

Predictive maintenance of electromechanical systems based on enhanced generative adversarial neural network with convolutional neural network

Azhar Muneer Abood, Ahmed Raof Nasser, Huthaifa Al-Khazraji

Department of Control and Systems Engineering, University of Technology, Baghdad, Iraq

Article Info

Article history:

Received Jun 29, 2022

Revised Dec 23, 2022

Accepted Jan 10, 2023

Keywords:

Convolutional neural network
Deep learning
Generative adversarial neural network
Predictive maintenance

ABSTRACT

Predictive maintenance (PdM) is a cost-cutting method that involves avoiding breakdowns and production losses. Deep learning (DL) algorithms can be used for defect prediction and diagnostics due to the huge amount of data generated by the integration of analog and digital systems in manufacturing operations. To improve the predictive maintenance strategy, this study uses a hybrid of the convolutional neural network (CNN) and conditional generative adversarial neural network (CGAN) model. The proposed CNN-CGAN algorithm improves forecast accuracy while substantially reducing model complexity. A comparison with standalone CGAN utilizing a public dataset is performed to evaluate the proposed model. The results show that the proposed CNN-CGAN model outperforms the conditional GAN (CGAN) in terms of prediction accuracy. The average F-Score is increased from 97.625% for the CGAN to 100% for the CNN-CGAN.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Azhar Muneer Abood

Department of Control and Systems Engineering, University of Technology

Baghdad, Iraq

Email: cse.20.09@grad.uotechnology.edu.iq

1. INTRODUCTION

The term predictive is defined as a state or behavior that will occur in the future. The task of maintaining a machine is necessary to keep it in good working order. Therefore, predictive maintenance (PdM) is defined as a way to predict the future failure of the machine's tool before it fails [1]. PdM became a key role in growing the productivity and the profitability of the industrial system. For this, it has obtained wide attention in the last years in research.

Due to condition monitoring of all industrial equipment, get together with deep learning methods, the maintenance task has been enhanced in modern production systems [2]. Besides, data acquisition, data collected by smart sensors are available nowadays to make a file estimation and prediction of the current health condition and machine tools [3]. Big data is not just the size of the collected data, but it also contains the properties, the variety, and the velocity of data. The essential pattern of overall data becomes a major concern for companies to investigate the utility of big data analytics. The main goal of large data analysis is to define the attributes of data with the aim to derive patterns and connections in the data. In addition, big data analytics aims to find the data functions that are descriptive such as classification, clustering, association, and logistic regression analysis [4], [5].

Deep learning (DL) is more similar to the human brain. It is a subgroup of machine learning methods, which is learning the many levels of representations of data with different levels of abstraction at each stage [6]. DL is an efficient data feature extraction algorithm because it can overcome the problem of extracting

features that are involved in nonlinear big data by shallow learning. DL may be supervised or unsupervised. The important features of data are extracted by using multiple levels of nonlinear processing units. The input to the next layer is provided from the output of the current layer. The method of stochastic gradient descent is used by the backpropagation algorithm to reach an ideal in the training set. Many types of deep structures are used such as recurrent neural network (RNN), long short term memory (LSTM), convolutional neural network (CNN), and deep belief network (DBN) [7], [8].

When compared to a regular or fully connected neural network (FCNN), the deep neural network has a different structure. A straightforward neural network can be modified to a deep neural network by adding two or more layers between the input and output layers [9], [10]. Conventional computing hardware that depends on the central processing units (CPU) is not appropriate to handle the multilayer architecture because this architecture deals with countless links between the layers. However, graphics processing unit (GPU) techniques overcome this issue by executing and detecting multiple features and higher-order feature relationships. Consequently, adding more layers between input and output in deep architectures was made possible by the use of GPU processing and the presence of a sizable training set of data [11].

In this paper, a modified version of generative adversarial neural network (GAN) named conditional GAN (CGAN) is introduced to predict the multiclass fault of an electromechanical system (motor) in the early stage using a data set of asynchronous common motor fault (ACMF) in a normal (healthy) state and seven abnormal (unhealthy) states. Next, CNN is used as extraction features of the training dataset of ACMF and then passes it to CGAN. Finally, a comparison between the CGAN model and the hybrid CNN-CGAN model is performed to evaluate the proposed model.

2. PROPOSED DEEP LEARNING ARCHITECTURE

This section presents the theoretical framework of the proposed deep learning model. The first subsection explains the GAN model. The next subsection introduces the CNN model.

2.1. Generative adversarial neural network

GANs are a come-up approach for both semi-supervised and unsupervised learning. It was proposed in 2014 by modeling high-dimensional distributions of data. There are various types of GAN networks like CGAN, Cycle GAN, Wasserstein GAN, and Vanilla GAN. GANs can be described by the training of two networks in competition with each other. The first network is known as a fake artist and the second as an art expert. In the GAN literature, the fake called the generator (G) generates fakes, with the purpose of making realistic data. The expert, called the discriminator (D) gets both fake and real data, to distinguish between them. The G and D are trained simultaneously, and in competition with each other [12].

Definitely, the G has no direct connection to the real data, the only path it learns via interaction with the D. On the other hand, D has reached both the fake and real samples taken from the stack of real data. The mistaken signal to D is supplied by determining whether the data came from the real stack or from G. G received the error from D, and then this error is used to train and make forgeries have better quality. The implementation of the network includes the G and D by multilayer consisting of convolutional and/or fully connected layers.

The G and D network is not necessary to be directly invertible and must be differentiable. The G network is an analysis of some description space, denoted a (latent space), to the space of the data [13]. Basically, in the GAN model, the D network may be similarly described as a function that maps from data to an eventuality that the data is from the real data allocation, rather than the G allocation: $D: (Dx) (0 \text{ or } 1)$. For a fixed G, the D may be trained to classify fault as either being from the training data (real, refer to 1) or from a fixed generator (fake, refer to 0). The G may keep being learned so as to lower the accuracy of the D when the D is optimal and it may be fixed. If the generator allocation is able to reach the real data allocation perfectly, then the D will be maximally confused, predicting 50% for all inputs [14]. Figure 1 shows the GAN architectures.

Depending on the binomial zero-sum game theory, networks of different types could exist for the G and D architectures, such as a fully-connected layer, CNN, and autoencoder. Typically G and D are modeled using nonlinear mapping equations. Through training, D tries to give a high likelihood based on real data, and give data from G a low probability. Conversely, G creates false data while learning the distribution of real data in order to fool D [15]. In this competitive learning, the enhancement of G and D efficiency can be introduced. The minimax two-player game can be expressed by (1),

$$\min_G \max_D V(D, G) = E_{x \sim p_{data}(x)} [\log D(x)] + E_{z \sim p_z(z)} [\log (1 - D(G(z)))] \quad (1)$$

Signify the distribution of real data's probability specified in the data space X and the probability distribution for the latent space Z $p_{data}(x)$, $p_z(z)$, respectively. The binary cross-entropy function $V(G, D)$ is often utilized in binary classification issues [16].

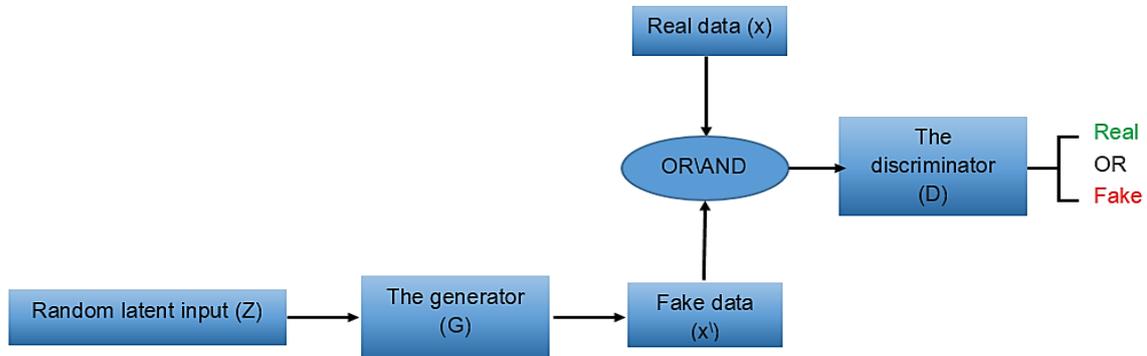


Figure 1. GAN architecture

2.2. Convolutional neural network

A CNN is a type of deep neural network that have multi hidden layers that are connected. The main advantage of CNN is to extraction feature of big data. It includes an input layer, convolutional layers, pooling also known (as the subsampling layer), and output layers [17], [18]. The feature map is generated by using the function in the convolution layer that is applied to input data, therefore the convolution layer is a mathematical linear operation between matrixes that are using the kernel. The pooling layer decreases the dimension of the extracted feature map. Many procedures are for pooling. The convolution and pooling layer represents the feature extraction of row data. The main features are extracted by repeating the convolutional and the pooling layer. After that, the extracted features are passed to the fully connected layer [19], [20]. Figure 2 presents the CNN structures.

CNN executes a series of convolutions between the filters and the input signal in its typical technique. A signal's weighted average as an input X_i is described as the convolution procedure in (2),

$$S_i = K_i * X_i \quad (2)$$

where S_i stands for the i -th feature map, and a weighting factor, also known as a filter or kernel, is denoted K_i [21].

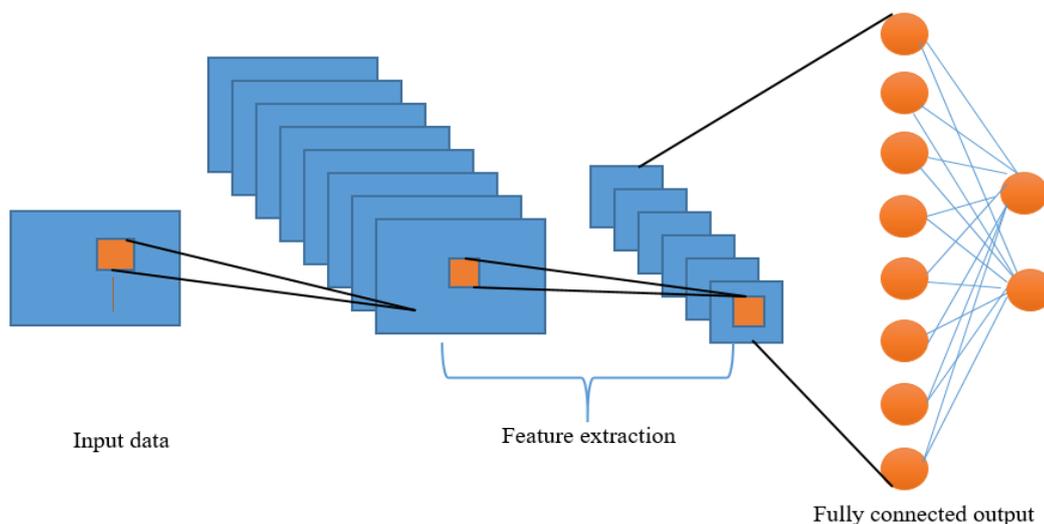


Figure 2. CNN structure

3. CASE STUDY

To improve the PdM strategy in an electromechanical system, this section presents an application of the CGAN and CNN-CGAN DL algorithms to the motor with multi-fault diagnosis. A public dataset is utilized to evaluate the proposed CNN-CGAN model. The comparison of the proposed model with the standalone CGAN model based on Precision, Recall, and F-Score metrics is introduced in this section.

3.1. Dataset

Asynchronous motor common fault (AMCF) dataset is applied for standalone CGAN and CNN-CGAN DL algorithms that is obtained from the Zenodo website. The dataset is generated by collecting the vibration signal of eight motors with one healthy data where the example of the healthy (normal) signal is shown in Figure 3(a) and seven unhealthy states where the example of the unhealthy (fault) signal is shown in Figure 3(b). It consists of 8,000 overall instances (the normal data and other seven faults) each of them has 1,000 rows with each row 1,024 of vibration signal and 1 column has the label of the fault so it is becoming $(8,000 \times 1,025)$. Table 1 shows the labels of each states [22].

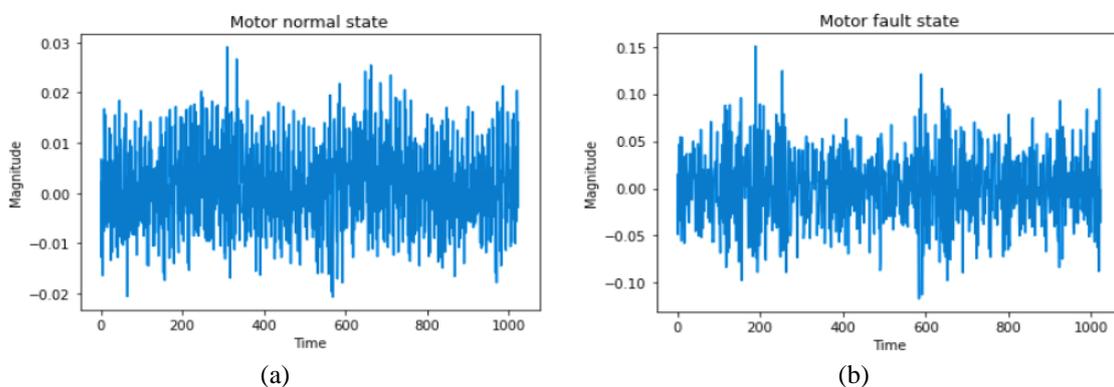


Figure 3. Vibration signal, (a) normal signal and (b) fault signal

Table 1. Types of the states of the AMCF

Label state	State descriptions
0	Normal
1	Short circuit of 2 turns (SC2T)
2	Short circuit of 4 turns (SC4T)
3	Short circuit of 8 turns (SC8T)
4	Air-gap eccentricity (AE)
5	Rotor bar broken (RBB)
6	Bearing cage broken (BCB)
7	Bearing abrasion fault (BAF)

3.2. Predictive maintenance for CGAN

The CGAN is an extension of the GAN for supervised methods. It is different from general GAN by extra information conditions (C) to manage the data production process in a supervised manner, like the class label of data that is dependent on it for the classification of multiclass faults diagnosis. The conditional (class label) is feeding into both the discriminator and generator as additional input data. The discriminator accepts samples and the information vector C to differentiate fake samples given C, while the generator takes not just a latent vector Z but also an extra information vector C. CGAN can regulate the number of Samples generated in this way, which is impossible with normal GAN [16], [23]. Figure 4 illustrates the CGAN architecture. There are different structures of CGAN such as multilayer perceptron, deep convolution neural, and autoencoder. The deep convolution CGAN (DCCGAN) makes substantial contributions to GAN by adding CNN's convolution layer and has more stability than it. The algorithm of CGAN that is used in this work is as:

- Input data.
- Split data into training data and testing data.
- Discriminator training=false.
- Generate noise (latent dim=m).
- Generated the fake data with (class label=c) in the generator network.

- Train the discriminator network with real data & (class label=c).
- Train both fake data & real data at the same time with a discriminator network.

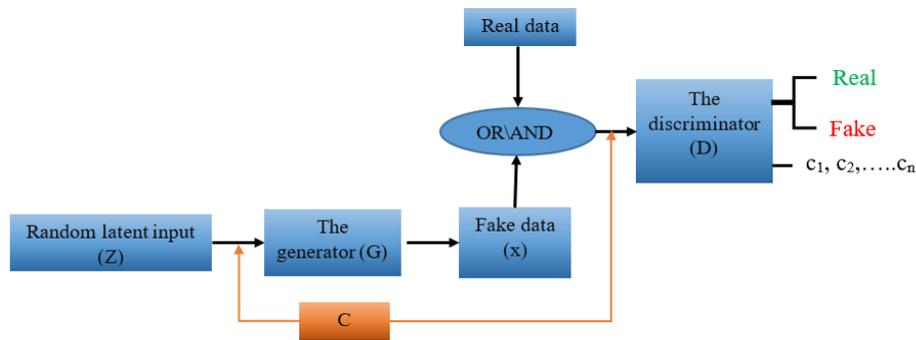


Figure 4. CGAN architectures

3.3. Predictive maintenance for CNN-CGAN

The proposed structure of the hybrid CNN-CGAN is made by adding the CNN method to CGAN. This work enhances the operation of CGAN so that it overcomes its limitations to classify the multi-label of data input. CNN strategies of deep learning are good and strong for the extraction of feature maps of row data. Therefore, the combination of two of these methods makes this model more effective and powerful for the classification of multiclass fault prediction. Figure 5 illustrates the general architecture of the CNN-CGAN hybrid model for PdM.

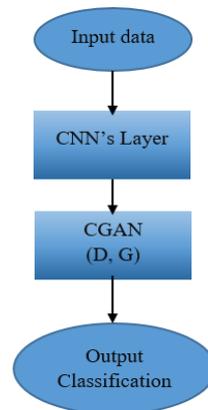


Figure 5. Architecture of the proposed CNN-CGAN model

In the D network, there are two dense layers. One for distinguishing real and fake output data contains one output which is using binary cross-entropy loss function and sigmoid nonlinear activation function. The other layer is for the classification of the eight fault classes of the motor machine, therefore the sparse_categorical_crossentropy loss function and softmax nonlinear activation function are applied to classify these faults. The optimizer in the D network is Adam.

The second network is G which is used to generate fake data. It begins in reverse order compared with the D network. The dense layer contains a 128-sized filter which uses the rectified linear unit (ReLU) as an activation function with an 8×8 window size for generating data as a result, the UpSampling2D layer doubles the input dimensions to 16×16 window size, then it is as input to a 2-dimensional convolution layer with filter size 128 and a kernel size of 3. Then a batch-normalization (BN) is applied. The UpSampling2D layer now doubles the dimensions to 32×32 feature, then passes to the last two-layer having the 128 and 1 filter size with a kernel size of 3. The binary cross-entropy loss function and (tanh) nonlinear activation function is used. The optimizer in the D network is Adam. Figure 6 illustrates the details of the proposed model for hybrid CNN-CGAN.

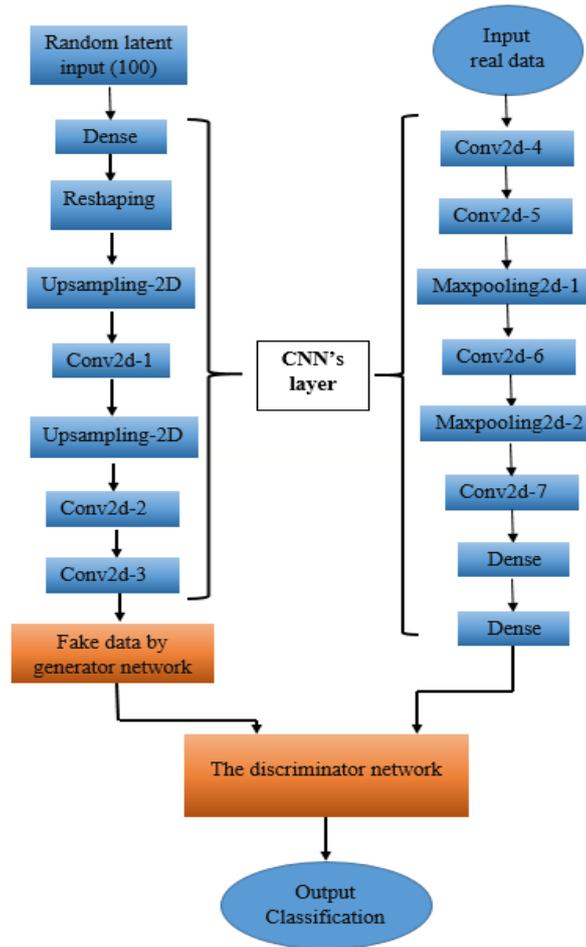


Figure 6. Description of the proposed CNN-CGAN model

3.4. Training and evaluation of the CNN-CGAN PdM model

The dataset is split into two-part the training set of 70% and the testing set of 30%. The training set is used to train the proposed CNN-CGAN model for PdM. The model is built on the TensorFlow platform [24] and uses GPU acceleration. The model's hyper-parameters are chosen based on trial and error and used the best values illustrated in Table 2.

Table 2. The D and G network hyper parameters of the CNN-CGAN model

Hyperparameter	D	G
Learning rate	0.0002	0.0002
Activation function	Leaky ReLU/softmax/sigmoid/	ReLU/ tanh
Loss function	Binary_crossentropy /sparse_categorical_crossentropy	Binary_crossentropy
Dropout	0.25	0.25
Batch normalization	8	8
Epoch	15,000	15,000
Batch size	32	32
Optimizer	Adam	Adam
No. of convolution layer	4	3
No. of layer	8	7
No. of max-pooling layer	2	—

After the CNN-CGAN model has been trained, it is assessed using the testing dataset that was previously produced. The model's performance is assessed using the model accuracy metrics Precision, Recall, and F-score as described in (3)-(5). These facts are derived from the following of the confusion matrix in Table 3,

- Precision is a metric that measures how exact a model is and can be determined as,

$$Precision = TP / (TP + FP) \tag{3}$$

- Recall is a metric that gauges the model's completeness and the calculation is,

$$Recall = TP / (TP + FN) \tag{4}$$

- F-Score is calculated by weighing Recall and Precision. It's commonly applied to assess model accuracy in situations where training dataset is unbalanced. The F-Score is given by [25],

$$F - score = (2 * Precision * Recall) / (Precision + Recall) \tag{5}$$

Where, respectively, the letters true positive (TP), false positive (FP), false negative (FN), and true negative (TN).

Table 3. Matrix of confusion

	A true prediction	A false prediction
Real true	TP	FN
Real false	FP	TN

Table 4 shows the performance metrics that were obtained using the test dataset to evaluate the CNN-CGAN PdM model. Because the dataset used to train the model has numerous classes for prediction, the evaluation metrics for each class were generated separately, and the model's overall effectiveness was calculated by averaging all of the results. The regular CGAN deep learning approach is trained and assessed with the CNN-CGAN model being applied to the same dataset to investigate the effectiveness of the proposed deep learning method for PdM. Table 5 shows the evaluation results for the CGAN PdM model. Figure 7 shows the confusion matrix of both the CNN-CGAN model as shown in Figure 7(a) and CGAN model as shown in Figure 7(b).

Table 4. Evaluation of CNN-CGAN results

Failure type	Precision	Recall	F-Score (%)
0	1.00	1.00	1.00
1	1.00	1.00	1.00
2	1.00	1.00	1.00
3	1.00	1.00	1.00
4	1.00	1.00	1.00
5	1.00	1.00	1.00
6	1.00	1.00	1.00
7	1.00	1.00	1.00
Average			1.00

Table 5. Evaluation of CGAN results

Failure type	Precision	Recall	F-Score (%)
0	1.00	1.00	1.00
1	1.00	0.99	1.00
2	0.97	1.00	0.98
3	1.00	0.98	0.99
4	1.00	1.00	1.00
5	1.00	1.00	1.00
6	1.00	0.84	0.91
7	0.86	1.00	0.93
Average			0.97625

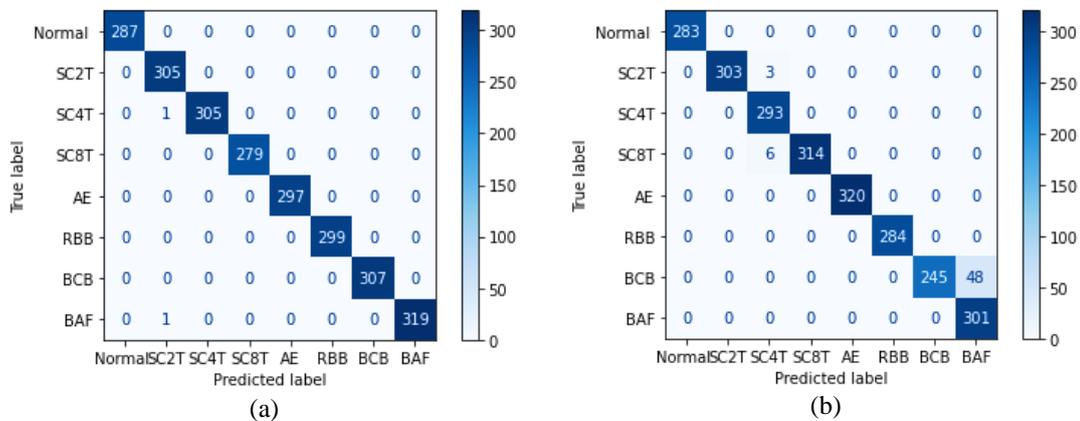


Figure 7. The confusion matrix of, (a) CNN- CGAN model and (b) CGAN model

Considering the assessment findings of the PdM CNN-CGAN model, it can be seen that the model has a high f score accuracy of 100% in predicting the motor fault state in terms of stating which component is broken, or will the machine continue to function without any problems. When the hybrid model CNN-CGAN is compared to the CGAN model for PdM, it is seen that the hybrid model improves the typical prediction accuracy of 2.375%. In terms of f score accuracy, the CGAN and hybrid model CNN-CGAN results are compared to those of comparable works. Table 6 shows the outcomes of each model. The outcomes reveal that the suggested hybrid model CNN-CGAN outperformed other relevant models in terms of fault prediction accuracy.

Table 6. PdM comparison of CNN-CGAN, CGAN, and related works

Algorithm	F-Score on average (%)
LSTM	92.45
CNN	94.24
CGAN	97.625
CNN- CGAN	100

4. CONCLUSION

The application of data-driven approaches in the field of PdM, such as DL algorithms has been made possible by the use of sensor technologies to acquire information on the status of production equipment. To construct a model in PdM of many related component production systems, a hybrid CNN-CGAN DL technique is provided in this paper. The proposed algorithm mixes the reliability of a CNN network with the ability of CGAN to classify the data as real and fake. The CNN-CGAN DL model is evaluated using an available real-world industrial motor multi_fault data. The proposed model obtains an average of 100% Precision, 100% Recall, and 100% F-Score on the testing dataset, according to the results. When compared to the normal CGAN model, the proposed CNN-CGAN model improves prediction accuracy by 2.375%. The accuracy of the hybrid model exceeds that of the real PdM works as well.

REFERENCES

- [1] S. Sajid, A. Haleem, S. Bahl, M. Javaid, T. Goyal, and M. Mittal, "Data science applications for predictive maintenance and materials science in context to Industry 4.0," *Materials Today: Proceedings*, vol. 45, pp. 4898–4905, 2021, doi: 10.1016/j.matpr.2021.01.357.
- [2] X. Bampoula, G. Siaterlis, N. Nikolakis, and K. Alexopoulos, "A deep learning model for predictive maintenance in Cyber-Physical production Systems Using LSTM autoencoders," *Sensors*, vol. 21, no. 3, p. 972, 2021, doi: 10.3390/s21030972.
- [3] S. Zhai, B. Gehring, and G. Reinhart, "Enabling predictive maintenance integrated production scheduling by operation-specific health prognostics with generative deep learning," *Journal of Manufacturing Systems*, vol. 61, pp. 830–855, 2021, doi: 10.1016/j.jmsy.2021.02.006.
- [4] C. K. M. Lee, Y. Cao, and K. H. Ng, "Big data analytics for predictive maintenance strategies." pp. 50–74, 2017, doi: 10.4018/978-1-5225-0956-1.ch004.
- [5] H. S. Abdullah and S. B. Neama, "Proposed business intelligence system through big data," *Engineering and Technology Journal*, vol. 34, 2016, doi: 10.30684/etj.34.4B.10.
- [6] A. M. Mahmood, A. Al-Yasiri, and O. Y. Alani, "Cognitive neural network delay predictor for high speed mobility in 5G C-RAN cellular networks," in *2018 IEEE 5G World Forum (5GWF)*, 2018, pp. 93–98, doi: 10.1109/5GWF.2018.8516715.
- [7] F. Zhou, S. Yang, H. Fujita, D. Chen, and C. Wen, "Deep learning fault diagnosis method based on global optimization GAN for unbalanced data," *Knowledge-Based Systems*, vol. 187, p. 104837, 2020, doi: 10.1016/j.knosys.2019.07.008.
- [8] A. Q. Albayati, A. S. Al-Araji, and S. H. Ameen, "A method of deep learning tackles sentiment analysis problem in Arabic texts," *Iraqi Journal of Computers, Communications, Control and Systems engineering*, vol. 20, no. 4, pp. 9–20, 2020, doi: 10.33103/uot.ijccce.20.4.2.
- [9] A. Khamparia and K. M. Singh, "A systematic review on deep learning architectures and applications," *Expert Systems*, vol. 36, no. 3, p. 12400, 2019, doi: 10.1111/exsy.12400.
- [10] Z. A. Mohammed, M. N. Abdullah, and I. H. Al Hussaini, "Predicting incident duration based on machine learning methods," *Iraqi Journal of Computers, Communications, Control and Systems Engineering*, vol. 21, no. 1, pp. 1–15, 2021, [Online]. Available: https://ijccce.uotechnology.edu.iq/article_168352.html.
- [11] Y. Yamato, "Study and evaluation of automatic GPU offloading method from various language applications," *International Journal of Parallel, Emergent and Distributed Systems*, vol. 37, no. 1, pp. 22–39, 2022, doi: 10.1080/17445760.2021.1971666.
- [12] A. Kusiak, "Convolutional and generative adversarial neural networks in manufacturing," *International Journal of Production Research*, vol. 58, no. 5, pp. 1594–1604, 2020, doi: 10.1080/00207543.2019.1662133.
- [13] X. Wang, X. Liu, P. Song, Y. Li, and Y. Qie, "A novel deep learning model for mechanical rotating parts fault diagnosis based on optimal transport and generative adversarial networks," *Actuators*, vol. 10, no. 7, p. 146, 2021, doi: 10.3390/act10070146.
- [14] A. Creswell, T. White, V. Dumoulin, K. Arulkumaran, B. Sengupta, and A. A. Bharath, "Generative adversarial networks: An Overview," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 53–65, 2018, doi: 10.1109/MSP.2017.2765202.
- [15] G. Ghoshel, J. Li, and T. Zhu, "A review of generative adversarial networks for electronic health records: Applications, evaluation measures and data sources," 2022.
- [16] Y. Hong, U. Hwang, J. Yoo, and S. Yoon, "How generative adversarial networks and their variants work," *ACM Computing Surveys (CSUR)*, vol. 52, no. 1, pp. 1–43, 2020, doi: 10.1145/3301282.

- [17] B. K. O. C. Alwawi and L. H. Abood, "Convolution neural network and histogram equalization for COVID-19 diagnosis system," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 24, no. 1, pp. 420–427, 2021, doi: 10.11591/IJEECS.V24.I1.PP420-427.
- [18] A. Nasser and H. Al-Khazraji, "A hybrid of convolutional neural network and long short-term memory network approach to predictive maintenance," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 12, no. 1, 2022, doi: 10.11591/ijece.v12i1.pp721-730.
- [19] J.-H. Lee, J.-H. Park, and I.-S. Lee, "Fault diagnosis of induction motor using convolutional neural network," *Applied Sciences*, vol. 9, no. 15, p. 2950, 2019, doi: 10.3390/app9152950.
- [20] S. A. H. Alazawi and M. N. Abdullah, "Online failure prediction model for open-source software system based on CNN." pp. 599–608, 2021, doi: 10.1007/978-981-16-2094-2_70.
- [21] M. Jiménez-Guarneros, J. Grande-Barreto, and J. de Jesus Rangel-Magdaleno, "Multiclass incremental learning for fault diagnosis in induction motors Using fine-tuning with a memory of exemplars and nearest centroid classifier," *Shock and Vibration*, vol. 2021, pp. 1–12, 2021, doi: 10.1155/2021/6627740.
- [22] Liang, "The motor fault diagnosis experiment dataset," 2019, doi: 10.5281/ZENODO.3553755.
- [23] J. Luo, J. Huang, and H. Li, "A case study of conditional deep convolutional generative adversarial networks in machine fault diagnosis," *Journal of Intelligent Manufacturing*, vol. 32, no. 2, pp. 407–425, 2021, doi: 10.1007/s10845-020-01579-w.
- [24] M. Abadi *et al.*, "TensorFlow: A system for large-scale machine learning," *In 12th USENIX symposium on operating systems design and implementation*, pp. 265–283, 2016, [Online]. Available: <https://tensorflow.org>.
- [25] P. K. Mallick, S. Mishra, and G.-S. Chae, "Digital media news categorization using Bernoulli document model for web content convergence," *Personal and Ubiquitous Computing*, pp. 1–16, 2020, doi: 10.1007/s00779-020-01461-9.

BIOGRAPHIES OF AUTHORS



Azhar Muneer Abood    was born in Baghdad, Iraq in 1983. She was given her B.Sc. degree in Computer and Software Engineering Department, Al-Mustansiriya University, Baghdad Iraq in 2005. Currently, she is a researcher at the University of Technology Control and Systems Engineering to get an MSc degree. She works at the University of Karbala, college of Engineering, Electrical and Electronic Eng. Department, laboratory unit. She can be contacted at email: cse.20.09@grad.uotechnology.edu.iq.



Ahmed Raof Nasser    was born in Baghdad, Iraq in 1984. He got his BSc degree from the University of Technology Control and Systems Eng. Faculty, Baghdad Iraq in 2006, MSc degree in Computer Eng. from Istanbul University, Istanbul Turkey in 2012, PhD degree from Hacettepe University, Ankara Turkey in Computer Engineering in 2018. Currently, he is a Lecturer in the Control and Systems Engineering Department at the University of Technology. His current interest is in "Data Mining", "Natural Language Processing" and "Machine and Deep Learning". He can be contacted at email: ahmed.r.nasser@uotechnology.edu.iq.



Huthaifa Al-Khazraji    was born in Baghdad, Iraq in 1984. He received his B.S degree in Control and Systems Engineering Department at the University of Technology, Iraq in 2006. And His M.S Degree in Industrial Engineering and Management at Politecnico di Torino University, Italy in 2010. From 2010 to 2014 he was Assistant Lecturer at the Control and Systems Engineering Department at the University of Technology, Iraq. He was accepted in Central Queensland University in Australia as a PhD student in March 2015. He received his PhD degree in 2019. Currently, he is a Lecturer in the Control and Systems Engineering Department at the University of Technology since 2019. His area of interest: Control Engineering, Industrial Engineering, and Optimization. He can be contacted at email: 60141@uotechnology.edu.iq.