Optimizing End-of-Line Product Testing through AI: A Case Study

Athina Tsanousa[1], Evangelos Bektsis[1], Ilias Gialampoukidis[1], Stefanos Vrochidis[1] and Ioannis Kompatsiaris [1]

[1] *Information Technologies Institute, Center for Research and Technology Hellas, 6th km Charilaou-Thermis, Thessaloniki, Greece*

Abstract

Smart manufacturing enables industries to simplify and improve their procedures, resulting in less damaged products and more sustainable operations. One common industrial process is End of Line testing, which tests the final functionality of a produced product. This work provides an exhaustive comparison among various AI approaches in classifying products of EOL testing, evaluating how they respond to the characteristics of such data, including variability in values and imbalanced samples. The conducted experiments are related to the use case of Whirlpool's pilot, during the i4Q project, where it is examined whether certain sensors are related to a faulty produced part.

Keywords

End-of-Line testing, smart manufacturing, machine learning, ensemble learning

1. Introduction

End of Line (EoL) testing is a process conducted at the conclusion of the manufacturing or assembly process, evaluating the functionality of the final product [1]. The product is tested against specific characteristics, varying among different industries, and its functionality is measured through different methods, including sensors. When sensor values are out of specific limits the product is characterized as faulty. This identification process can be reframed as a classification problem where the objective is to categorize each manufactured item into one of two classes: "faulty" or "non-faulty."

In the context of EoL testing in industrial manufacturing, data acquisition presents a multifaceted set of challenges that can impact both the reliability and utility of the derived insights. Data acquisition in End-of-Line testing is full of complications ranging from the integrity and uniformity of data to security concerns. However, through the application of various techniques including data preprocessing, compression, encryption, and machine learning, some of these challenges can be addressed, enabling more accurate and efficient identification of defects and enhancing overall product quality.

In traditional manufacturing environments, the EoL tests might involve manual inspections or specific machinery designed to test particular attributes of the product. However, with the advent of data science techniques, this can now be approached as a machine learning classification task. The attributes or features extracted from the product during EoL tests can serve as the input variables to the classification model. Converting this fault diagnosis into a classification problem offers several advantages. First, it allows for automation, thereby potentially reducing the time and human resource investment required for EoL testing. Second, it opens up the possibility of employing a range of advanced machine learning techniques like late fusion models, which could improve the

Proceedings Acronym: I-ESA 2024 12th International Conference on Interoperability for Enterprise Systems and Applications, April 10–12th, 2024, Crete, Greece

atsan@iti.gr (A. Tsanousa); evanbekt@iti.gr (E. Bektsis); heliasgj@iti.gr (I.Gialampoukidis) ; stefanos@iti.gr (S. Vrochidis); ikom@iti.gr (I. Kompatsiaris)

0000-0001-6599-4446 (A. Tsanousa); 0000-0002-6760-1498 (E. Bektsis); 0000-0002-5234-9795 (I. Gialampoukidis); 0000-0002-2505-9178 (S. Vrochidis); 0000-0001-6447-9020 (I. Kompatsiaris)

accuracy of the fault detection process. Third, it facilitates real-time or near-real-time analysis, enabling immediate corrective actions if necessary.

In the present work standard classification algorithms are compared with late fusion methods in order to assess which approach performs better in predicting faulty products from sensors used in the EoL testing.

2. Related work

Applications focused on EoL testing data are limited in bibliography. Although the methodologies are not that different from other applications of smart manufacturing, certain characteristics of the EOL data need to be considered. The authors of [1] investigate the efficiency of several methods in dealing with analytical challenges such as small sample size, imbalanced data, heterogeneous products and overlapping samples of EoL data. They applied several existing ensemble algorithms and preprocessing techniques and also sampling techniques for imbalanced data. The major contribution of this paper is a thorough comparative study of these methods to identify whether they are able to cope with the analytical challenges of EoL data. They investigate how classification techniques may help quality engineers to determine the cause of a quality issue by automatically recommending the most likely faulty components, which is a challenging classification problem.

Since the industrial monitoring procedures require multiple data sources to extract a decision, fusion is often applied in relevant literature. The authors of [2] proposed a new multi-task multi-sensor fusion network (M2FN) to improve fault diagnosis performance. The proposed method first uses convolutional neural networks (CNN) to extract and fuse features from raw vibration and current signals. After that, to improve the discriminative ability of the learned features, a multi-task learning module (MTL) is designed which contains a classification task and a deep metric learning task. Instead of using one single classification task for fault diagnosis, they design a multi-task learning module to jointly optimize feature learning and classification, including one classification task with the cross-entropy (CE) loss function and one deep metric learning task with the center loss function. This MTL module can effectively improve the discriminative capability of the learned features.

Feature level fusion is also employed in [3] with the use of neural network (NN), where the authors extract features from the sensors using multiple generators and they embed the features in a latent space. Following a procedure that includes adversarial learning, the fused features are fed in the classifier. The system proposed in [4] consists of both early and late fusion modules. The authors propose a multimodal deep learning-based fault detection model that combines tabular and time series data. They exploit deep learning methods (Multilayer Perceptron (MLP), CNN, gated recurrent unit) that are combined in an early fusion level and then apply a late fusion module.

3. Methodology

This section describes the methodology applied, the dataset used for the experiments and the results. The EoL dataset was used in the respective Whirlpool pilot of the i4Q project. The use case implemented in this pilot, specifically regarding the EoL dataset, is to replace a physical test by a virtual one performed by an AI (Artificial Intelligence) enabled set of i4Q solutions. In this context, machine and deep learning algorithms were applied in the EoL data, to predict faulty cases and the best one in terms of performance and ease of applicability, was integrated in the pilot's infrastructure. In this paper however, a more research orientated approach is presented, showcasing the comparative analysis of multiple late fusion methods that were tested on the EoL data with standard classifiers.

3.1. Dataset

The data collection for this dataset took place over the course of several years within the controlled environment of a factory lab, ensuring that measurements were acquired under standardized conditions. A set of sensors were strategically placed on the dishwashing machines to capture essential parameters, such as water usage, temperature, and power consumption. These sensors played a pivotal role in monitoring the performance of the dishwashers. The primary

objective of this dataset was to assess and classify the performance of the dishwashers as either functioning correctly or encountering issues. Therefore, the dataset's target variable was binary, with one class indicating that the dishwasher was working correctly and the other representing instances where issues were detected during the test cycles. More information regarding the data cannot be disclosed due to privacy issues.

In industries like manufacturing, it is a common phenomenon to have a substantial majority of products performing as intended (in this case, dishwashers working correctly) while a smaller fraction may exhibit faults or issues, leading to a significant imbalance in the dataset. This poses a challenge for machine learning models, as they tend to be biased towards predicting the majority class due to its prevalence. To tackle the class imbalance effectively, two key strategies were employed: Cost-Sensitive Learning: Recognizing the significance of correctly identifying defective products, cost-sensitive learning techniques were applied. These techniques assign different misclassification costs to each class, emphasizing the importance of accurately identifying the minority class. This approach ensures that the model is penalized more for misclassifying defective products, thus encouraging better performance on the minority class. Additionally, a balanced sampling strategy was adopted to mitigate the effects of class imbalance. Instead of relying solely on the majority class samples, an equal or similar ratio of defective (minority) and non-defective (majority) products was included in the dataset used for training and evaluation. This balanced approach allowed the model to learn from both classes effectively, improving its ability to make accurate predictions for defective products.

3.2. Data preprocessing

Following data collection, a rigorous data preprocessing phase was executed to ensure data quality and consistency. The dataset underwent data cleaning, which involved addressing missing values and correcting errors. Missing data were handled using statistical imputation or flagged for further review when imputation wasn't reliable. Erroneous values or outliers were identified and rectified to maintain data integrity. Data normalization was applied to standardize measurements across different features, preventing variable scales from biasing analyses by applying Min-Max scaling.

As part of the experimentation process, a crucial step involved generating a representative sample from the original dataset. It's essential to clarify that this sample selection process was not driven by chronological considerations; instead, it aimed to ensure a diverse representation of data points from various time points. The objective was to create a sample that effectively captured the dataset's inherent patterns and characteristics, irrespective of their temporal sequence. This approach was instrumental in framing the problem as a classification task. To achieve this, a random sample was drawn from the dataset, encompassing data points from different temporal instances.

It's worth noting that the original dataset was substantial in size, with an initial size of 2.3 gigabytes. The preprocessing steps mentioned above, significantly impacted the dataset's size. The resulting dataset, after preprocessing and before sampling, was still substantial in volume (approximately 800MB).

3.3. Experiments and Results

This section describes the two approaches adopted. First one refers to classifiers applied to the whole set of available sensors, concatenated, while the second approach includes classifiers applied to each individual sensor and then combining the results with late fusion methods.

3.3.1. Standard classification approach

The choice of classifiers for this experimentation was driven by the need to assess their effectiveness in classifying dishwasher performance based on the dataset. Here is a brief overview of the classifiers used: Light Gradient Boosting Machine (LGBM) is a gradient boosting framework known for its efficiency and speed. It's particularly well-suited for large datasets and can handle complex relationships in the data. XGBoost (eXtreme Gradient Boosting) is another gradient

boosting algorithm with strong predictive performance, often considered a benchmark in machine learning competitions. MLP is a type of artificial neural network known for its ability to capture complex patterns in data. It's a versatile choice for classification tasks but may require careful tuning. Decision trees (DT) are interpretable models that partition the data based on feature values. They are known for their simplicity and ease of understanding. TabNet is a relatively new interpretable deep learning model designed for tabular data. It combines elements of decision trees and deep learning. Random Forest is an ensemble method that combines multiple decision trees. It's known for its robustness and ability to handle noisy data. SVM (Support Vector Machines) is a powerful classifier that can handle both linear and non-linear data. The non-linear kernel version is used to capture complex relationships in the data. The linear kernel version of SVM is suitable for problems where the data is linearly separable.

The experimentation process aimed to evaluate the performance of each classifier on the entire dataset. After data preprocessing, each classifier was trained on a subset of the dataset, using a portion of the data for training and the rest for testing. This allowed us to assess their performance in terms of training accuracy and generalization to unseen data (test accuracy). A noteworthy addition to the experimentation process was the use of grid search for hyperparameter tuning. Grid search involved systematically exploring a range of hyperparameter values for each classifier to identify the combination that yielded the best performance. This hyperparameter optimization step was essential to fine-tune the models for optimal accuracy and generalization.

Table 1 provides an overview of the results achieved by each classifier, revealing the superiority of Random Forests, while boosting classifiers like LGBM and XGBOOST performed quite well too.

Table 1
Classifier Results

Models	Train Accuracy	Test Accuracy
LGBM	94.2%	94.1%
XGBOOST	94.1%	93.6%
MLP	66%	65.7%
DT	95.8%	92.7%
RF	96.7%	95.4%
SVM NON LINEAR	51.4%	51.4%
SVM LINEAR	51.4%	51.4%
TabNet	64.8%	64.6%

3.3.2. Late Fusion approach

This analysis provides insights into how different classifiers like Logistic Regression, K-Nearest Neighbors, Naive Bayes, Decision Tree, and Support Vector Classifier perform across features such as voltage, current, power, phase, temperature, ambient temperature, and water and if applying fusion methods to combine improves the final performance metrics. Table 2 provides an overview of the classifier results before any fusion method is applied. This table provides a clear comparison of the best performing models for each feature in terms of test accuracy and balanced test accuracy.

This comprehensive evaluation of classifiers across different features is essential in smart manufacturing EoL testing. It reveals significant variations in performance based on the feature, with models like Logistic Regression, Naive Bayes (NB), and Decision Tree often showing a more balanced performance, whereas K-Nearest Neighbors (KNN), despite high accuracy, frequently struggles with balanced accuracy. The consistently low performance of the Support Vector Classifier across all features suggests its limited applicability in this context. These insights are

fundamental in understanding the strengths and weaknesses of each classifier and form a basis for the subsequent selection of appropriate models and fusion methods for optimizing EoL testing processes.

Table 2
Individual results before fusion

Feature	Best Test Accuracy Model	Test Accuracy	Best Balanced Test Accuracy Model	Balanced Test Accuracy
Voltage	KNN	96.03%	Logistic Regression	51.84%
Current	Decision Tree	55.54%	Decision Tree	56.70%
Power	KNN	96.50%	Logistic Regression	53.53%
Phase	Decision Tree	60.59%	Decision Tree	56.70%
Temperature	KNN	95.62%	Decision Tree	56.70%
Ambient Temperature	Decision Tree	63.90%	Decision Tree	61.81%
Water	KNN	84.81%	Decision Tree	65.36%

The results of the classification obtained from the individual sensors, were afterwards combined with late fusion methods. The purpose of this approach was to examine whether more information could be extracted from the data in this way.

The late fusion methods adopted were the following: The Simple Average Fusion method combines predictions from multiple models by calculating their average. It is straightforward and effective in reducing model variance but may struggle with imbalanced datasets or conflicting model predictions. Max (maximum) Voting Fusion chooses the class with the most votes from different models. It uses the strength of each model and works well when the models are different from each other. However, it can be less effective if all models make similar errors. Min (Minimum) Voting Fusion, the opposite of Max voting, takes the least common prediction as the final outcome. This can be useful in avoiding commonly occurring classes but may be unreliable if the less common classes do not lead to accurate predictions. Majority Voting Fusion requires most models to agree on the class, selecting the class with the majority of votes. This method balances out extreme predictions and works well when models complement each other. Its effectiveness might decrease if there is no clear majority or if the models are imbalanced. Ensemble Averaging Fusion averages the outputs from different models, combining their strengths to improve accuracy. This method is effective in handling varied data characteristics and reducing biases from individual models. However, it is more complex to implement and needs a careful selection of models to be averaged. It computes a weighted average of predictions, where weights are typically determined based on the individual performance of each classifier.

The performance of these models is assessed using metrics such as train/test accuracy and balanced train/test accuracy (Table 3). Logistic Regression showed varied performance across different fusion methods, with ensemble averaging showing more balanced accuracy and F1 scores. This suggests its effectiveness in handling class imbalances, a key issue for datasets in manufacturing. K-Nearest Neighbors maintained high train and test accuracy across different fusion methods but had lower balanced accuracy. This indicates potential issues with data imbalance, important in the accuracy-focused environment of EoL testing. Naive Bayes varied in performance, with ensemble averaging showing better balanced accuracy. This variation suggests the model's suitability for datasets with inherent biases, often found in the manufacturing sector. Decision Tree showed consistent performance across all fusion methods, with balanced accuracy and F1 scores

indicating its effective management of both classes in the dataset, crucial for EoL testing applications. Table 3 provides a clear and concise overview of the best performing fusion methods for each model in terms of test accuracy and balanced test accuracy.

Table 3
Results of fusion methods

Model	Best Fusion Method (Test Accuracy)	Test Accuracy	Best Fusion Method (Balanced Test Accuracy)	Balanced Test Accuracy
Logistic Regression	Min. voting	96.74%	Ensemble averaging	59.00%
KNN	Simple average	97.38%	Max voting	61.88%
Naive Bayes	Min. voting	97.01%	Ensemble averaging	58.30%
Decision Tree	Simple average	63.24%	Simple average	65.49%

Comparing the results of Table 3 with the ones presented in Table 1, it is evident that there is a small improvement in performance when using late fusion, instead of applying classifiers to all sensors.

4. Conclusions

This paper presented the experiments related to a pilot use case of the i4Q project. The pilot refers to a white goods company that, through i4Q, intends to replace the lab tests of EoL testing with virtual ones, based on AI predictive algorithms. The paper is not solely focused on the implementation for the pilot, but on the research work done to investigate whether a late fusion method could outperform the other methods.

The exhaustive experiments applied to these data revealed that although late fusion employs advanced methods to combine information from multiple sources, there is a small improvement in performance compared to the classifiers applied to concatenated data. This has been observed in other applications with real data, based on our experience. The choice for a late fusion framework over one classifier, depends not only on performance metrics but other factors, such as ease of integration and deployment and time effectiveness.

Acknowledgements

This work was supported by the i4Q project, funded by the European Union's Horizon 2020 Research and Innovation Program under Grant Agreement No. 958205.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

1. V. Hirsch, P. Reimann and B. Mitschang, "Data-Driven Fault Diagnosis in End-of-Line Testing of Complex Products," 2019 IEEE International Conference on Data Science and Advanced Analytics (DSAA), Washington, DC, USA, 2019, pp. 492-503, doi: 10.1109/DSAA.2019.00064.
2. Cui, J., Xie, P., Wang, X., Wang, J., He, Q., & Jiang, G. (2022). M2FN: An end-to-end multi-task and multi-sensor fusion network for intelligent fault diagnosis. Measurement, 204, 112085.
3. Ren, X., Wang, B., Qin, Y., & Jia, L. (2022, October). Adversarial Embedding Fusion Network for Multi-sensor Fusion Fault Diagnosis of Wheelset Bearings. In 2022 Global Reliability and Prognostics and Health Management (PHM-Yantai) (pp. 1-7). IEEE.
4. Kim, G., Choi, J. G., Ku, M., Cho, H., & Lim, S. (2021). A multimodal deep learning-based fault detection model for a plastic injection molding process. IEEE Access, 9, 132455-132467.