

Prototipo de Inteligencia Artificial para predicciones de daños en motores de inducción eléctrica

Artificial Intelligence prototype for damage prediction in electric induction motors

Pedro De León ^{1*}, Alexandra Nuñez ¹

¹ Grupo de Investigación en Gestión y Calidad de la Energía (GIGCE), Universidad Tecnológica de Panamá 

*Autor de correspondencia: pedro.deleon@utp.ac.pa

RESUMEN. Los motores de inducción eléctrica son los más utilizados en la actualidad para realizar procesos industriales por lo que mantenerlo en óptimas condiciones garantiza la eficiencia de estos. Mantener un plan de mantenimiento predictivo nos garantiza prevenir daños mayores y evitar parar líneas de producciones u operaciones normales en fallas comunes, las cuales se suelen dar en los cojinetes, estator y rotor. La finalidad de este proyecto es crear un prototipo utilizando una red neuronal convolucional (CNN) con aprendizaje supervisado que procesa sus capas imitando al cortex visual del ojo humano para identificar distintas características puntuales de una imagen, esta metodología está basada en el “pre- procesamiento” de datos. Con esto, surge la idea de crear una APP desde el celular para detectar las fallas de los motores por medio del espectrograma de audio recolectado con el micrófono de un dispositivo móvil para que de manera práctica se pueda hacer una predicción en cualquier instante.

Palabras clave. *App, Fallas, Mantenimiento predictivo, Motores de inducción, Red neuronal*

ABSTRACT. Electric induction motors are the most used today to carry out industrial processes, so keeping them in optimal conditions guarantees their efficiency. Maintaining a predictive maintenance plan guarantees us to prevent major damages and avoid stopping production lines or normal operations in common faults such as in bearings, stator and rotor. The purpose of this project is to create a prototype using a convolutional neural network (CNN) with supervised learning that processes its layers imitating the visual cortex of the human eye to identify different punctual characteristics of an image, this methodology is based on the "pre-processing" of data. With this, the idea arises of creating an APP from the cell phone to detect engine failures through the audio spectrogram collected with the microphone of a mobile device so that a prediction can be made in a practical way at any moment.

Keywords. *App, Failures, Predictive maintenance, Induction motors, Neural network.*

1. Introducción

Los motores de inducción eléctrica son uno de los componentes más utilizados a nivel industrial en nuestro país, mantener estos equipos en su óptimo funcionamiento garantizaría la eficiencia de los procesos, así como reduciría los costos de operación. Varios estudios han demostrado un alto rendimiento para la clasificación de fallos basado en el sonido para motores de inducción, además de realizar la medición y obtención de datos y señales, se obtengan patrones característicos de las señales de exposición a sonidos para poder obtener posibles efectos que causen desperfectos a largo o corto plazo del motor. Nuestro prototipo es una aplicación

móvil accesible que utiliza el micrófono para recolectar datos acústicos de un motor. Su función es analizar esta información y evaluarla para predecir si el motor opera con normalidad o si presenta una posible falla.

1.1 Modelo propuesto

El prototipo se desarrolló empleando una red neuronal convolucional (CNN) entrenada en la plataforma Edge Impulse, utilizando como base el dataset público MIMII, específicamente el ID-004, que contiene registros de audio en condiciones normales y anómalas de ventiladores. Las señales se transformaron en espectrogramas mediante STFT con parámetros de

Citación: P. León y A. Nuñez, “Prototipo de Inteligencia Artificial para predicciones de daños en motores de inducción eléctrica”, *Revista de I+D Tecnológico*, vol. 21, no. 2, pp. (0), 2025.

Tipo de artículo: Original. **Recibido:** 23 de abril de 2025. **Recibido con correcciones:** 22 de agosto de 2025. **Aceptado:** 22 de agosto de 2025.

DOI.

Copyright: 2025 P. León y A. Nuñez. This is an open access article under the CC BY-NC-SA 4.0 license (<https://creativecommons.org/licenses/by-nc-sa/4.0/>).

ventana y hop length adecuados, aplicando normalización y técnicas de data augmentation para mejorar la robustez frente a ruido. La arquitectura CNN incluyó capas convolucionales con filtros de pequeño tamaño, funciones de activación ReLU, operaciones de max pooling, una capa densa intermedia con dropout y salida softmax binaria (normal/falla). El entrenamiento se realizó con el optimizador Adam y validación temprana para evitar sobreajuste, dividiendo los datos en conjuntos de entrenamiento, validación y prueba de forma estratificada. Finalmente, el modelo fue desplegado en un prototipo físico con ventiladores de 60 W para validar la inferencia en condiciones reales mediante un dispositivo móvil.

2. Materiales y Métodos/ Metodología

El diseño se compone del desarrollo de una red neuronal convolucional, por lo cual dispondremos de una computadora para el desarrollo y programación de la misma, y un teléfono móvil para las pruebas.

Para el proceso de recolección de datos y verificación del entrenamiento de la red neuronal dispondremos de un modelo a escala de un ventilador con sus conexiones eléctricas correspondientes el cual necesitaremos:

- ✓ Motor de 60W (ventilador)
- ✓ Cajillas eléctricas
- ✓ Cable eléctrico 12 awg
- ✓ Caja de paso
- ✓ Tuberías eléctricas

Este proyecto estará basado en una metodología de investigación con distintos apoyos de libros, proyectos de investigación, artículos científicos; los cuales son esenciales para adquirir la base de datos de los ventiladores en estado normal y estado de falla. Por diferentes fuentes encontramos datasets con información acerca de bombas, motores, ventiladores, etc. Como mencionamos anteriormente el enfoque que le damos es por medio del aprendizaje supervisado, con la ventaja que dentro de estos datasets se encuentra la información ya clasificada del sonido de un ventilador en buen estado y uno con comportamiento anómalo. De esta forma lograremos clasificar en cuál de estos dos estados se encuentran nuestros ventiladores. La CNN se entrenará en el entorno de desarrollo para algoritmos de machine learning “Edge impulse”, para su posterior prueba en el ventilador de 60 W en comportamiento normal y provocando comportamiento anómalo.

2.1 Desarrollo de red neuronal

Una red neuronal convolucional es un tipo de red neuronal artificial que consiste en múltiples capas de filtros convolucionales de una o más dimensiones. Después de cada capa, por lo general se añade una función para realizar un mapeo causal no-lineal. Como redes de clasificación al principio se encuentra la fase de extracción de características, compuesta de neuronas convolucionales y de reducción de muestreo. Al final de la red se encuentran neuronas de perceptrón sencillas para realizar la clasificación final sobre las características extraídas.

Decidimos utilizar una red de tipo convolucional porque el objetivo es clasificar distintas fallas que podría presentar el ventilador por medio de audio, por lo que es cuestión de transformar esas ondas de sonido en un espectrograma y que la red neuronal se capaz de identificar el espectrograma como una imagen. Esta red es una de las más eficientes al momento de clasificar imágenes debido a su estructura.

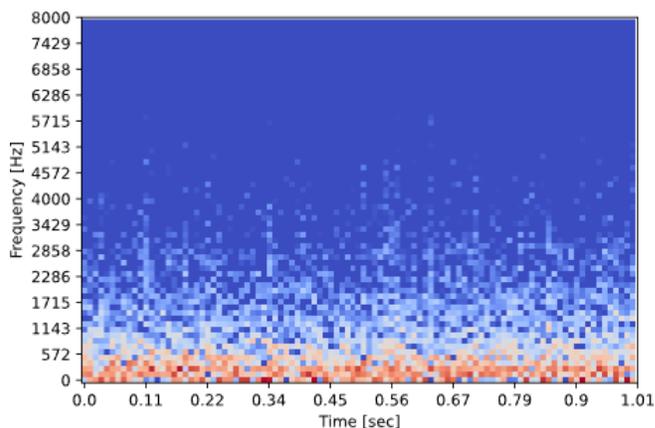


Figura 1. Espectrograma de sonido de ventilador en condición anómala.

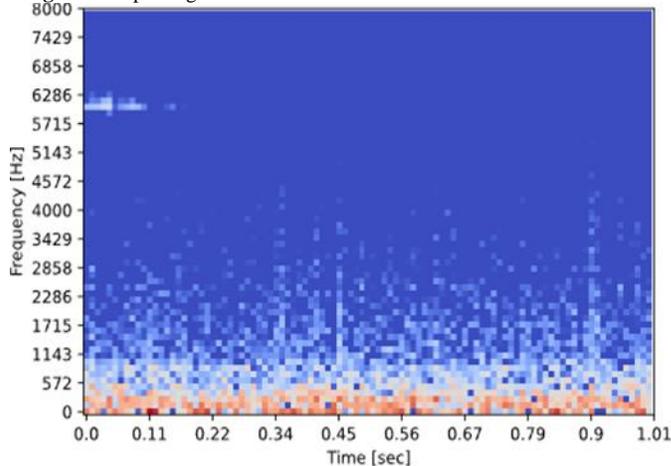


Figura 2. Espectrograma de sonido de ventilador en condición normal.

En la Figura 1, podemos observar el espectrograma resultante de un ventilador con un comportamiento anómalo y en la Figura 2, podemos el de un ventilador con un comportamiento normal. Ambos espectrogramas fueron hechos a partir de los audios del dataset [1] y procesados por el entorno de “Edge Impulse”.

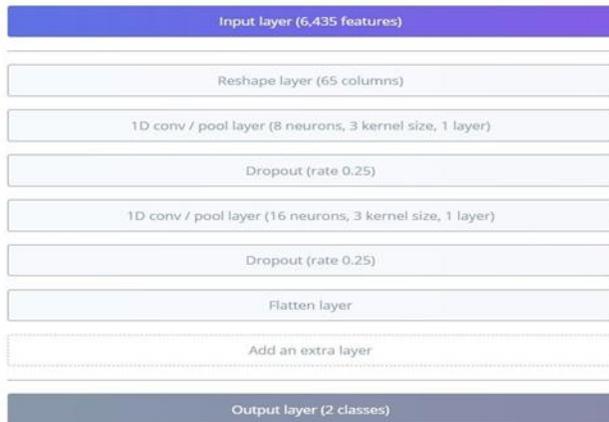


Figura 3. Estructura de red neuronal convolucional utilizada, desde Edge impulse.

En la Figura 3 se puede observar la composición de la red neuronal del prototipo donde se establece unas capas de pre- procesamiento que son los reshape que se encargan de extraer los datos más puntuales y de interés para que el procesamiento de la red sea más eficiente, después se establece dos redes convolucionales de 16 y 8 neuronas, 3 kernel y 2 capas de salida.

2.2 Programación

Edge Impulse es una plataforma para el desarrollo de algoritmos de Machine Learning de forma bastante intuitiva. Es decir, esta plataforma también nos ayuda para desarrollar algoritmos de aprendizaje máquina enfocados a implementarse en sistemas embebidos como microcontroladores o computadoras con recursos reducidos. Tiene disponibles diversas herramientas que la hacen adecuada tanto para principiantes como usuarios avanzados. En este proyecto se pondrá a prueba dicho entorno de desarrollo, “Edge Impulse”. En la Figura 4 podemos observar la estructura de nuestro algoritmo, la primera etapa consta de la data utilizada de los datasets, la segunda etapa es la del preprocesamiento de audio, para convertirlo en espectrograma, la tercera etapa es la de la red neuronal que va a tener la función de clasificar el estado de la máquina y la 4 etapa es del resultado con 2 posibles salidas (normal o posible falla). La gran

ventaja de este entorno es que permite no tener que involucrarte tanto en el código, sino en la estructura del algoritmo.

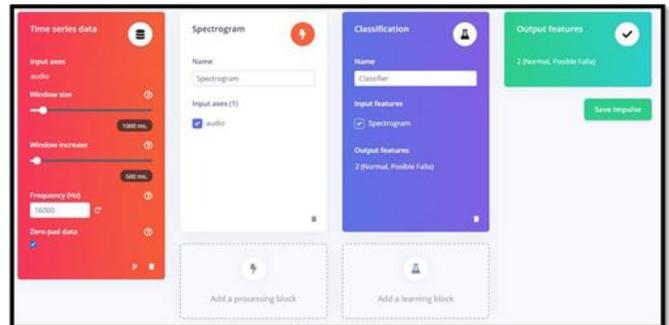


Figura 4. Interfaz de edge impulse.

2.3 Base de datos

El funcionamiento óptimo de una red neuronal depende mucho de la recopilación de datos, el ser capaz de tener una base de datos eficiente y robusta garantiza la mayor eficiencia del aprendizaje de la red, así como la capacidad de dar una respuesta correcta al momento de predecir una falla. La recolección de datos se convierte en una limitante al momento de desarrollar la red, ya que debido a las múltiples fallas posibles aumenta la cantidad de labels que la red neuronal debe saber identificar y clasificar.

El entrenamiento de nuestra red neuronal se realizó utilizando una base de datos de dominio público, MIMII Dataset: Sound Dataset for Malfunctioning Industrial Machine Investigation and Inspection [1]. Este dataset ofrece una amplia cantidad de datos y facilita la construcción de un modelo con buena capacidad de generalización. Sin embargo, es importante aclarar que existe un desacople de dominio, ya que los ventiladores registrados en MIMII no corresponden a los ventiladores propios utilizados en nuestro modelo a escala. En otras palabras, aunque la red fue entrenada con MIMII, los sonidos y condiciones reales de nuestros ventiladores pueden diferir, lo que puede afectar la generalización directa del modelo. Dentro del dataset, se evaluaron múltiples equipos identificados por ID-#, y se seleccionó el ID-004, ya que mostró los mejores resultados de aprendizaje y la mayor eficiencia al predecir fallas en nuestro prototipo.

2.4 Modelo a escala

El modelo a escala es una herramienta utilizada para poder probar el funcionamiento de nuestra red neuronal, al determinar que nuestros datos son extrapolables decidimos crear un modelo a escala con dos ventiladores de 60w, uno nuevo y el otro con meses de uso con el fin de llevar uno a posibles fallas y compararlo con el de perfecto estado para garantizar el funcionamiento de nuestra red.



Figura 5. Ventilador de 60w.



Figura 6. Maqueta de ventiladores.

3. Resultados y discusión

Después del entrenamiento, “Edge impulse” nos genera dos modelos, de los cuales podemos observar su rendimiento en la Figura 7, 8, 9 y 10. De las 3097 muestras de comportamiento normal, el modelo sin optimizar clasifico de forma correcta 2718 muestras (87.8 %) y el modelo optimizado 2915 muestras (94.1 %). De las 1072 muestras de comportamiento anómalo, el modelo sin optimizar clasifico de forma correcta 772 muestras (72.0 %) y el modelo optimizado 578 muestras (53.9%). En la Precisión general tuvieron un desempeño bastante similar. Pero en términos de Velocidad, el modelo optimizado tiene un tiempo de inferencia de aproximadamente 6.4 veces menor, uso

pico de memoria RAM de 3.2 veces menor y uso de memoria flash casi igual.



Figura 7. Valores de Accuracy, Loss y matriz de confusión del modelo sin optimizar.



Figura 8. Valores de desempeño en dispositivo del modelo sin optimizar.

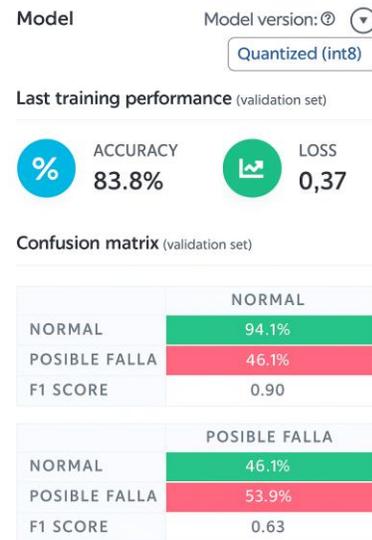


Figura 9. Valores de Accuracy, Loss y matriz de confusión del modelo optimizado.



Figura 10. Valores de desempeño en dispositivo del modelo optimizado.

El comportamiento de nuestra red neuronal fue bastante eficiente probándola con la maqueta de ventiladores. En la Figura 11 podemos observar el despliegue del algoritmo creado en “Edge Impulse” con los resultados arrojados en condición normal y condición de falla, con valores mayores a 0.90 para ambos casos.



Figura 11. Despliegue del algoritmo en un Smartphone.

En la Figura 11, se da la vista final del despliegue del algoritmo desde un celular, donde te indica si se mantiene en funcionamiento normal o posible falla y te dice la cantidad de muestras que está tomando, así como el porcentaje de inferencia por la decisión mostrada.

4. Conclusiones

ver aumento de la temperatura y de corriente, también se pueden detectar algunas fallas mecánicas por medio de observar la amplitud de los armónicos de la corriente (para el caso de la presencia de barras rotas en el rotor). A pesar de la desventaja de no poder detectar todos los tipos de falla, el prototipo creado tiene la gran ventaja de poder ser utilizado con gran facilidad, ya que hoy en día en la gran mayoría de industrias que trabajan con poseen motores (o diferentes tipos de maquinaria industrial) poseen personal con algún tipo de medio de comunicación (comúnmente un smartphone).

El plan a futuro de nuestro proyecto es crear un dispositivo que sea capaz de realizar un monitoreo continuo y no esporádico, el cual puede medir parámetros como corriente, sonido, vibraciones, voltaje y temperatura. Basado en estos parámetros crear una red de inteligencia artificial que sea capaz de analizar todos estos de manera simultánea y poder realizar una inferencia de si el motor se encuentra funcionando de manera correcta o presenta una posible falla y que sea capaz de determinar exactamente qué tipo de falla es, e informarte basado en que parámetro el realizo esta inferencia con el fin de que al momento de realizar mantenimiento o una reparación los tiempos de respuestas sean más rápidos y mejore la eficiencia de los procesos.

Iniciamos con el desarrollo del prototipo inicial utilizando un microcontrolador, Arduino Portenta H7 que sería el encargado de recolectar todos los parámetros antes mencionados de la mano de múltiples sensores conectados.

Para el desarrollo del prototipo inicial utilizaríamos:

- ✓ Portenta h7
- ✓ Portenta visión shield (micrófono)
- ✓ Transformador de corriente sct-01-30^a
- ✓ Sensor temperatura infrarrojo mlx90614
- ✓ TRANSFORMADOR DE VOLTAJE AC ZMPT101B
- ✓ Acelerómetro LSM9DS1-DOF

La finalidad de este prototipo es ver la fiabilidad y capacidad que necesitaría el dispositivo para poder analizar todos los datos de manera simultánea y poder inferir una decisión. Basado en estos datos determinaríamos si podemos hacer el procesamiento de la información en sitio o debemos usar un dispositivo alterno y mediante algún tipo de conexión de red analizar la información desde otro dispositivo. Todo esto con la idea de tener una red interna con múltiples dispositivos analizando cada motor y que una red neuronal más robusta sea capaz de tomar una decisión como encender, apagar o cambiar de motor basado en los resultados de las redes más pequeñas.

CONFLICTO DE INTERESES

Los autores declaran no tener algún conflicto de interés.

CONTRIBUCIÓN Y APROBACIÓN DE LOS AUTORES

Pedro De León: Conceptualización, Software, Investigación, Metodología, Alexandra Nuñez: Validación, Curación de datos, Redacción - Preparación del borrador original. Todos los autores afirmamos que hemos leído y aprobado la versión final de este artículo.

REFERENCIAS

- [1] Mimi Dataset: Sound Dataset for Malfunctioning Industrial Machine Investigation and Inspection. Purohit, Harsh; Tanabe, Ryo; Ichige, Kenji; Endo, Takashi; Nikaido, Yuki; Suefusa, Kaori; Kawaguchi, yoheij.
- [2] D. Bruneo y F. De Vita, "On the use of LSTM networks for predictive maintenance in smart industries", en 2019 IEEE International Conference on Smart Computing (smartcomp), 2019.
- [3] O. Aydin y S. Guldamlasioglu, "Using LSTM networks to predict engine condition on large scale data processing framework", en 2017 4th International Conference on Electrical and Electronic Engineering (ICEEE), 2017.
- [4] E. F. Reino Chérrez y J. R. Cochancela Araujo, "desarrollo de un sistema de predicción de falla de cojinete de turbina hidráulica basado en redes neuronales. Recurrentes con lstm", universidad politécnica salesiana, sede cuenca, 2022.
- [5] P. E. Gardel Sotomayor, "Aportaciones al mantenimiento predictivo de motores de inducción mediante una metodología de diagnóstico basada en el uso combinado de técnicas estadísticas y Redes Neuronales Artificiales", Universidad de Valladolid, España, 2014.
- [6] L. Islas Ortega, "red de sensores inalámbricos para mantenimiento predictivo pdm e-2019", infotec, méxico, 2020.
- [7] R. E. Marse y J. D. Schroeder, "Monitoreo, detección y diagnóstico de fallos en motores asíncronos trifásicos", Universidad Tecnológica Nacional, Facultad Regional Paraná, 2020.
- [8] O. Guerrero Castro, "Prevención de las fallas de los motores trifásicos de inducción mediante una adecuada selección", Tecnología en Marcha, vol. 23, núm. 1, pp. 78–93, jun. 2009.
- [9] D. Novoa Paradela, C. Eiras Franco, Ó. Fontenla Romero, y F. Lamas López, "Mantenimiento predictivo de motores de buques mediante aprendizaje automático".
- [10] M. A. Zamora y J. A. Jiménez, "Análisis de vibraciones en motores eléctricos de inducción para el diagnóstico de fallas," Revista Técnica Energía, vol. 15, no. 2, pp. 45–56, 2018.
- [11] A. Widodo y B. Yang, "Support vector machine in machine condition monitoring and fault diagnosis," Mechanical Systems and Signal Processing, vol. 21, no. 6, pp. 2560–2574, 2007.
- [12] A. Vogl, M. A. Aguado, y M. G. Martínez, "Aplicación de inteligencia artificial en el mantenimiento predictivo de motores de inducción," Ingeniería Mecánica, vol. 24, no. 3, pp. 66–74, 2021.
- [13] S. H. Kia, H. Henao, y G. Capolino, "Diagnosis of broken-bar fault in induction machines using discrete wavelet transform without slip estimation," IEEE Transactions on Industry Applications, vol. 45, no. 4, pp. 1395–1404, 2009.
- [14] J. Kennedy, D. Patterson, y J. McConnell, "Smartphone-based machine condition monitoring using acoustic data," Procedia CIRP, vol. 78, pp. 198–203, 2018.
- [15] S. Yin, S. X. Ding, A. H. Abadlia, y B. Zhou, "A review on basic data-driven approaches for industrial process monitoring," IEEE Trans. Ind. Electron., vol. 61, no. 11, pp. 6418–6428, 2014.
- [16] T. Ince, S. Kiranyaz, L. Eren, M. Askar, y M. Gabbouj, "Real-time motor fault detection by 1-D convolutional neural networks," IEEE Transactions on Industrial Electronics, vol. 63, no. 11, pp. 7067–7075, 2016.
- [17] R. B. Randall, Vibration-based Condition Monitoring: Industrial, Aerospace and Automotive Applications. Wiley, 2011.
- [18] Y. LeCun, Y. Bengio, y G. Hinton, "Deep learning," Nature, vol. 521, pp. 436–444, 2015.
- [19] Z. Chen, C. Li, y R. Sanchez, "Machine learning for bearing fault diagnosis: A review," Mechanical Systems and Signal Processing, vol. 138, pp. 106608, 2020.
- [20] A. B. Pimentel, "Aplicación de redes neuronales convolucionales al diagnóstico de fallas en motores de inducción," Revista Iberoamericana de Automática e Informática Industrial, vol. 17, no. 1, pp. 45–54, 2020.
- [21] S. K. Singh y P. Kumar, "Predictive maintenance of rotating machinery using deep learning techniques," en 2020 IEEE 6th International Conference on Control, Automation and Robotics (ICCAR), pp. 378–382.
- [22] J. L. González y R. Pérez, "Monitoreo de condición de motores eléctricos utilizando sensores IoT," Revista Politécnica, vol. 41, no. 2, pp. 12–20, 2019.
- [23] A. Kusiak, "Smart manufacturing must embrace big data," Nature, vol. 544, pp. 23–25, 2017.
- [24] M. He y D. He, "Deep learning based approach for bearing fault diagnosis," IEEE Transactions on Industry Applications, vol. 53, no. 3, pp. 3057–3065, 2017.
- [25] P. Rodríguez y L. Sánchez, "Uso de espectrogramas de audio para diagnóstico de fallas en motores eléctricos," Revista Colombiana de Tecnologías de Avanzada, vol. 1, no. 27, pp. 89–95, 2016.
- [26] G. Xu, M. Liu, y Z. Jiang, "Health monitoring and fault diagnosis of electric machines: A review," IEEE Trans. Ind. Electron., vol. 64, no. 4, pp. 3940–3955, 2017.
- [27] A. de Almeida, P. Fonseca, y J. Delgado, "Electric motors performance and efficiency monitoring using portable devices," Energy Reports, vol. 6, pp. 303–309, 2020.
- [28] M. B. Prieto, G. Cirrincione, A. Espinosa, J. Ortega, y H. Henao, "Bearing fault detection by a novel condition-monitoring scheme based on statistical-time features and neural networks," IEEE

Transactions on Industrial Electronics, vol. 60, no. 8, pp. 3398–3407, 2013.

[29] J. Arcos-Avilés, J. Orozco, y R. Rodríguez, “Aplicación de Edge Computing en mantenimiento predictivo industrial,” *Revista Española de Electrónica Industrial*, vol. 12, no. 2, pp. 45–54, 2022.

[30] S. Sun, J. Liang, y H. Zheng, “Data-driven approaches for intelligent fault diagnosis of rotating machinery: A review,” *IEEE Access*, vol. 8, pp. 128674–128692, 2020.