Unbalance Detection of a Rotating Shaft Using Vibration Data

Complex Engineering Problem

191203 Ahmad Mujtaba
BCE

191227 Muhammad Talha
BCE

191241 Muhammad Ali Tahir
BCE

***Abstract*—Fault detection at rotating machinery using vibration sensors allows for early diagnosis of machine damage and the prevention of production downtime by taking suitable steps. The vibration data was analyzed in this research utilizing machine learning and deep learning approaches. In this case, we used an online dataset as the foundation for developing and testing imbalance detection methods. Random forest and 1D CNN were utilized as algorithms, and their prediction accuracy was 99% and 82%, respectively.**

***Index Terms*—machine learning, deep learning, sensors**

I. Introduction

Rotating machinery, such as turbines, motors, and engines, is essential in a wide range of industrial applications. Unbalance, on the other hand, can have serious consequences in rotating systems, such as increased mechanical stress, decreased efficiency, and potential failure. As a result, detecting and diagnosing unbalance in rotating shafts is critical for ensuring safe and reliable operation [1]. It is possible to identify and quantify various types of faults, including unbalance, by analyzing the vibration patterns of a rotating shaft. Unbalance occurs when a rotating component's center of mass deviates from its intended position, resulting in uneven mass distribution and vibration during operation. The purpose of this report is to look into the identification of unbalance in a rotating shaft using vibration data [2]. We will use a dataset obtained from Kaggle, a popular online data science and machine learning platform.

A. Dataset Explanation

The dataset is made up of vibration signals captured from a rotating shaft setup. Each data sample is made up of a time series of vibration measurements taken at regular intervals while the rotating system is running. The dataset includes several features, such as:

- **V_in:** The voltage indicates the input voltage of the controller.
- **Vibration:** The vibration signals captured by sensors placed on the rotating shaft.
- **Operating Parameters:** Additional parameters related to the operating conditions, such as rotational speed, load, and temperature.

After analyzing the dataset using a correlation matrix shown in Figure 1 we concluded that V_in does not play a significant in model training so we dropped that feature. The dataset also includes labeled instances indicating the presence or absence of unbalance in the rotating shaft to aid in the analysis and evaluation of unbalance detection methods. We chose these labels based on rotational strength. We will have performed normalization, pre-processing, and EDA on the combined dataset of different unbalanced strengths in this report. Finally, for training and evaluation, we used one machine-learning model and one deep-learning model.

II. Methodology

The Methodology we have followed is that we have used EDA for feature engineering after that we used Random Forest Classifier for the machine learning model and 1D-CNN for our deep learning model.

A. Feature Engineering

Before training the models we used a correlation matrix for feature selection in which we selected only:

- **Measured_RPM**: This variable represents the measured RPM (Rotations Per Minute) of the rotating shaft.
- **Vibration_1**: This variable corresponds to the first vibration signal captured by a sensor placed on the rotating shaft.
- **Vibration_2**: This variable represents the second vibration signal captured by a sensor placed on the rotating shaft.
- **Vibration_3**: This variable denotes the third vibration signal captured by a sensor placed on the rotating shaft.

The correlation matrix for our data can be given as:

B. Random Forest Classifier

To detect unbalance in a rotating shaft using vibration data, we employ a Random Forest classifier, which is a popular machine learning algorithm known for its robustness and ability to handle complex datasets. This classifier is an ensemble model that combines multiple decision trees to make predictions. The ensemble nature of Random Forests allows for better generalization and reduces the risk of overfitting compared to individual decision trees. We create an instance of the Random Forest classifier with `n_estimators=100`, indicating the number of decision trees in the ensemble, and `random_state=42` for reproducibility. Next, we train the Random Forest classifier using the training data. Once the

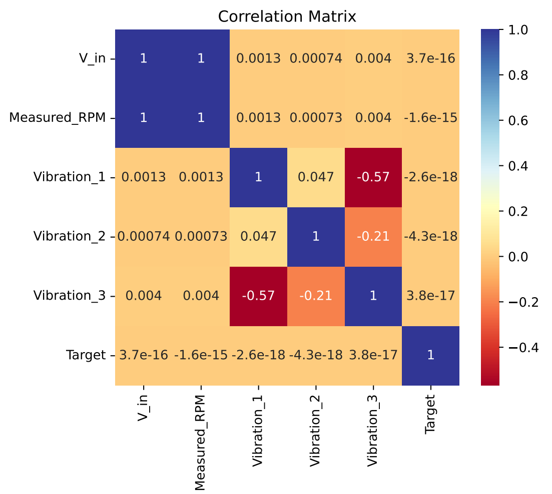

Fig. 1. Correlation Matrix.

model is trained, we apply it to make predictions on the testing data. Additionally, we generate a confusion matrix to gain insights into the classification results. The confusion matrix shows the number of true positive, true negative, false positive, and false negative predictions.

C. 1D CNN

To build a predictive model for unbalanced detection using vibration data, we employ a Convolutional Neural Network (CNN) implemented with the Keras library. CNNs have shown excellent performance in processing sequential data, making them suitable for analyzing time-series vibration signals. The following is an overview of the model architecture:

- **Convolutional Layer:** The model starts with a convolutional layer that consists of 128 filters and a kernel size of 1. This layer is responsible for extracting relevant features from the input vibration data.
- **MaxPooling Layer:** A max-pooling layer with a pool size of 2 and stride of 2 follows the convolutional layer. It reduces the spatial dimensions of the extracted features while preserving the most important information.
- **Flatten Layer:** The output from the max-pooling layer is flattened to convert the two-dimensional representation into a one-dimensional vector.
- **Dense Layers:** The flattened features are passed through multiple dense layers to capture high-level representations and learn complex relationships. The model consists of two dense layers with 128 and 64 units, respectively, both utilizing the Rectified Linear Unit (ReLU) activation function.
- **Output Layer:** The final dense layer has 5 units and uses the softmax activation function to produce probabilistic outputs across the 5 possible unbalance classes.

III. RESULTS

We have created results based on the accuracy, the confusion matrix, and the classification report for both models.

A. Random Forests

Figure 2 shows us the confusion matrix after training and testing.

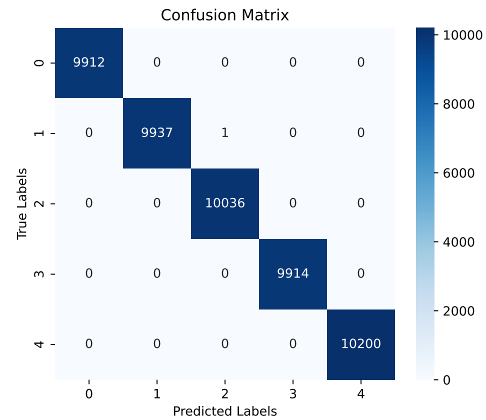

Fig. 2. Confusion Matrix of Random Forest Classifier

Figure 3 shows us the classification report in which we can see the values of accuracy, F1 score, precision, recall, and support.

```
Classification Report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00      9912
           1       1.00      1.00      1.00      9938
           2       1.00      1.00      1.00     10036
           3       1.00      1.00      1.00      9914
           4       1.00      1.00      1.00     10200

    accuracy                           1.00     50000
   macro avg       1.00      1.00      1.00     50000
weighted avg       1.00      1.00      1.00     50000
```

Fig. 3. Classification Report of Random Forest Classifier

B. 1D CNN

Figure 4 shows us the confusion matrix after training and testing.

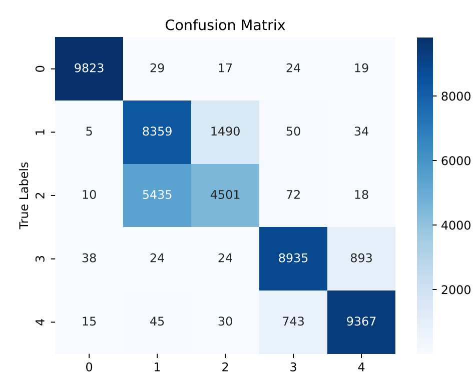

Fig. 4. Confusion Matrix of 1D CNN Classifier

Figure 5 shows us the classification report in which we can see the values of accuracy, F1 score, precision, recall, and support.

```
1563/1563 [==============================] - 3s 2ms/step
              precision    recall  f1-score   support

           0       0.99      0.99      0.99      9912
           1       0.60      0.84      0.70      9938
           2       0.74      0.45      0.56     10036
           3       0.91      0.90      0.91      9914
           4       0.91      0.92      0.91     10200

    accuracy                           0.82     50000
   macro avg       0.83      0.82      0.81     50000
weighted avg       0.83      0.82      0.81     50000
```

Fig. 5. Classification Report of 1D CNN Classifier

IV. Conclusion

A dataset of vibration data for the categorization of imbalance on a rotating shaft with changing speed and unbalanced strength was developed for this investigation. We were unable to use all of the samples due to GPU power limitations, therefore we only used 50000 samples per target variable. Several ways to solve the related classification challenge were examined. We employed random forest and 1D CNN to classify the uneven strength and achieved astounding accuracy of 99% and 82%, respectively. The model performed poorly when tested with unknown data due to a lack of training data. In future studies, this behavior might be utilized by developing a more powerful and realistic model that uses more data to improve forecast accuracy.

References

[1] Saleem, M. A., Diwakar, G., Satyanarayana, M. R. S. (2012). Detection of unbalance in rotating machines using shaft deflection measurement during its operation. IOSR J. Mech. Civ. Eng, 3(3), 08-20.

[2] Gangsar, P., Pandey, R. K., Chouksey, M. (2021). Unbalance detection in rotating machinery based on support vector machine using time and frequency domain vibration features. Noise Vibration Worldwide, 52(4-5), 75-85.