International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161

Available at http://inpressco.com/category/ijcet

Research Article

Scalable Machine Learning Pipelines for Big Telemetry Data in Semiconductor Manufacturing

Ivan Martis*

Independent Researcher

Received 01 July 2025, Accepted 20 July 2025, Available online 21 July 2025, **Vol.15, No.4 (July/Aug 2025)**

Abstract

The semiconductor industry faces increasing challenges in maintaining high yields and reducing costs as manufacturing processes become more complex. A new and effective tool for optimising processes is big data analytics, enabling manufacturers to extract valuable insights from vast amounts of production data and make data-driven decisions. This study proposes a comprehensive machine learning (ML) pipeline tailored for analyzing telemetry data using the SECOM dataset from the UCI repository. The methodology includes data cleaning, missing value imputation, feature scaling via Min-Max normalization, dimensionality reduction, and Synthetic Minority Oversampling Technique (SMOTE) to handle class imbalance. A Decision Tree Classifier (DTC) is utilized to classify good and defective products, achieving an accuracy of 88% in addition to excellent results in terms of recall, F1-score, ROC-AUC, and accuracy. Based on a comparison, the offered DTC model performs much better than popular traditional and deep learning techniques and can be trusted for spotting and addressing faults in real life.

Keywords: *Telemetry Data, Semiconductor Manufacturing, Fault Detection, Classification Methodology, High-Dimensional Data, Imbalanced Data, SECOM Dataset, SMOTE, Proactive Maintenance, Production Analysis.*

Introduction

Beginning in the early 1990s, the introduction of the CIM systems has enabled factories to achieve higher efficiency and more consistent wafer results through different control software and sensors [1][2]. Most of the time-consuming steps, like uploading new recipes to semiconductor machines, can now be done by CIM rather than through the manual steps of an engineer. It makes their work less hectic and less time-consuming. Besides, AI can stop problems with equipment and poor wafer quality arising from errors caused by manual users of semiconductor tools.

As semiconductor technology advances, more intricate methods of chip manufacturing are needed, so it's important to find the best ways to optimize the manufacturing process to achieve maximum results at a lower cost. Big data analytics has emerged as a powerful tool in this context, enabling manufacturers to extract valuable insights from vast amounts of process data and to make data-driven decisions for yield improvement [3][4][5]. The integration of the use of big data analytics has brought about a sea change in the semiconductor sector, allowing businesses to uncover previously unattainable insights from vast amounts of production data and make data-driven decisions. This technological advancement has not only improved manufacturing processes but also accelerated innovation cycles, allowing semiconductor firms to remain competitive in a rapidly evolving market.

Various data sources use automatic communication systems known as telemetry. The use of telemetry data enhances customer experiences while simultaneously monitoring security, application health, quality, and performance [6][7]. What we call "telemetry" really refers to the process of tracking and sending data from faraway places to an IT system that can analyze and monitor it in one specific location. Reliable data gathering and transfer to centralize systems for efficient utilize is made possible via telemetry. One development in telemetry is the rise of big data, which involves gathering large amounts of unstructured data and storing it in a single location [8].

ML and AI have recently advanced to the point where it is more difficult than ever to find a purpose for the massive amounts of data produced during semiconductor manufacture [9]. While engineering heuristics and statistics have traditionally been used for semiconductor process control, it is now encouraged to use scientific data-driven process control instead [10][11][12]. Numerous impacts improve production efficiency by establishing a connection between data obtained from equipment and the wafer process output; a semiconductor

*Corresponding author's ORCID ID: 0000-000-0000-0000
DOI: https://doi.org/10.14741/ijcet/v.15.4.6

manufacturing process includes information on the process that is both direct and indirect [13].

The whole process of an ML model, from data preparation to model training and assessment, is included in an ML pipeline [14][15][16]. It provides a structured framework for chaining together different components, enabling automation and reproducibility in ML experiments and applications. ML pipelines streamline the workflow by ensuring that each step is executed systematically, leading to more efficient development cycles [17].

A. Motivation and Contribution of the Paper

Semiconductor manufacturers gather telemetry data in real time, recording operational data from equipment and sensors about things like temperature, pressure, and what chemicals are present. The reason for this study is that effective anomaly detection and improved process management are very important in the semiconductor industry. Even though modern equipment provides a lot of high-dimensional telemetry, its complexity and the imbalanced data sets make it hard to prevent faults. This results in both major yield losses and extra costs. For this reason, this research strives to devise and test a solid approach that can manage these data issues, increasing efficiency, decreasing waste, and enhancing how products are produced in semiconductor fabrication. You will find the contributions of this study listed below:

- The research used SECOM data from the UCI Repository for the foundation of the analysis.
- Put in place a complete pre-processing process that involved addressing missing data, cleaning the information, scaling it, and lowering the number of variables, all to help analyze the data properly.
- Opted to solve the problem of class imbalance present in these datasets using the SMOTE approach, which ensures a balanced number of minority classes.
- Developed and applied a Decision Tree (DT) Classifier as the core classification algorithm for finding trends and irregularities in the telemetry data that has already been pre-processed.
- Provided a comprehensive evaluation of the effectiveness of the built classification model by assessing its performance using a set of standard measures, including F1-score, accuracy, precision, and recall.

B. Novelty & Justification of the Study

The novelty of this study lies in its integrated methodology that systematically addresses the complex challenges inherent in semiconductor manufacturing telemetry data. While DTCs are well-established, their strategic application within a unified pipeline tailored for high-dimensional, noisy, and imbalanced industrial data represents a significant advancement. The urgent requirement for more precise and timelier anomaly identification and process optimization in semiconductor production serves as the basis for the rationale behind this study. Delayed or missed fault identification can result in considerable financial losses due to reduced yield, increased downtime, and material waste. By enabling the extraction of actionable insights from challenging telemetry datasets, the proposed methodology supports improved operational efficiency, enhanced product quality, and cost-effective manufacturing, thereby contributing to the overall competitiveness of the semiconductor industry.

C. Structure of paper

The article is divided into several important parts. Section II examines the body of research on the use of telemetry data in semiconductor production to provide the foundation. Section III presents the suggested technique. Section IV gives a clear analysis of the data and important findings as part of presenting the research results. Lastly, Section V finishes the paper by talking about the weaknesses of the study and suggesting what could be studied in the future.

Literature Review

This literature review provides a comprehensive analysis of recent advancements in the use of telemetry data for semiconductor manufacturing. Table I summarizes the key aspects of the reviewed studies, including the methodologies adopted, performance metrics achieved, major findings, identified limitations, and proposed future research directions.

Wang et al., (2025) suggest a tuning strategy and yield diagnostic based on ensemble learning and Bayesian optimisation, which show exceptional performance even with a little amount of data. They evaluate the strategy using actual 2-D semiconductor device manufacturing process data. According to experimental findings, the yield prediction method has produced regression fitting results with an explained variance (EVAR) of at least 0.62 and a mean absolute error (MAE) of no more than 8 points, indicating that the model fits the dataset well. To confirm the efficacy of our strategy, they also remanufactured a batch of devices using the yield adjustment suggestions. After analyzing several important variables, including mobility and hysteresis, the test findings showed a final yield score of 86 points, which represented a 62% improvement [18].

Chandu, Mathur and Gupta, (2025) study presents a DL-based algorithm for accurate automated failure type detection of wafer maps. This study set out to investigate DCNNs as a tool for defect identification in semiconductor wafer maps during production. The study makes use of the WM-811K dataset, which includes 811,457 wafer maps with different kinds of defects, to train a model that can correctly detect and

categorize these faults. Preliminary performance parameters indicate that the DCNN model did well, as shown by its 93.75% accuracy rate, 93.81% precision, 93.79% recall, and 93.76% F1-Score that it is superior to traditional ML models, viz., the CNN-WBM model and Logistic Regression (LR) [19].

Patel et al., (2024). This study presents a novel defect detection model that makes use of Explainable AI (XAI) and Federated Learning (FL). FL's decentralized methodology protects data privacy by improving model learning over several nodes without necessitating the pooling of sensitive data. Simultaneously, XAI guarantees that even when trained on dispersed datasets, the generated models retain transparency and reliability. Stakeholders may build ML models on node-specific data using this FL-based defect detection technique without centralizing sensitive data. It supports a variety of ML models, nodes with different capabilities and data volumes, and heterogeneous and asynchronously stored data. By addressing the opacity of deep learning models, FL and XAI demonstrate their predictive behaviour in identifying semiconductor defects. Using a publicly available dataset, empirical findings show a significant increase in the accuracy of defect detection, with an outstanding test accuracy of 98.78% [20].

Dineshkumar et al., (2024) research proposes a unique approach for evaluating semiconductor wafer surface defects using deep convolutional neural networks. First, features were extracted and feature maps were created using a unique structure for feature pyramid networks with atrous convolution (FPNAC). Second, region proposals are generated by feeding the feature plots into the region proposal network (RPN). To correctly categorize and segment the flaws, the region recommendations are finally associated with matching size by way of the inputs of a Radial Basis Function Neural Network (RBFNN), which consists of three branches. The suggested RBFNN produces good overall performance, as evidenced by the experimental findings, which show Mean Intersection over Union (MIoU) of 90.06% and Mean Pixel Accuracy (MPA) of 94.97% [21].

Pradeep et al. (2023) Utilizing ML methods on computational data gathered from the production unit's sensors, they have been able to forecast wafer failure in semiconductor manufacturing, reduce equipment failure by allowing predictive maintenance, and boost productivity. By using the suggested feature selection procedure, training time has been significantly decreased while retaining good accuracy. The model-building methods used in this study include Neural Networks, LR, RF-Classifier, SVM, DTC, and XG-Boost. To investigate accuracy and precision, a large number of case studies were conducted. The Random Forest Classifier's accuracy of over 93.62% outperformed all other models [22].

Yuen et al. (2023) propose A new way of classifying defects with just a small dataset, called GENSS, that uses geometric properties to create synthetic samples and then selects appropriate models based on Structural Similarity and image hashing. At the beginning, researchers found that the suggested method provides high accuracy, at a level of 85.71 % [23].

Despite notable progress in semiconductor fault detection and yield prediction utilizing DL and ML methods, current methods have significant drawbacks. Such as poor scalability, reliance on complex architectures, lack of interpretability, and insufficient handling of data imbalance. Many models require large, labeled datasets or perform inadequately with imbalanced or heterogeneous data common in real-world manufacturing environments. Additionally, integration into scalable and real-time production pipelines remains a challenge. To overcome these gaps, this work proposes a scalable and interpretable ML pipeline that incorporates Min-Max normalization for feature scaling, SMOTE for addressing class imbalance, and a DTC for efficient and explainable fault classification. This unified approach enhances model performance, supports practical deployment in semiconductor manufacturing, and ensures robust defect detection across diverse telemetry data.

Table 1 Summary of Literature Overview and Review on Semiconductor Manufacturing

Author	Methodology	Datasets	Key Findings	Limitations	Future Approach
Wang et al., (2025)	Ensemble Learning + Bayesian Optimization	Real 2-D semiconductor fabrication process data	MAE ≤ 8, EVAR ≥ 0.62; yield improved by 62% after tuning	Performance evaluation is limited to one dataset; it lacks comparison with other methods	Extend to diverse fabrication datasets; integrate more advanced optimization techniques
Chandu, Mathur & Gupta (2025)	Deep Convolutional Neural Network (DCNN)	WM-811K (811,457 wafer maps)	Accuracy: 93.75%, Precision: 93.81%, Recall: 93.79%, F1: 93.76%; outperforms CNN-WBM and LR	May not generalize well to unseen wafer types; lacks interpretability	Incorporate explainability into DCNN; evaluate on real-time streaming wafer data
Patel et al. (2024)	Federated Learning (FL) + Explainable AI (XAI)	Public semiconductor fault dataset	Accuracy: 98.78%; enables privacy-preserving and interpretable fault detection.	Not specified which public dataset; asynchronous model updating could be challenging.	Apply to edge computing environments; improve handling of heterogeneous data distributions.

Dineshkumar et al., (2024)	FPN with Atrous Convolution + RBF Neural Network	Semiconductor wafer surface images	MIoU: 90.06%, MPA: 94.97%; efficient segmentation of surface defects	Dataset not disclosed; RBFNN scalability unclear	Test on multi-fault/multi-class wafer data; explore lightweight architectures for deployment
Pradeep et al. (2023)	ML Models (RF, SVM, DT, LR, XGBoost, NN) + Feature Selection	Sensor data from manufacturing units	RF achieved 93.62% accuracy; predictive maintenance is possible with sensor analytics.	Lacks detail on sensor types; real-time integration not discussed	Expand to time-series sensor analysis; explore online learning methods
Yuen et al. (2023)	GENSS (Synthetic Data Gen + SSIM + Image Hashing)	Extremely small wafer defect datasets	Accuracy: 85.71%; effective on small data using synthetic augmentation	Synthetic data quality may vary; low data scenarios only	Enhance synthetic data realism; test generalizability across domains
Collart et al. (2022)	Rule-based + Statistical ML Models	High Volume Manufacturing (HVM) cryogenic pump data	Accuracy up to 93%, Recall up to 87%; guides maintenance of cryogenic pumps	Domain-specific to cryogenic pumps; rule-based logic may not generalize	Extend model training to other equipment types; integrate IoT-based predictive systems

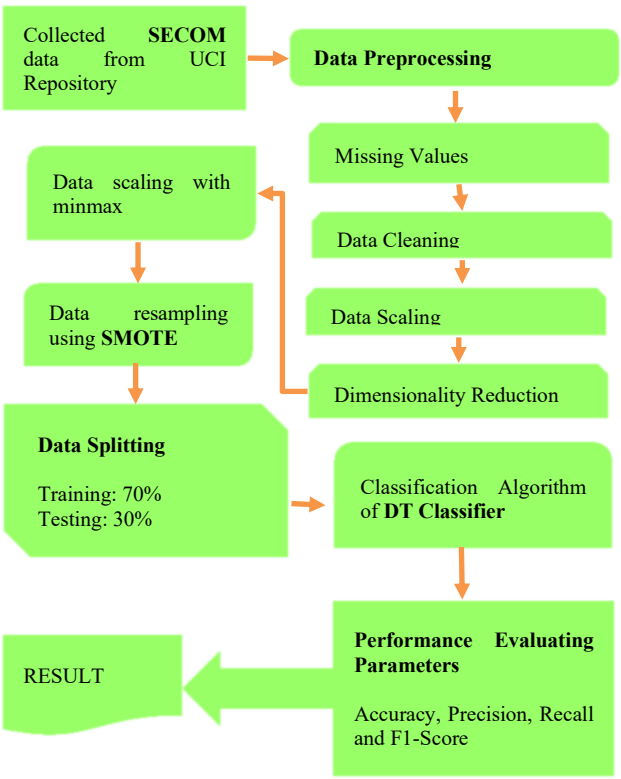

Fig.1 Flowchart diagram of Semiconductor Manufacturing

Methodology

The methodology of ML for Big Telemetry Data in Semiconductor Manufacturing is illustrated in Figure 1. A flowchart diagram of Semiconductor Manufacturing comprises multiple systematic stages. Initially, telemetry data is collected from the SECOM dataset available on the UCI Repository. A robust preprocessing pipeline follows, addressing missing values, executing data cleaning, and applying feature scaling using Min-Max normalization to standardize the input features. Dimensionality reduction techniques are then employed to mitigate high dimensionality and enhance computational efficiency. Class imbalance is addressed using the Synthetic Minority Over-sampling (SMOTE) technology. Following that, the enhanced dataset is used to produce training (70%) and testing (30%) sets. A DT classifier's performance is evaluated using standard metrics like as Accuracy, Precision, Recall, and F1-Score, guaranteeing a trustworthy evaluation of the model's effectiveness in a high-volume production setting.

Here is a comprehensive, step-by-step breakdown of the processes illustrated in the flowchart:

Data Gathering

This method and ML models using the SECOM dataset to forecast whether items will be excellent or bad based on information about semiconductor manufacturing processes. The UCI Repository is the source of the dataset. The SECOM dataset has been extensively utilized by researchers aiming to address real-world classification challenges such as fault diagnosis and detection. SECOM includes actual data produced by the manufacturing of semiconductor procedures and is used in studies to categorise items as either excellent or poor. There are 41,951 missing values in the dataset, which makes up around 4.54% of the total. In all, it has 590 characteristics, excluding time and labels. There are only 1568 examples in the dataset, which includes 104 examples of the faulty product class and 1464 examples of the excellent product class.

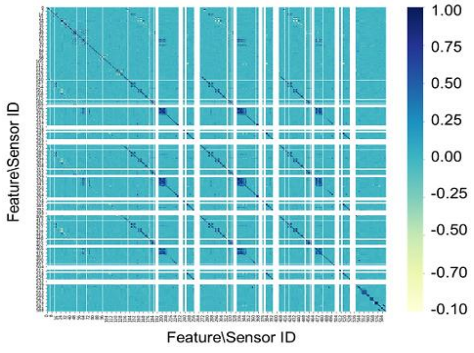

Fig. 2 Features Correlation Analysis of SECOM Dataset

Fig. 2 presents the heat map "Features Correlation Analysis of SECOM Dataset", which illustrates the correlation matrix among various sensor-based features. The color bar represents correlation coefficients ranging from approximately –0.10 (light yellow) to 1.00 (dark blue). Dark blue areas along the diagonal and certain off-diagonal blocks indicate strong positive correlations, implying that paired features tend to increase together. Light blue regions represent weak or negligible correlations, while light yellow, though sparse, denotes weak negative relationships. The visible grid-like partitioning suggests a structured grouping of related sensor features within the dataset.

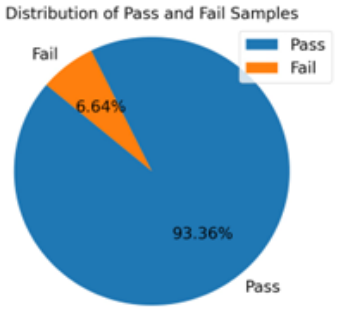

Fig. 3 Pie chart of the Distribution of Pass/Fail samples

Fig. 3 depicts the pie chart titled "Distribution of Pass and Fail Samples", highlighting a significant class imbalance within the dataset. The "Pass" category, shown in blue, comprises 93.36% of the total samples, whereas the "Fail" category, represented in orange, accounts for only 6.64%. This distribution indicates a dominant presence of successful outcomes, emphasizing the need to address class imbalance during model training to ensure robust classification performance.

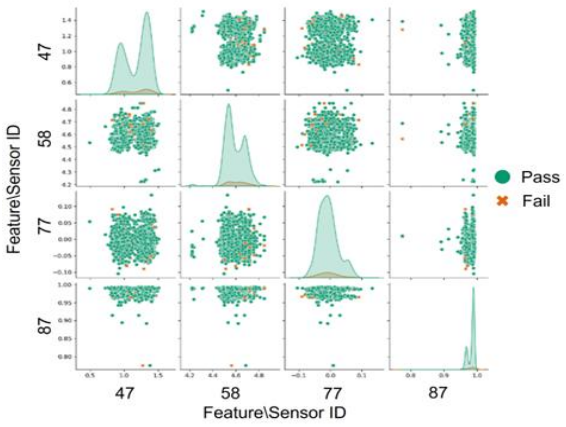

Fig. 4 Exploratory Data Analysis for SECOM Data

Fig. 4 shows the pair plot "Exploratory Data Analysis for SECOM Data", illustrating distributions and relationships among four sensor features (IDs 47, 58, 77, and 87), labeled by outcome: "Pass" (green) and "Fail" (orange). Diagonal plots display KDE-based distributions, while off-diagonal plots reveal inter-feature relationships. Significant overlap between classes suggests limited discriminative power from these features alone.

Data Pre-processing

To ensure optimal performance in analyzing telemetry data in semiconductor manufacturing, the datasets must undergo preprocessing to eliminate inconsistencies, irrelevant information, and missing values. This process typically involves tasks such as data scaling, data cleansing, dimensionality reduction, and missing value imputation. The following discusses a few preprocessing methods:

- **Data Cleaning:** Prior to preprocessing and analysis, columns containing single values and missing values were eliminated. For each variable, the percentage of missing data values was determined. The corresponding 32 variables were removed from the analysis if the percentage of missing values prior to replacement was more than half, since this was deemed to be an unnecessary replacement.
- **Missing Values:** In the actual world, missing values happen for a variety of causes. For example, survey respondents could fail to respond to some questions[24].
- **Data Scaling:** The SECOM dataset's data size was normalized and modified to avoid the issue of some feature values being either too big or too small, which might cause the data to diverge to infinity or converge to zero during the training of the classification model.
- **Dimensionality Reduction:** There are 590 features in the high-dimensional dataset SECOM. The curse of dimensionality occurs when ML uses more characteristics in high-dimensional data.

Data scaling with minmax

The features are rescaled to a standard range of [0,1] using Min-Max normalization. This ensures that no one feature dominates the learning process because of its scale. The transformation is defined by the following equation:

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

Where the initial feature value is represented by X, X_{min} and X_{max} represent the feature's lowest and highest values, respectively. This normalization technique retains the relationships among the original data values while constraining them within a bounded interval, thereby facilitating efficient model training.

Data Resampling with SMOTE

Data resampling primarily seeks to address the data problem of inequality between majority and minority groups. To avoid the testing data from being overfit, this step is only performed for the training dataset. Two distinct approaches are used: The minority class is oversampled using SMOTE, whereas the majority class is under sampled in conjunction with SMOTE. To generate new data points, the minority group used SMOTE, which leverages current data points to interpolate. Using the formula, the new synthetic data point is produced (2):

$$x_{new=x_i+\lambda(x_j-x_i)} \quad (2)$$

S class instances, and λ is a random number between 0 and 1.

Data Splitting

At the data splitting stage, a vital part of machine learning is dividing the dataset into testing and training sets, with the former typically receiving 70% of the data and the latter 30% of the data.

Classification with Decision Tree Classifier

There is widespread use of DT in using remote sensing technologies to categorize land cover. Applying ML to data analysis via the development of several tree-structured algorithm classes is what the DT is all about. The data properties are tested at each node, the results are stored in each branch, and the algorithm's class predictions are shown by the leaves[25][26]. Classification rules are used as a practical method for implementing a DT. The DT works without depending on how variables are distributed, or it can be said that it is independent of these assumptions.

The purpose of emergency mode recognition involves testing simulation and ML, and we will decide to use the "DT" algorithm[27][28]. Here, the computational steps are some simple decision rules grouped into a tree as shown in equations (3 and 4).

The rule for each decision is based on whether some piece of information surpasses a certain threshold. Each of the leaves represents a predictable class there[29]. Being able to find out in which a trained DT examines the characteristics of an electrical network means you have to travel through the tree from the root and pass through the nodes until you reach a side node.

$$H(X) = -\sum_{i=1}^{n} p(x_i)\log_2 p(x_i) \quad (3)$$
$$H(X|Y) = -\sum_{j=1}^{m} p(x_j|y_i)\log_2 p(x_j|y_i) \quad (4)$$

where $H(X)$ s the original entropy, ; $H(X|Y)$ becomes the new total entropy when the node is split according to condition$Y$; $p(x_i)$ means the ratio of objects in the training sample with class i; $p(y_0)$ adds up to the fraction of objects in the sample that do not satisfy the node's condition; $p(y_1)$ is the sum of the objects in the sample that are classified by the node;; $p(x_i|y_0)$ tequals the number of objects in the sample with class i that do not meet the condition in the node; $p(x_i|y_1)$the proportion of objects that have class i among the objects that satisfy the condition in the node.

Evaluation Parameters

Accuracy, precision, recall, F1-score, and the loss function were some of the performance metrics applied to the DTC on the telemetry data in semiconductor manufacturing. The effectiveness of the model in linking variables can best be seen through these metrics. The evaluation relied on the SECOM dataset to guarantee that the results are sound and accurate. We will now review how the parameters in the confusion matrix are calculated below:

- **True Positives (TP):** The model correctly identifies a faulty product or manufacturing instance based on telemetry signals indicating an actual fault was present and correctly detected.
- **True Negatives (TN):** The model correctly classifies a normal (non-faulty) product or instance, indicating no fault was present and none was predicted.
- **False Positives (FP):** It seems that the model picked up on a typical product all wrong or processed as faulty, suggesting a fault where none exists, potentially leading to unnecessary inspections or rework.
- **False Negatives (FN):** The model fails to detect an actual fault, misclassifying a faulty instance as normal, which can result in defective products progressing through the manufacturing process undetected.

Evaluation metrics for specific classes were calculated using standard classification formulas:

Accuracy

The percentage of both defective and non-faulty cases that were accurately predicted relative to the total number of instances. It reflects the overall correctness of the model, but may be misleading if the data is imbalanced. It is more formally defined as in Eq. (5):

$$\text{Accuracy} = \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \quad (5)$$

Precision

A ratio of the number of actual favorable results to the total number of positive outcomes projected. It shows the proportion of genuine problems in the identified group, which helps limit incorrect decisions and extra checks, and precision is determined using the following formula (6):

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (6)$$

Recall

The ratio of all favorable outcomes to 100% accurate predictions. It assesses how well the model can identify flawed examples, which is critical for preventing defective products from advancing in the manufacturing process. Mathematically, we define it as in Eq. (7):

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (7)$$

F1-Score

Recall and precision harmonic mean. It is generated as shown below and offers FP and FN are both considered in a balanced measure. It is particularly helpful in situations when the class distribution of fault vs. non-fault data is unequal. In eq. (8):

$$\text{F} - \text{Score} = 2 \times \frac{\text{TP+TN}}{\text{TP+TN+FP+FN}} \quad (8)$$

Our dataset has. been collected from two large enterprise.

1) systems, named ESX-1 and ESX-2. The security raw events
2) were collected over 5 months for ESX-1, over 30 days for
3) ESX-2, respectively, in which the detection of threat information
4) was separately recorded by the SOC security analysts
5) V. whenever a network intrusion occurs. The list of threats
6) Detection information contains the threat occurrence time, related
7) attacks, category of attack, respond contents, attack IP address,
8) and victim network information.
9) In our datasets, we investigated 798 detecting cyber threats
10) X.in ESX-1, which are dispersed across the entire collection
11) period. Looking at the type of occurred attacks in recorded
12) cyber threats, there are 240 scanning, 547 system hacking, and
13) 11 worm attacks. Similarly, in ESX-2 there are 941 scannings,
14) 3,077 system hacking, and 51 worm attacks. This categorising
15) of attack type was manually performed by SOC analysts. By
16) category, the system hacking attack includes a cross site script,
17) DDoS, brute force attack, and injection attack. A trojan and
18) backdoor attack belongs to scanning attack. Overall the
19) number of attacks was found 4,079 cyber-threats
20) Our dataset has been collected from two large enterprise
21) systems, named ESX-1 and ESX-2. The security raw events
22) were collected over 5 months for ESX-1, over 30 days for
23) ESX-2, respectively, in which the detecting threat information
24) was separately recorded by the SOC security analysts
25) whenever a network intrusion occurred. The list of threat
26) detection information contains threat occurrence time, related
27) attacks, category of attack, respond contents, attack IPaddress,
28) and victim network information.
29) In our datasets, we investigated 798 detecting cyber threats
30) in ESX-1, which are dispersed across the entire collection
31) period. Looking at the type of occurred attacks in recorded
32) cyber threats, there are 240 scanning, 547 system hacking, and
33) 11 worm attacks. Similarly, in ESX-2 there are 941 scanning,
34) 3,077 system hacking and 51 worm attacks. This categorising
35) of attack type was manually performed by SOC analysts. By
36) category, the system hacking attack includes a cross site script,
37) DDoS, brute force attack, and injection attack. A trojan and
38) backdoor attack belongs to scanning attack. Overall the
39) number of attacks were found 4,079 cyber-threats
40) Our dataset has been collected from two large enterprise
41) systems, named ESX-1 and ESX-2. The security raw events
42) were collected over 5 months for ESX-1, over 30 days for
43) ESX-2, respectively, in which the detecting threat information
44) was separately recorded by the SOC security analysts
45) whenever a network intrusion occurred. The list of threat
46) detection information contains threat occurrence time, related
47) attacks, category of attack, respond contents, attack IPaddress,
48) and victim network information.
49) In our datasets, we investigated 798 detecting cyber threats

50) in ESX-1, which are dispersed across the entire collection
51) period. Looking at the type of occurred attacks in recorded
52) cyber threats, there are 240 scanning, 547 system hacking, and
53) 11 worm attacks. Similarly, in ESX-2 there are 941 scanning,
54) 3,077 system hacking, and 51 worm attacks. This categorising
55) of attack type was manually performed by SOC analysts. By
56) category, the system hacking attack includes a cross site script,
57) DDoS, brute force attack, and injection attack. A trojan and
58) backdoor attack belongs to scanning attack. Overall the
59) number of attacks were found 4,079 cyber-threats
60) Our dataset has been collected from two large enterprise
61) systems, named ESX-1 and ESX-2. The security raw events
62) were collected over 5 months for ESX-1, over 30 days for
63) ESX-2, respectively, in which the detecting threat information
64) was separately recorded by the SOC security analysts
65) whenever a network intrusion occurred. The list of threat
66) detection information contains threat occurrence time, related
67) attacks, category of attack, respond contents, attack IPaddress,
68) and victim network information.
69) In our datasets, we investigated 798 detecting cyber threats
70) in ESX-1, which are dispersed across the entire collection
71) period. Looking at the type of occurred attacks in recorded
72) cyber threats, there are 240 scanning, 547 system hacking, and
73) 11 worm attacks. Similarly, in ESX-2 there are 941 scanning,
74) 3,077 system hacking, and 51 worm attacks. This categorising
75) of attack type was manually performed by SOC analysts. By
76) category, the system hacking attack includes a cross site script,
77) DDoS, brute force attack, and injection attack. A trojan and
78) backdoor attack belongs to scanning attack. Overall the
79) number of attacks were found 4,079 cyber-threats.

Results Analysis and Discussions

The experiments were carried out on an HP laptop featuring an Intel® Core™ i7-11800H processor clocked at 2.30 GHz, supported by A 1 TB NVMe SSD and 16 GB of DDR4 RAM. Operating on 64-bit Windows 10 Pro, the system provided a modern and robust environment for implementing the suggested approach. The suggested DTC demonstrated its strong performance in evaluating telemetry data in semiconductor production by achieving a remarkable classification accuracy of 88% using the well-known SECOM dataset. The findings presented in Table II confirm that DTC offers many advantages by making use of advanced ML techniques in industrial telemetry.

Table 2 Results of DTC Model for Telemetry Data in Semiconductor Manufacturing

Metrics	DT
Accuracy	88
Precision	48
Recall	86
F_1-Score	60

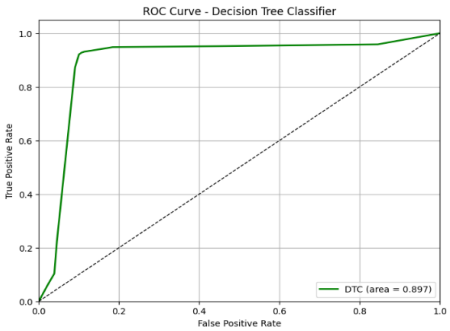

Fig. 5 ROC Curve of DT Classifier

Figure 5 is a Receiver Operating Characteristic (ROC) curve for the DTC using the dataset. On the ROC curve, the TPR is plotted against the FPR at different thresholds for classifying cases, letting us easily observe how the model compares different situations. It should be mentioned that this DTC model delivers admirable performance, reaching an AUC of 0.897. Since the model received a high AUC score, it can identify if an issue is positive or negative, meaning it finds faults accurately with few false alarm cases.

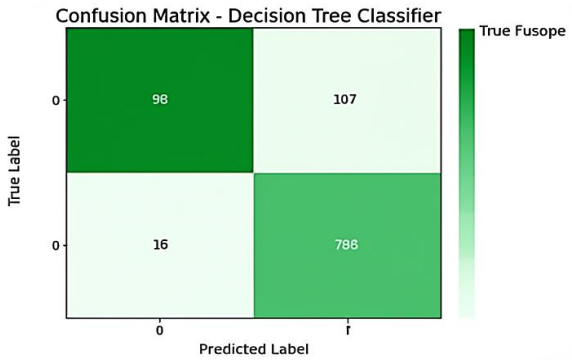

Fig. 6 Confusion Matrix of DT Classifier

The DTC performance is presented in figure 6 as the number of true and false predictions made by the

model. The data shows that out of all the instances, 98 were identified as True Negatives, and 786 as TP. On the other hand, it marked 107 zero values as one FP and 16 one values as zero FN.

Comparative Analysis

The table highlights the similarities between our CNN model and top ML methods like KNN, NB, and XGBoost, when these algorithms are used on telemetry data in the semiconductor production field. DTC showed the best results with an 88% accuracy, higher than the results of both traditional ML and DL methods. The best performance was shown by LR with an accuracy of 70.22%, while the Recurrent Neural Network (RNN) was at 68.1% and then the Multi-Layer Perceptron (MLP) at 52.2%.

Table 3 Comparison of ML and DL Model's for Telemetry Data in Semiconductor Manufacturing

Models	Accuracy
LR [30]	70.22
RNN [31]	68.1
MLP [32]	52.2
DTC	88

The DTC achieved an accuracy of 88% with telemetry data from semiconductor manufacturing, which is much better than the results produced by other traditional ML and DL models. This demonstrates that the DTC model can handle the complicated interactions and non-linear patterns noticed in data from many different industries. Since it is understandable, computationally efficient, and easy to apply, it works very well for both real-time finding faults and process optimization in semiconductor fabrication factories.

Conclusion And Future Work

The fast-changing nature of semiconductor manufacturing has caused significant amounts of telemetry data to be collected that highlight the finer details of each process. It is vital to examine this diverse and complex data to sustain high product quality and the smooth running of operations. These models can spot hidden issues and irregularities, so faults are discovered early and maintenance can be scheduled in advance. It makes use of these capabilities to design a dependable way to identify faulty products by processing actual semiconductor telemetry data. This work developed a detailed ML framework that works well with big telemetry data from semiconductor manufacturing using the SECOM dataset. It was decided to use a DTC, and the results showed an excellent accuracy of 88%, which is much higher than LR, RNN, and MLP. Additionally, the model worked well in measuring Precision, Recall, F1-Score, and AUC, confirming that it can properly detect defective products in manufacturing settings. Moreover, because DL is clear to understand and can be processed quickly, it is ideal for use in industry.

Additionally, the next steps will be to line up RF and GB methods in the model, which can boost the model's robustness and may enhance its classification performance. Moreover, I will research how using feature selection and deep learning models like CNNs and Transformers improves the process of incorporating both space and time in the models. New demands of Industry 4.0 and smart factories will be addressed with the help of real-time industrial operation and the use of instant telemetry data for identifying faults.

References

[1] T. Song, Y. Qiao, Y. He, J. Li, N. Wu, and B. Liu, "A New Framework of the EAP System in Semiconductor Manufacturing Internet of Things," Electronics, vol. 12, no. 18, pp. 1–16, Sep. 2023, doi: 10.3390/electronics12183910.

[2] V. Rajavel, "Optimizing Semiconductor Testing: Leveraging Stuck-At Fault Models for Efficient Fault Coverage," Int. J. Latest Eng. Manag. Res., vol. 10, no. 2, pp. 69–76, Mar. 2025, doi: 10.56581/IJLEMR.10.02.69-76.

[3] T. P. -, "Process Optimization in Semiconductor Manufacturing: The Role of Big Data Analytics in Yield Improvement," Int. J. Multidiscip. Res., vol. 1, no. 2, pp. 1–11, Sep. 2019, doi: 10.36948/ijfmr.2019.v01i02.23444.

[4] S. Garg, "AI-Driven Innovations in Storage Quality Assurance and Manufacturing Optimization," Int. J. Multidiscip. Res. Growth Eval., vol. 6, no. 2, pp. 1083–1087, 2025, doi: 10.54660/.IJMRGE.2025.6.2.1083-1087.

[5] S. R. Thota, S. Arora, and S. Gupta, "Quantum-Inspired Data Processing for Big Data Analytics," in 2024 4th International Conference on Advancement in Electronics & Communication Engineering (AECE), 2024, pp. 502–508. doi: 10.1109/AECE62803.2024.10911758.

[6] A. Gogineni, "Artificial Intelligence-Driven Fault Tolerance Mechanisms for Distributed Systems Using Deep Learning Model," J. Artif. Intell. Mach. Learn. Data Sci., vol. 1, no. 4, pp. 2401–2406, Dec. 2023, doi: 10.51219/JAIMLD/anila-gogineni/519.

[7] S. Murri, "Data Security Environments Challenges and Solutions in Big Data," Int. J. Curr. Eng. Technol., vol. 12, no. 6, pp. 565–574, 2022.

[8] R. Jain, M. Rohit, A. Kumar, A. Bakliwal, A. Makwana, and M. Rahevar, "Prediction of Telemetry Data using Machine Learning Techniques," IJERT, vol. 11, no. 9, pp. 58–64, 2022, doi: 10.17577/IJERTV11IS090048.

[9] V. Panchal, "Mobile SoC Power Optimization : Redefining Performance with Machine Learning Techniques," IJIRSET, vol. 13, no. 12, pp. 1–17, 2024, doi: 10.15680/IJIRSET.2024.1312117.

[10] J. Moyne, J. Samantaray, and M. Armacost, "Big Data Capabilities Applied to Semiconductor Manufacturing Advanced Process Control," IEEE Trans. Semicond. Manuf., vol. 29, no. 4, pp. 283–291, Nov. 2016, doi: 10.1109/TSM.2016.2574130.

[11] A. kumar Polinati, "Devops And Ai: Automating Software Delivery Pipelines For Continuous Integration And Deployment," Nanosci. Nanotechnol. Platf., vol. 20, no. 4, pp. 1–18, 2024.

[12] V. Rajavel, "Integrating Power-Saving Techniques into Design for Testability of Semiconductors for Power-Efficient Testing," Am. J. Eng. Technol., vol. 07, no. 03, pp. 243–251, Mar. 2025, doi: 10.37547/tajet/Volume07Issue03-22.

[13] D. H. Seol, J. E. Choi, C. Y. Kim, and S. J. Hong, "Alleviating Class-Imbalance Data of Semiconductor

Equipment Anomaly Detection Study," Electronics, vol. 12, no. 3, pp. 1–13, Jan. 2023, doi: 10.3390/electronics12030585.
[14] R. Q. Majumder, "Machine Learning for Predictive Analytics: Trends and Future Directions," Int. J. Innov. Sci. Res. Technol., vol. 10, no. 04, pp. 3557–3564, 2025.
[15] S. P. Kalava, "AI-Powered Development: How Artificial Intelligence is Shaping Software Productivity," Sci. Res. Community, p. 4, 2024.
[16] A. Polleri, R. Kumar, M. M. Bron, G. Chen, and R. S. B. Shekhar Agrawal, "Identifying a classification hierarchy using a trained machine learning pipeline," 17303918, 2022
[17] M. Bdair, "Enhancing Machine Learning Workflows: A Comprehensive Study of Machine Learning Pipelines," 2022.
[18] C. Wang et al., "Yield Diagnosis and Tuning for Emerging Semiconductors During Research Stage," IEEE Access, vol. 13, pp. 78915–78927, 2025, doi: 10.1109/ACCESS.2025.3563761.
[19] H. S. Chandu, S. Mathur, and S. Gupta, "Artificial Intelligence-Driven Approaches for Automatic Wafer Map Failure Detection in Semiconductor Manufacturing," in 2025 IEEE International Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI), 2025, pp. 1–6. doi: 10.1109/IATMSI64286.2025.10985054.
[20] T. Patel, R. Murugan, G. Yenduri, R. H. Jhaveri, H. Snoussi, and T. Gaber, "Demystifying Defects: Federated Learning and Explainable AI for Semiconductor Fault Detection," IEEE Access, vol. 12, pp. 116987–117007, 2024, doi: 10.1109/ACCESS.2024.3425226.
[21] R. Dineshkumar, K. S. Kumar, C. Murugan, R. R. Al-Fatlawy, and P. Harshitha, "An Efficient Wafer Semiconductor Surface Defect Inspection Using Radial Basis Functional Neural Network," in 2024 Second International Conference on Data Science and Information System (ICDSIS), 2024, pp. 1–4. doi: 10.1109/ICDSIS61070.2024.10594056.
[22] D. Pradeep, B. V. Vardhan, S. Raiak, I. Muniraj, K. Elumalai, and S. Chinnadurai, "Optimal Predictive Maintenance Technique for Manufacturing Semiconductors using Machine Learning," in 2023 3rd International Conference on Intelligent Communication and Computational Techniques (ICCT), IEEE, Jan. 2023, pp. 1–5. doi: 10.1109/ICCT56969.2023.10075658.
[23] S. L. Yuen et al., "GENSS: Defect Classification Method on Extremely Small Datasets for Semiconductor Manufacturing," in 2023 27th International Computer Science and Engineering Conference (ICSEC), IEEE, Sep. 2023, pp. 419–424. doi: 10.1109/ICSEC59635.2023.10329633.
[24] S. Arora, S. R. Thota, and S. Gupta, "Data Mining and Processing in the Age of Big Data and Artificial Intelligence - Issues, Privacy, and Ethical Considerations," in 2024 4th Asian Conference on Innovation in Technology (ASIANCON), 2024, pp. 1–6. doi: 10.1109/ ASIANCON62057.2024. 10838087.
[25] K. Battista, K. A. Patte, L. Diao, J. A. Dubin, and S. T. Leatherdale, "Using Decision Trees to Examine Environmental and Behavioural Factors Associated with Youth Anxiety, Depression, and Flourishing," Int. J. Environ. Res. Public Health, vol. 19, no. 17, pp. 1–16, Aug. 2022, doi: 10.3390/ijerph191710873.
[26] B. B. Anjani kumar Polinati, Sangeeta Singh, Satyasri Akula, Raveendra Reddy Pasala, Monu Sharma, Sukanth Kumar Korkanti, "Revolutionizing Information Management: AI-Driven Decision Support Systems for Dynamic Business Environments," J. Inf. Syst. Eng. Manag., vol. 10, no. 35s, pp. 1–14, 2025.
[27] X. Zhang, P. M. Treitz, D. Chen, C. Quan, L. Shi, and X. Li, "Mapping mangrove forests using multi-tidal remotely-sensed data and a decision-tree-based procedure," Int. J. Appl. Earth Obs. Geoinf., vol. 62, pp. 201–214, Oct. 2017, doi: 10.1016/j.jag.2017.06.010.
[28] V. Panchal, "Energy-Efficient Core Design for Mobile Processors: Balancing Power and Performance," Int. Res. J. Eng. Technol., vol. 11, no. 12, pp. 1–11, 2024.
[29] A. Kulikov, A. Loskutov, D. Bezdushniy, and I. Petrov, "Decision Tree Models and Machine Learning Algorithms in the Fault Recognition on Power Lines with Branches," Energies, vol. 16, no. 14, pp. 1–19, Jul. 2023, doi: 10.3390/en16145563.
[30] B. Jalil, K. Lumpur, S. L. M. Belaidan, B. Jalil, and K. Lumpur, "Fault Detection And Prediction In The Semiconductor Manufacturing," Int. Conf. Distrib. Comput. Optim. Tech., vol. 11, no. 11, pp. 2023–2028, 2023, doi: 10.34218/IJM.11.11.2020.192.
[31] V. K. Pendyala and E. J. Yellig, "A Benchmark Time Series Dataset for Semiconductor Fabrication Manufacturing Constructed using Component-based Discrete-Event Simulation Models arXiv : 2408 . 09307v1 [cs . LG] 17 Aug 2024," no. NeurIPS 2024.
[32] Y. El Mourabit, Y. El, H. Zougagh, and Y. Wadiai, "Predictive System of Semiconductor Failures based on Machine Learning Approach," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 12, pp. 199–203, 2020, doi: 10.14569/IJACSA.2020.0111225.