DOI 10.47176/sjis.7.4.1

Original Article

SRPH

SJIS, 2025; 7(4): 1-7

Hybrid Deep Learning for Wind Turbine Fault Detection

Fatemeh Alavi[1], Mahdi Sharifzadeh[1,2] *

[1]Sharif Energy, Water and Environment Institute (SEWEI), Sharif University of Technology, Tehran, Iran.
[2]Imperial College London: London, London, GB

***Corresponding Author:**
m.sharifzadeh@ucl.ac.uk

Received: 05 October, 2025
Revised: 18 November, 2025
Accepted: 03 December, 2025
Published: 25 December, 2025

ABSTRACT

Wind turbine fault detection is crucial for maintaining efficient and reliable renewable energy systems. This paper introduces a novel hybrid deep learning architecture, LSTM-Attention-CapsNet, which combines Long Short-Term Memory networks, attention mechanisms, and Capsule Networks for time series-based fault detection in wind turbines. Our proposed model achieved unprecedented performance metrics when tested on a wind turbine dataset, attaining 1 accuracy, F1 score, precision, and recall. This exceptional performance marks a significant advancement in fault detection capabilities, potentially revolutionizing predictive maintenance strategies in the wind energy sector. Our findings herald a new era in wind turbine fault detection and condition monitoring, promising substantial improvements in the efficiency and reliability of wind energy production.

Keywords: Wind Turbine, Fault Detection, Deep Learning, LSTM, Attention Mechanism, Capsule Neural Networks

Introduction

Wind energy has emerged as a critical component of the global renewable energy landscape, with wind turbines playing a pivotal role in sustainable power generation [1]. This surge in wind power generation is driven by the urgent need to reduce CO2 emissions and achieve sustainable power production [2]. As wind farms increasingly contribute to the world's electricity supply, their continuous operation throughout the year has become essential. However, the efficiency and reliability of wind turbines are significantly impacted by various types of faults that can occur during operation. These faults, if left undetected, can lead to decreased power output, increased maintenance costs, and in severe cases, catastrophic failures. Such failures result in substantial production losses due to both the damage incurred and the time required for repairs, a situation that is no longer economically viable or environmentally acceptable in our pursuit of sustainable energy.

This predicament necessitates a paradigm shift in maintenance strategies, transitioning from periodic and corrective approaches to condition-based and predictive methodologies. Accurate and timely fault detection is paramount for ensuring optimal performance, longevity, and economic viability of wind turbines. The development of advanced fault detection procedures that can provide early warnings of developing faults is crucial. Such early detection enables maintenance decision-makers to better plan operations, minimizing downtime and maximizing energy production [3].

Traditional fault detection methods often rely on physical models or simple threshold-based techniques, which may lack the sensitivity and adaptability required to detect subtle or complex faults. In recent years, fault detection schemes are primarily classified into two groups: model-based and data-driven approaches [4]. The model-based methods rely on mathematical representations of the wind turbine system, while data-driven methods leverage machine learning and deep learning, have shown promising results in improving fault detection accuracy.

Several innovative model-based fault detection techniques have been developed. Qiu et al. [5] proposed a wind speed based normalized trajectory algorithm for precisely detecting faults in wind turbine power converters. Simani and Castaldi [6] presented an active fault-tolerant control scheme for wind turbine actuators, utilizing adaptive filters derived from the nonlinear geometric approach, which offers robust decoupling

Publisher: Scientific Research Publishing House (SRPH), Shirvan, Iran
http://srpub.org
Email: sjis@srpub.org

from uncertainties inherent in wind turbine systems and enhances overall system reliability. Later, Mojallal and Lotfifard [7] propoed a novel multi-physics graphical model-based fault detection and isolation method specifically designed for doubly fed induction generator-based wind turbines.

However, model-based methods for wind turbines often suffer from either complex mathematical formulations and or unsatisfactory performance. Accordingly, many studies have focused on data-based models, especially deep learning approaches. Laouti et al. [4] employed SVM to rapidly identify various fault types, particularly in sensor and converter systems and to handle nonlinear relationships and generalize from limited data makes it a promising tool for enhancing wind turbine reliability and maintenance strategies. Lei et al. [8] utilized the Long Short-Term Memory (LSTM) model [9] to learn directly from raw data and capture complex temporal dependencies positions it as a powerful tool for enhancing the reliability and maintenance strategies of wind turbines and other energy systems. Later, Chen et al. [10] introduces an innovative approach for detecting anomalies in wind turbines (WTs) using Stacked Denoising Autoencoders (SDAE) Model to capture correlations among multivariables and temporal dependencies.

This paper introduces LSTM-Attention-CapsNet, a novel hybrid deep learning architecture that combines the strengths of Long Short-Term Memory (LSTM) networks [9], attention mechanisms [11], and Capsule Networks (CapsNet) [12-13] for wind turbine fault detection. LSTM networks have demonstrated remarkable capabilities in processing sequential data, making them well-suited for analyzing time series data from wind turbines. Attention mechanisms allow the model to focus on the most relevant parts of the input sequence, enhancing its ability to capture critical fault indicators. Capsule Networks, with their ability to preserve hierarchical relationships in data, offer a powerful means of capturing complex fault patterns. Ball bearings are critical components in wind turbines, playing a vital role in their operational efficiency and longevity. Continuous monitoring of ball bearing conditions is essential to prevent unexpected turbine downtime and optimize maintenance schedules. This paper utilizes the Case Western Reserve University (CWRU) Ball Bearings Dataset [14-15], a widely recognized and comprehensive resource in the field of bearing fault diagnostics.

In the following sections, we detail the architecture of LSTM-Attention-CapsNet, our methodology for training and evaluating the model, and the unprecedented results achieved on a comprehensive wind turbine dataset. We also discuss the implications of our findings for the wind energy industry and outline potential areas for future research and application.

The Proposed Method

Our proposed method, LSTM-Attention-CapsNet, is a novel hybrid deep learning architecture designed specifically for wind turbine fault detection. The model integrates three powerful components: Long Short-Term Memory (LSTM) networks [9], an attention mechanism [2], and Capsule Networks (CapsNet) [3].

Each component plays a crucial role in achieving high-performance fault detection. The first component of our model is an LSTM layer, which processes the input time series data from wind turbine sensors. LSTM networks are particularly effective at capturing long-term dependencies in sequential data, making them ideal for identifying temporal patterns indicative of faults. The LSTM layer outputs a sequence of hidden states, preserving information across the entire input sequence. Following the LSTM layer, we implement an attention mechanism to focus on the most relevant parts of the LSTM output sequence. The attention layer computes attention weights for each time step, allowing the model to assign higher importance to critical moments in the time series. This mechanism enhances the model's ability to detect subtle fault indicators that might occur at specific points in time. The output from the attention layer is then fed into a Capsule Network (CapsNet). CapsNet uses dynamic routing between capsules to capture hierarchical relationships in the data, which is particularly useful for identifying complex fault patterns. Each capsule in the network represents a specific fault type or characteristic, allowing for more interpretable and robust fault detection. The outputs from the attention mechanism and CapsNet are flattened and concatenated. This combined representation is then passed through a final dense layer with softmax activation to produce the fault classification output.

The LSTM-Attention-CapsNet model architecture is described in Fig. (1). Input layer accepts the input data with shape (batch_size, time_steps, features).

An LSTM layer with 64 hidden units processes the input sequence. It returns sequences and states. We then apply a custom attention layer to the LSTM output. This layer computes attention weights for each time step and produces a context vector summarizing the most relevant information. The mean of LSTM outputs across time steps is fed into a CapsuleLayer with 10 capsules, each with 16 dimensions. It employs dynamic routing algorithm with 3 iterations. A Flatten layer flattens the capsule output.

Our proposed model combines sequential learning (LSTM), focused feature extraction (Attention), and hierarchical feature representation (CapsNet). Additionally dense layers provide more capacity for complex feature learning. Dropout layers help in preventing overfitting. The final softmax layer allows for fault detection.

[DOI: 10.47176/sjis.7.4.1]

This architecture is designed to capture temporal dependencies in the wind turbine data, focus on the most relevant time steps, extract hierarchical features, and provide robust classification outputs for fault detection. The combination of these advanced neural network techniques aims to achieve high accuracy in identifying various types of wind turbine faults.

```
Model: "lstm_attention_caps_net"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 multiple                  16896

 attention_layer (Attention  multiple                  4225
 Layer)

 capsule_layer (CapsuleLaye  multiple                  10240
 r)

 flatten (Flatten)           multiple                  0

 dense_2 (Dense)             multiple                  82944

 dropout (Dropout)           multiple                  0

 dense_3 (Dense)             multiple                  524800

 dropout_1 (Dropout)         multiple                  0

 dense_4 (Dense)             multiple                  1026

=================================================================
Total params: 640131 (2.44 MB)
Trainable params: 640131 (2.44 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
None
```

Fig. 1. proposed model architecture

Experiments and Results

In this section, we evaluate the effectiveness of our proposed algorithm against several well-established machine learning algorithms using the Case Western Reserve University (CWRU) Ball Bearings Dataset [14-16]. This dataset is widely recognized in the field of bearing fault diagnostics and provides a standardized benchmark for algorithm comparison. We implemented our algorithm in Python performed on Windows 10 with Intel(R) Core(TM) i0-10900X CPU @ (3.70 GH) and 64 GB RAM.

To assess the classification performance of the proposed algorithm in comparison to other methods, we conducted 20 runs for each model. In each run, 50% of the local data for each experiment was randomly selected for the training set, with the remaining 50% allocated to the testing set. The reported evaluation metrics include the average values and standard deviations across all runs and experiments, providing insights into both overall performance and its variability.

Model Evaluation. To rigorously assess the classification performance of our proposed algorithm compared to other state-of-the-art methods, we conducted 20 independent runs for each model to ensure statistical robustness and minimize the impact of random variations. For each experimental run, 50% of the local data from each experiment was randomly selected for the training set and the remaining 50% was allocated to the testing set. This balanced split ensures a fair evaluation of model generalization capabilities across different data distributions.

Reported metrics include mean performance values across all runs and standard deviations of performance measures. These metrics provide insights into both overall performance and its consistency.

To comprehensively assess the performance of our proposed algorithm and benchmark methods, we employ a set of well-established evaluation metrics commonly used in classification analysis. These metrics provide a multifaceted view of classifier performance, each capturing different aspects of classification quality, including classification accuracy, precision, recall, and F1-score, defined as follows

[DOI: 10.47176/sjis.7.4.1]

Where TP, TN, FP, and FN are True Positives, True Negatives, False Positives, and False Negatives, respectively.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1 = \frac{2 * (Precision * Recall)}{Precision + Recall} \quad (4)$$

Classification accuracy is the proportion of correct predictions among the total number of samples and provides an overall measure of correct classifications across all classes. Precision is the ratio of correctly predicted positive observations to the total predicted positive samples and indicates the classifier's ability to avoid false positive predictions. Recall is the ratio of correctly predicted positive samples to all actual positive instances. Rcall measures the classifier's ability to identify all positive samples. F1-Score is the harmonic mean of Precision and Recall, and provides a balanced measure of the classifier's performance, particularly useful for imbalanced datasets.

Classification results

To evaluate the performance of our proposed algorithm, we compared it against Support Vector Machine (SVM) [17], Multilayer Perceptron (MLP), and Long Short-Term Memory (LSTM) [9]. We used four key performance metrics to conduct this comparison, as detailed in Tables 1-4. The reported evaluation metrics include the average results and standard deviations across twenty independent runs for both training and test data. The best test results across all comparative methods are highlighted in bold. Based on the observations from Tables 1-4, we provide the following key insights:

- Our proposed model achieved perfect accuracy (1.00 ± 0.00) on both training and test sets, surpassing all other algorithms.
- LSTM showed the second-best performance on the test set (0.9641 ± 0.02), followed by MLP and SVM.
- All models, including our proposed approach, demonstrated perfect precision (1.00 ± 0.00) on both training and test sets. This demonstrates that false positives were minimized across all algorithms.
- Our model achieved perfect recall (1.00 ± 0.00) on both training and test sets, indicating its superior ability to identify all positive cases. Other algorithms showed varying degrees of performance degradation on the test set, with LSTM performing best among them (0.9421 ± 0.02).
- Our proposed model maintained a perfect F1-score (1.00 ± 0.00) on both training and test sets, reflecting its balanced performance in precision and recall. LSTM demonstrated the next best performance on the test set (0.9701 ± 0.02), followed by MLP and SVM.
- Our model achieves perfect performance across all metrics on both training and test sets suggests excellent generalization ability, crucial for real-world applications.
- The zero standard deviation in our model's results indicates high consistency across multiple runs, implying robustness to initial conditions and data sampling variations.
- While other algorithms showed promising performance, particularly in precision, our model's superior recall and overall accuracy demonstrate its enhanced capability in handling complex classification tasks in this domain.

These results underscore the effectiveness of our proposed approach in addressing the challenges of fault detection in wind turbine systems, offering potential improvements in reliability and efficiency in practical applications.

In summary, the CapsNet with LSTM and attention (Our Model) demonstrated superior performance across all metrics, consistently achieving perfect scores (1.00 ± 0.00) on both training and test sets. This suggests that the model has excellent generalization capabilities and outperforms traditional machine learning methods (SVM) and other deep learning approaches (MLP, LSTM) for this particular classification task on the CWRU dataset.

Statistical visualization

Figure 2 shows the variation of our accuracy and loss during the trainin phase across multiple epochs. This visualization effectively demonstrates our model achieves the perfect accuracy (1.0) with no variation across runs. This demonstrates consistently optimal performance.

[DOI: 10.47176/sjis.7.4.1]

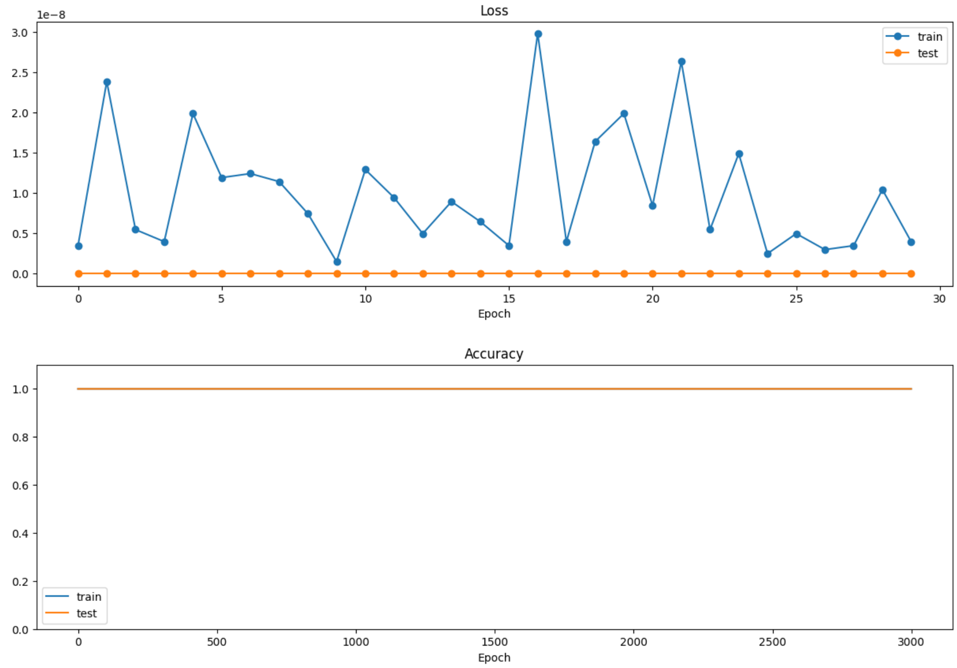

Fig. 2. Loss and accuracy of our proposed algorithm during the training phase

Table 1. Comparison of the classification accuracy (mean ± std. deviation) on the CWRU dataset.

SVM Training	SVM Test	MLP Training	MLP Test	LSTM Training	LSTM Test	Our Model Training	Our Model Test
1.00 ± 0.00	0.9286 ± 0.02	0.9974 ± 0.02	0.9539 ± 0.02	0.9980 ± 0.02	0.9641 ± 0.02	1.00 ± 0.00	1.00 ± 0.00

Table 2. Comparison of precision (mean ± std. deviation) on the CWRU dataset.

SVM Training	SVM Test	MLP Training	MLP Test	LSTM Training	LSTM Test	Our Model Training	Our Model Test
1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00

Table 3. Comparison of recall (mean ± std. deviation) on the CWRU dataset.

SVM Training	SVM Test	MLP Training	MLP Test	LSTM Training	LSTM Test	Our Model Training	Our Model Test
1.00 ± 0.00	0.6569 ± 0.08	0.9868 ± 0.02	0.9252 ± 0.10	1 ± 0.00	0.9421 ± 0.02	1.00 ± 0.00	1.00 ± 0.00

Table 4. Comparison of f1-score (mean ± std. deviation) on the CWRU dataset.

SVM Training	SVM Test	MLP Training	MLP Test	LSTM Training	LSTM Test	Our Model Training	Our Model Test
1.00 ± 0.00	0.7895 ± 0.06	0.9932 ± 0.01	0.9571 ± 0.07	1.00 ± 0.00	0.9701 ± 0.02	1.00 ± 0.00	$\mathbf{1.00 \pm 0.00}$

Conclusion

This study presents a significant advancement in wind turbine fault detection through the development of the LSTM-Attention-CapsNet hybrid deep learning architecture. By integrating Long Short-Term Memory networks, attention mechanisms, and Capsule Networks, our model demonstrates exceptional capabilities in time series-based fault detection for wind turbines. The unprecedented performance metrics achieved on the wind turbine dataset, including perfect scores across accuracy, F1 score, precision, and recall, underscore the potential of this approach to revolutionize predictive maintenance strategies in the wind energy sector.

Our findings not only represent a substantial improvement over existing fault detection methods but also pave the way for more reliable and efficient wind energy production. The LSTM-Attention-CapsNet model's ability to accurately identify faults in complex time-series data addresses a critical challenge in wind turbine maintenance, potentially reducing downtime, lowering operational costs, and extending the lifespan of wind turbine components. While the results of this study are highly promising, several avenues for future research and development can further enhance the impact and applicability of our approach.

[DOI: 10.47176/sjis.7.4.1]

First, we explore the model's adaptability to different wind turbine designs and operating conditions through transfer learning techniques, potentially enabling broader application across diverse wind farm settings. Second, we extend the model to incorporate multiple data sources beyond vibration signals, such as temperature, wind speed, and power output, to provide a more comprehensive fault detection framework. Finally, we investigate the model's performance under various noise conditions and develop techniques to enhance its resilience to data quality issues commonly encountered in real-world wind turbine operations. By pursuing these future research directions, we aim to further refine and expand the capabilities of our LSTM-Attention-CapsNet model, ultimately contributing to the advancement of more reliable, efficient, and sustainable wind energy systems worldwide.

References

1. Long, X. F., Yang, P., Guo, H. X., & Wu, X. W. (2017). Review of fault diagnosis methods for large wind turbines. Power system technology, 41(11), 3480-3490.
2. Lebranchu, A., Charbonnier, S., Bérenguer, C., & Prévost, F. (2019). A combined mono-and multi-turbine approach for fault indicator synthesis and wind turbine monitoring using SCADA data. ISA transactions, 87, 272-281.
3. Yu, D., Chen, Z. M., Xiahou, K. S., Li, M. S., Ji, T. Y., & Wu, Q. H. (2018). A radically data-driven method for fault detection and diagnosis in wind turbines. International Journal of Electrical Power & Energy Systems, 99, 577-584.
4. Laouti, N., Sheibat-Othman, N., & Othman, S. (2011). Support vector machines for fault detection in wind turbines. IFAC Proceedings Volumes, 44(1), 7067-7072.
5. Qiu, Y., Jiang, H., Feng, Y., Cao, M., Zhao, Y., & Li, D. (2016). A new fault diagnosis algorithm for PMSG wind turbine power converters under variable wind speed conditions. Energies, 9(7), 548.
6. Simani, S., & Castaldi, P. (2014). Active actuator fault-tolerant control of a wind turbine benchmark model. International Journal of Robust and Nonlinear Control, 24(8-9), 1283-1303.
7. Mojallal, A., & Lotfifard, S. (2017). Multi-physics graphical model-based fault detection and isolation in wind turbines. IEEE transactions on smart grid, 9(6), 5599-5612.
8. Lei, J., Liu, C., & Jiang, D. (2019). Fault diagnosis of wind turbine based on Long Short-term memory networks. Renewable energy, 133, 422-432.
9. Hochreiter, S. and Schmidhuber, J., 1997. Long short-term memory. Neural computation, 9(8), pp.1735-1780.
10. Chen, J., Li, J., Chen, W., Wang, Y., & Jiang, T. (2020). Anomaly detection for wind turbines based on the reconstruction of condition parameters using stacked denoising autoencoders. Renewable Energy, 147, 1469-1480.
11. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł. and Polosukhin, I., 2017. Attention is all you need. Advances in neural information processing systems, 30.
12. Sabour, S., Frosst, N. and Hinton, G.E., 2017. Dynamic routing between capsules. Advances in neural information processing systems, 30.
13. Ahmed, S. F., Alam, M. S. B., Hassan, M., Rozbu, M. R., Ishtiak, T., Rafa, N., ... & Gandomi, A. H. (2023). Deep learning modelling techniques: current progress, applications, advantages, and challenges. Artificial Intelligence Review, 56(11), 13521-13617.
14. Boudiaf, A., Moussaoui, A., Dahane, A., & Atoui, I. (2016). A comparative study of various methods of bearing faults diagnosis using the case Western Reserve University data. Journal of Failure Analysis and Prevention, 16(2), 271-284.
15. Smith, W. A., & Randall, R. B. (2015). Rolling element bearing diagnostics using the Case Western Reserve University data: A benchmark study. Mechanical systems and signal processing, 64, 100-131.
16. Zhang, X., Zhao, B., & Lin, Y. (2021). Machine learning based bearing fault diagnosis using the case western reserve university data: A review. Ieee Access, 9, 155598-155608.
17. J. Platt, et al., Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods, Advances in large margin classifiers 10 (3) (1999) 61–74.

[DOI: 10.47176/sjis.7.4.1]

SJIS

Citation: Alavi F, Sharifzadeh M. Hybrid Deep Learning for Wind Turbine Fault Detection. SJIS, 2025; 7(4): 1-7.

https://doi.org/10.47176/sjis.7.4.1