre-training of the generator and discriminator: The generator and discriminator are trained iteratively by following the structure proposed in Figure 3, while the GAN-LSTM predictor's parameters are frozen.

3) Joint training of the generator, GAN-LSTM predictor, and discriminator: The generator, GAN-LSTM predictor, and discriminator are all trained in this step. Each joint training epoch is composed of two sub-steps. Firstly, the GAN-LSTM predictor, the generator, and the discriminator are jointly trained following the structure proposed in Figure 3. Then, the GAN-LSTM predictor is fine-tuned using the real data.

C. RUL Prediction

The one-step ahead GAN-LSTM predictor is used to predict the trajectory of the degradation feature up to the EOL. An input sequence of length l is fed into the trained LSTM predictor with the time step of the last element marked as $t_{\text{Inspection}}$. Each next-step prediction value is used as a new data point, and the prediction is performed until the predicted value crosses the predefined threshold of EOL and marking the time as T_{EOL}. Then, the RUL is computed as:

$$\text{RUL}(t) = T_{\text{EOL}} - t_{\text{Inspection}} \quad (5)$$

The RUL prediction procedure is specified in TABLE I.

TABLE I. THE PROCEDURE OF RUL PREDICTION PROCEDURE

Algorithm: RUL prediction
Inputs: Real run-to-failure feature series $F = F_t$,
Inspection time $t_{\text{Inspection}}$,
LSTM predictor Θ_p,
Failure threshold V_{cutoff}
Output: RUL

Initialize $t_i = t_{\text{Inspection}}, V_i^{\text{RMS}} = F(t_i)$,
While $(V_i^{\text{RMS}} \leq V_{\text{cutoff}})$:
  Set the input $F_{\text{Input}} = F_{t_i-l+1:t_i}$;
  Next step prediction: $V_{i+1}^{\text{RMS}} = \Theta_p(F_{\text{Input}})$;
  Update $F_{t_i+1} = V_{i+1}^{\text{RMS}}$;
  $t_i = t_i + 1$
end while
$T_{\text{EOL}} = t_i$
Output: return RUL $= T_{\text{EOL}} - t_{\text{Inspection}}$

III. EXPERIMENTAL VALIDATION

The run-to-failure bearing dataset from XJTU-SY was utilized to evaluate the proposed method's performance [14]. The XJTU-SY bearing dataset consisted of run-to-failure vibration data of 15 rolling element bearings (LDK UER 204) divided into three groups of five bearings. Each group was subjected to a certain operating condition. Two PCB 352C33 accelerometers were mounted perpendicularly along the radial direction. Data were collected for 1.28 sec every minute with a sampling frequency of 25.6 kHz. The operating conditions for the entire bearing dataset are shown in TABLE II. The extracted $V_{0.2\omega-sf/2}^{\text{RMS}}$ features (from FPT to EOL) are shown in Figure 4.

A five-fold cross-validation study was conducted on the set of 15 bearings. 15 bearings were divided into five folds as:

Fold 1: Bearing 1-1, Bearing 2-1, Bearing 3-1

Fold 2: Bearing 1-2, Bearing 2-2, Bearing 3-2

Fold 3: Bearing 1-3, Bearing 2-3, Bearing 3-3

Fold 4: Bearing 1-4, Bearing 2-4, Bearing 3-4

Fold 5: Bearing 1-5, Bearing 2-5, Bearing 3-5

TABLE II. XJTU-SY BEARING DATASET.

	Operating condition		
	Condition 1	Condition 2	Condition 3
Radial load	12 kN	11 kN	10 kN
Speed	35 Hz	37.5 Hz	40 Hz
Bearing ID	Bearing 1-1	Bearing 2-1	Bearing 3-1
	Bearing 1-2	Bearing 2-2	Bearing 3-2
	Bearing 1-3	Bearing 2-3	Bearing 3-3
	Bearing 1-4	Bearing 2-4	Bearing 3-4
	Bearing 1-5	Bearing 2-5	Bearing 3-5

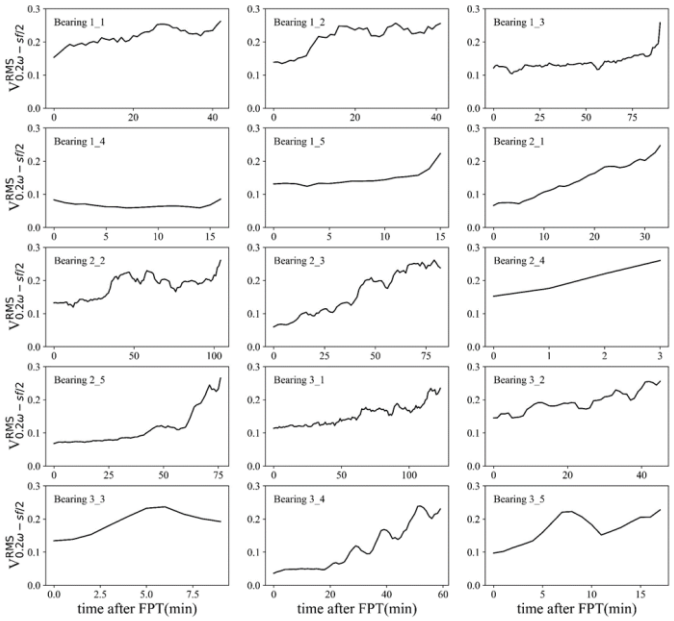

Figure 4. RMS features of each bearing

While evaluating the model, one fold was chosen to be the test set, and the other four folds were used for training the models. For example, if fold 1 served as the test set (Bearing 1-1, Bearing 2-1, and Bearing 3-1 are test bearings), then the data collected from the other 12 bearings served as the training set. The length of the network input, l, the learning rate of the GAN-LSTM predictor, and the learning rate of both the generator and discriminator were heuristically set as 20, 0.001, and 0.0001, respectively. As mentioned in section 2, the GAN-LSTM predictor, generator, and the discriminator were pre-trained using the available datasets for 60, 1000, and 1000 epochs, respectively. During the joint training, the GAN-LSTM predictor, generator, and discriminator were trained for 60 epochs. The architecture of the GAN-LSTM network is shown in TABLE III.

TABLE III. THE SPECIFIC CONFIGURATION OF THE PROPOSED GAN-LSTM NETWORK.

Module name	layer	Parameters
Generator	Fully connected layer	Output shape = (Samples, 64), Activation = Linear
	Fully connected layer	Output shape = (Samples, 32), Activation = Linear
	Fully connected layer	Output shape = (Samples, 20), Activation = ReLU
Discriminator	Fully connected layer	Output shape = (Samples, 64), Activation = Linear
	Fully connected layer	Output shape = (Samples, 128), Activation = ReLU
	Fully connected layer	Output shape = (Samples, 64), Activation = ReLU
	Fully connected layer	Output shape = (Samples, 1), Activation = Sigmoid
GAN-LSTM predictor	LSTM layer	Number of LSTM Units = 60
	Fully connected layer	Output shape = (Samples, 1), Activation = Linear

To evaluate the performance of the proposed approach, the LSTM model without being trained jointly with GAN network was used as a baseline model, named as LSTM predictor. The LSTM predictor + noise model was also included for benchmarking the proposed method; this model was trained with the real training data and the training data synthetically generated by adding Gaussian noise to the real training data.

In addition to deep learning models, a quadratic regression model was also included. The regression fitting was performed on features at the current and previous 19 time steps. The model used to capture the bearing degradation trend is defined as:

$$V^{\mathrm{RMS}}(t) = m_1 t^2 + m_2 t + m_3 \quad (6)$$

where $V^{\mathrm{RMS}}(t)$ represents the feature value at time t, and m_1,m_2 and m_3 are the unknown model parameters which were determined by the ordinary least square method. Similar to the GAN-LSTM predictor, after model parameters m_1,m_2 and m_3 are determined, the feature at the next time step, $V^{\mathrm{RMS}}(t+1)$, will be predicted, and this next-step prediction will be used as a new data point. This process of quadratic regression is repeated until the predicted value crosses the predefined threshold. The regression model focuses on capturing a local quadratic trend. However, if the feature does not change significantly within the previous 19 steps, the model cannot provide reliable prediction results. In this paper, if the predicted values do not reach the threshold for the next 100 steps, the model's RUL prediction at that inspection time point will be deemed unreliable and the model will take the most recent reliable prediction result as the predicted RUL.

To evaluate the performance of the generator network and the quality of the synthetic data, the t-Distributed Stochastic Neighbor Embedding (t-SNE) analysis [31] was conducted to visualize how well the generated distribution resembles the distribution of real data. The t-SNE analysis reduces the data dimensionality by mapping from high-dimensional to lower-dimensional spaces. **Error! Reference source not found.** shows the distributions of the real and generated data for cross-validation run fold 1. This graphical comparison shows the ability of the generator network in mimicking the distribution of the real data, hence producing high-quality synthetic data which helps deal with the challenge of limited training data.

The root-mean-square error (RMSE) of RUL prediction was used to evaluate the performance of the selected methods in predicting the trajectory of the degradation feature, written as:

$$\mathrm{RMSE} = \sqrt{\frac{1}{(t_{\mathrm{EOL}} - t_{\mathrm{FPT}} + 1)} \sum_{t=t_{\mathrm{FPT}}}^{t_{\mathrm{EOL}}} \left(\mathrm{RUL}_t(t) - \widehat{\mathrm{RUL}}(t)\right)^2} \quad (7)$$

where t_{FPT} is the time when prognostics starts, $\mathrm{RUL}_t(t)$ and $\widehat{\mathrm{RUL}}(t)$ are the true RUL and predicted RUL at time step t, respectively. RMSE is a measure of the error in RUL prediction during the period of bearing prognostics (from t_{FPT} to t_{EOL}). The RUL prediction results for all 15 bearings are summarized in TABLE IV where the bearings are arranged in an ascending order of length of prognostic time period ΔT, defined as $\Delta T = t_{\mathrm{EOL}} - t_{\mathrm{FPT}} + 1$. The model which provides the least RMSE error for each bearing is highlighted in bold. Overall, on average, the proposed GAN-LSTM predictor produced more accurate results with 45%, 29%, and 27% improvement in RMSE value compared to the quadratic regression model, baseline LSTM predictor model, and LSTM predictor + noise model. Note that for bearings 2-4, 3-5, 3-3, 1-5, and 1-4, the number of time steps in the prognostic time period is smaller than the selected input length of the LSTM predictor. For these bearings, some data before t_{FPT} were used as input, and these data do not provide enough degradation information for making accurate predictions. Also, with the exception of bearings 2-1, 3-4, and 2-5, the proposed model performed significantly better for bearings with longer prognostic time periods..

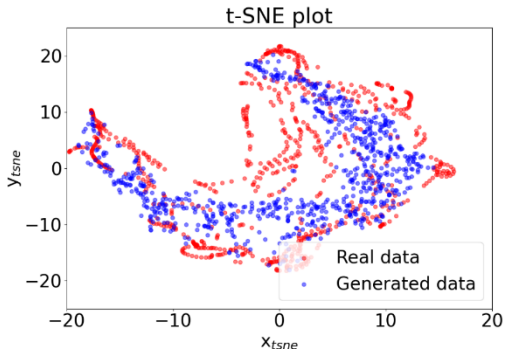

Figure 5. t-SNE plot of real data and generated data

Figure 6 shows the predicted RUL and degradation curves for test bearing 3-2. During the early stage of bearing degradation (from time $t = 0$ to $t = 14$ min), the proposed GAN-LSTM predictor produced more accurate results. As the degradation progressed, the extracted feature became closer to the predefined failure threshold and easier to predict, making the RUL predictions by the three approaches look more similar. Also, note that the quadratic regression model produced the least accurate results at the beginning of degradation but converged to the LSTM predictor with occasional peaks in error. The degradation trajectories predicted by the four models at three different inspection times are plotted in Figure 7. The GAN-LSTM predictor yielded more accurate predictions of the degradation trend, especially at the early stage of degradation, which resulted in the better RUL estimations depicted in Figure 6, and listed in TABLE IV. Moreover, at time $t = 20$ min, the degradation trend became relatively flat. The quadratic model provided unreliable results while the future trajectory generated by the GAN-LSTM predictor still captured the global trend. Given the ability of LSTM to learn long-term dependencies, it was able to capture the global trend, whereas the quadratic regression model focused more on capturing a local quadratic trend. The peaks in RUL prediction at times around 29 min and 42 min in Figure 6 are also attributed to this characteristic of the quadratic regression model, where changes in local trends resulted in inaccurate predictions.

TABLE IV. RUL PREDICTION RESULTS BY THE GAN-LSTM PREDICTOR AND OTHER APPROACHES (SORTED BY ASCENDING ORDER OF PROGNOSTIC DURATION)

Bearing ID	t_{FPT} (min)	t_{EOL} (min)	ΔT (min)	RMSE			
				Quadratic regression	LSTM predictor	LSTM predictor + noise	GAN-LSTM predictor
Bearing 2-4	32	35	4	9.94	13.26	**4.74**	24.39
Bearing 3-5	20	25	6	147.51	13.47	**4.18**	10.89
Bearing 3-3	343	352	10	7.96	14.91	**7.35**	21.99
Bearing 1-5	26	41	16	52.21	22.97	**13.49**	30.93
Bearing 1-4*	106	122	17	62.46	42.66	**22.05**	48.43
Bearing 2-1	456	489	34	6.31	**5.34**	10.14	42.59
Bearing 1-2	55	96	42	15.32	13.91	13.21	**12.83**
Bearing 1-1	79	121	43	14.16	15.71	14.4	**9.04**
Bearing 3-2	2450	2495	46	34.19	12.2	10.9	**4.89**
Bearing 3-4	1420	1479	60	13.42	**8.23**	12.99	12.6
Bearing 2-5	123	199	77	40.46	**11.72**	13.36	17.41
Bearing 2-3	316	398	83	27.71	15.74	25.12	**9.93**
Bearing 1-3	60	150	91	61.07	25.59	31.12	**20.37**
Bearing 2-2	50	154	105	45.38	43.99	43.56	**29.69**
Bearing 3-1	2404	2527	124	31.54	53.84	47.29	**21.25**
Cumulative#				39.7	31	29.91	**21.9**

*Bearing 1_4 undergoes catastrophic failure and is therefore very difficult for prediction.
#Cumulative represents the RMSE among all the bearings weighted by the prognostic time duration ΔT.

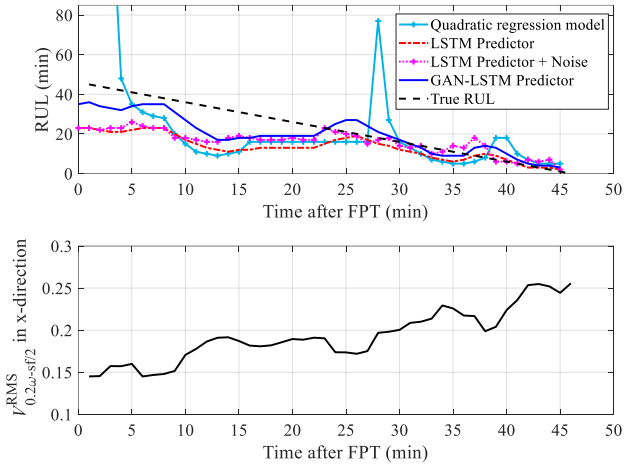

Figure 6. (a) RUL prediction result for bearing 3-2 and (b) the degradation curve of bearing 3-2

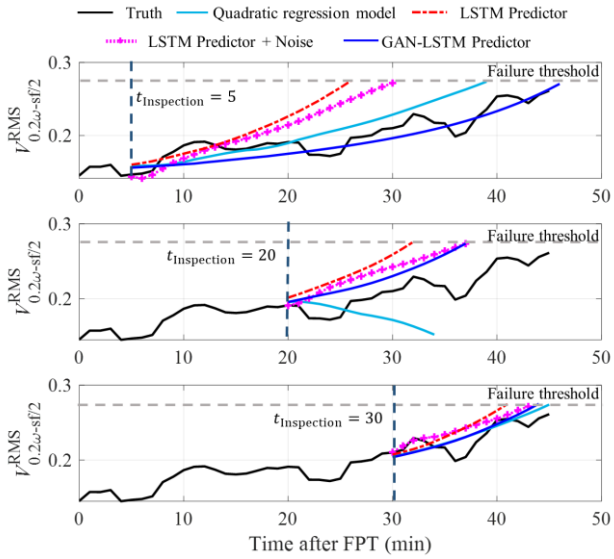

Figure 7. The predicted trajectories at three inspection times for bearing 3_2

IV. CONCLUSION

In this paper, we proposed a long short-term memory (LSTM) predictor trained within a generative adversarial network (GAN) architecture for the failure prognostics of rolling element bearings. The ISO standards are adopted in defining the bearing failure threshold, making the proposed approach industry-relevant. By integrating the training of LSTM predictor within the GAN architecture, the LSTM predictor utilizes the prognostic information from both the original, real data and the generated data to better capture the degradation trajectories.

Using a publicly available bearing run-to-failure dataset (XJTU-SY), we show the superiority of our approach when compared to the quadratic regression model, LSTM predictor model, and LSTM predictor trained with data augmented by adding noise. The proposed GAN-LSTM predictor better captures the degradation trajectory, thus leading to a more accurate RUL prediction with a reduction of RMSE error in RUL prediction by 29% when compared with the LSTM predictor model in a five-fold cross-validation study.

ACKNOWLEDGMENT

This work was supported in part by the U.S. National Science Foundation under Grant IIP-1919265, the Regents Innovation Fund that is part of the Proof-of-Concept Initiative at Iowa State University, and Grace Technologies. Any opinions, findings, or conclusions in this paper are those of the authors and do not necessarily reflect the views of the sponsors.

REFERENCES

[1] Lee, J., Wu, F., Zhao, W., Ghaffari, M., Liao, L. and Siegel, D., 2014. Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications. *Mechanical systems and signal processing*, *42*(1-2), pp.314-334.

[2] Atamuradov, V., Medjaher, K., Dersin, P., Lamoureux, B. and Zerhouni, N., 2017. Prognostics and health management for maintenance practitioners-review, implementation and tools evaluation. *International Journal of Prognostics and Health Management*, *8*(060), pp.1-31.

[3] Cubillo, A., Perinpanayagam, S. and Esperon-Miguez, M., 2016. A review of physics-based models in prognostics: Application to gears and bearings of rotating machinery. *Advances in Mechanical Engineering, 8(8),* p.1687814016664660.

[4] Wang, T., Liu, Z. and Mrad, N., 2020. A Probabilistic Framework for Remaining Useful Life Prediction of Bearings. *IEEE Transactions on Instrumentation and Measurement, 70,* pp.1-12.

[5] Liu, L., Song, X., Chen, K., Hou, B., Chai, X. and Ning, H., 2021. An enhanced encoder–decoder framework for bearing remaining useful life prediction. *Measurement, 170,* p.108753.

[6] Wu, J., Hu, K., Cheng, Y., Zhu, H., Shao, X. and Wang, Y., 2020. Data-driven remaining useful life prediction via multiple sensor signals and deep long short-term memory neural network. *ISA transactions, 97,* pp.241-250.

[7] Zhu, J., Chen, N. and Peng, W., 2018. Estimation of bearing remaining useful life based on multi-scale convolutional neural network. *IEEE Transactions on Industrial Electronics, 66(4),* pp.3208-3216.

[8] Ren, L., Sun, Y., Wang, H. and Zhang, L., 2018. Prediction of bearing remaining useful life with deep convolution neural network. *IEEE Access, 6,* pp.13041-13049.

[9] Yang, B., Liu, R. and Zio, E., 2019. Remaining useful life prediction based on a double-convolutional neural network architecture. *IEEE Transactions on Industrial Electronics, 66(12),* pp.9521-9530.

[10] Wu, Y., Yuan, M., Dong, S., Lin, L. and Liu, Y., 2018. Remaining useful life estimation of engineered systems using vanilla LSTM neural networks. *Neurocomputing, 275,* pp.167-179.

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

[12] Zhang, H., Zhang, Q., Shao, S., Niu, T. and Yang, X., 2020. Attention-based LSTM network for rotatory machine remaining useful life prediction. *IEEE Access, 8,* pp.132188-132199.

[13] Wang, Y., Xiang, J., Markert, R. and Liang, M., 2016. Spectral kurtosis for fault detection, diagnosis and prognostics of rotating machines: A review with applications. *Mechanical Systems and Signal Processing, 66,* pp.679-698.

[14] M. Sadoughi, H. Lu and C. Hu, "A Deep Learning Approach for Failure Prognostics of Rolling Element Bearings," *2019 IEEE International Conference on Prognostics and Health Management (ICPHM), San Francisco, CA, USA, 2019,* pp. 1-7, doi: 10.1109/ICPHM.2019.8819442.

[15] Yaguo Lei, Naipeng Li, Liang Guo, Ningbo Li, Tao Yan, Jing Lin,Machinery health prognostics: A systematic review from data acquisition to RUL prediction,*Mechanical Systems and Signal Processing, Volume 104,2018,*Pages 799-834, ISSN 0888-3270, https://doi.org/10.1016/j.ymssp.2017.11.016.

[16] 14:00-17:00, "ISO 10816-3:2009," ISO. https://www.iso.org/cms/render/live/en/sites/isoorg/contents/data/standard/05/05/50528.html (accessed Oct. 28, 2020).

[17] R. L. Eshleman, Basic Machinery Vibrations: An Introduction to Machine Testing, Analysis, and Monitoring. *VIPress,* 1999.

[18] Wang, B., Lei, Y., Li, N. and Li, N., 2018. A hybrid prognostics approach for estimating remaining useful life of rolling element bearings. *IEEE Transactions on Reliability, 69(1),* pp.401-412.

[19] Nectoux, P., Gouriveau, R., Medjaher, K., Ramasso, E., Chebel-Morello, B., Zerhouni, N. and Varnier, C., 2012, June. PRONOSTIA: An experimental platform for bearings accelerated degradation tests. *In IEEE International Conference on Prognostics and Health Management, PHM'12*. (pp. 1-8). IEEE Catalog Number: CPF12PHM-CDR.

[20] H. J. Nussbaumer, "The Fast Fourier Transform," in *Fast Fourier Transform and Convolution Algorithms*, H. J. Nussbaumer, Ed. Berlin, Heidelberg: Springer, 1981, pp. 80–111.

[21] Li, X., Zhang, W. and Ding, Q., 2019. Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction. *Reliability Engineering & System Safety*, *182*, pp.208-218.

[22] Ren, L., Sun, Y., Cui, J. and Zhang, L., 2018. Bearing remaining useful life prediction based on deep autoencoder and deep neural networks. *Journal of Manufacturing Systems, 48,* pp.71-77.

[23] Shi, Z. and Chehade, A., 2021. A dual-LSTM framework combining change point detection and remaining useful life prediction. *Reliability Engineering & System Safety, 205,* p.107257.

[24] Yu, Y., Hu, C., Si, X., Zheng, J. and Zhang, J., 2020. Averaged Bi-LSTM networks for RUL prognostics with non-life-cycle labeled dataset. *Neurocomputing*, *402*, pp.134-147.

[25] Barzegar, V., Laflamme, S., Hu, C. and Dodson, J., 2021. Multi-time resolution ensemble lstms for enhanced feature extraction in high-rate time series. *Sensors*, *21*(6), p.1954.

[26] Yu, K., Lin, T.R., Ma, H., Li, X. and Li, X., 2021. A multi-stage semi-supervised learning approach for intelligent fault diagnosis of rolling bearing using data augmentation and metric learning. *Mechanical Systems and Signal Processing*, *146*, p.107043.

[27] X. Zhang, Y. Qin, C. Yuen, L. Jayasinghe and X. Liu, "Time-Series Regeneration with Convolutional Recurrent Generative Adversarial Network for Remaining Useful Life Estimation," in IEEE Transactions on Industrial Informatics, doi: 10.1109/TII.2020.3046036.

[28] Y. Huang, Y. Tang and J. Vanzwieten, "Prognostics with Variational Autoencoder by Generative Adversarial Learning," in *IEEE Transactions on Industrial Electronics*, doi: 10.1109/TIE.2021.3053882.

[29] Ruder, S., 2016. An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.

[30] Goodfellow, I.J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y., 2014. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*.

[31] Van der Maaten, L. and Hinton, G., 2008. Visualizing data using t-SNE. *Journal of machine learning research*, *9*(11).