BURN Termination Disposition System Development using Supervised/Unsupervised Natural Language Processing

MunSik Park, *Test Solution Engineer,* Choon Hian Chai, *Test Solution Engineer Manager,* Fei Keng Tan, *Senior Test Solution Engineer Lead,* Kin Keong Leong, *Principal Test Solution Engineer*

*Abstract:* When termination occurs at the Burn oven sessions, Engineering Data Technicians (EDT) primarily check the error messages and resolve the termination. If the technician cannot handle the problem, the case is escalated to Test Solutions Engineers (TSE). After escalation, TSE investigates issues and gives resolution to EDT through email. However, replies might be delayed during night or non-working hours. As a result, it can cause work inefficiency. Therefore, this paper aims to develop web applications using Machine Learning (ML) to improve work efficiency for related teams, especially in the absence of TSE. The overall methodology is collecting past data logs from the Burn oven session, applying the Natural Language Processing (NLP) technique and ML algorithms to train our model, and suggesting a disposition method through a web application. First, Perl script collects oven error message data, about 3000 data sets, and Term Frequency-Inverse Document Frequency (TF-IDF) is applied to error message data sets. Second, clustering algorithms are tested, and Density-Based Spatial Clustering of Applications with Noise (DBSCAN) is applied to error messages to cluster data. After that, the data is labeled through reviews by experienced engineers. Third, classification algorithms are tested, and Support Vector Classifier (SVC), which has the highest F1 score, is chosen for the web application. Web applications can handle up to 140 termination cases per week. Also, it can be used to crunch through oven error logs and look for the commonality of the termination error message. Ultimately, it will reduce the overall cycle time.

*Index Terms:* Decision support systems, Machine learning, Semiconductor devices, Support vector machines

# Introduction

The semiconductor industry is one of the most complicated industries in which productivity enhancement, yield enhancement, continual cost reduction, fast ramp-up, on-time delivery, and cycle time reduction are the important ways for operational excellence to maintain competitive advantages [1]. Cycle time and yield are both important measures, yet cycle time is frequently prioritized as the ability to manufacture and supply goods more quickly than rivals can give a notable advantage, resulting in greater market share and improved profits. Cycle time is often expressed as the x-factor, the ratio of actual cycle time to processing time, and should be as close to 1 as possible. In fact, the biggest contributor to cycle times is wait times. Extended delays in production are likely due to the need for engineers' decisions when issues arise on the production floor. Hence, developing an automated system that offers recommended actions for engineers or technicians is crucial to enhancing their work productivity.

## Research Background

Currently, in the Burn stage, EDT and TSE have different roles but work closely together to ensure smooth Backend production line testing. TSE members prepare and maintain test programs to test products while the EDT team monitors and handles the manufacturing issues. Since ovens need to run 24 hours/7 days, real-life manufacturing issues are bound to arise. The EDT team performs first-level troubleshooting of manufacturing issues on the production line. EDT normally escalate issues to TSE whenever they encounter issues they cannot resolve. After escalating the issues, TSE investigates and provides a resolution or debug process to the EDT team to handle the issue. However, when errors occur during non-working hours (weekends, midnight, and holidays), TSE may not be available immediately to provide dispositions, causing delays. Delayed dispositions lead to termination lots occupying the oven for extended periods. This delay affects production cycle time due to increased test cycles, especially BURN steps with longer test times. This paper aims to improve the work efficiency of the BURN TSE and EDT engineers, especially in the absence of TSE. A new methodology for solving the problems mentioned above is proposed by developing a web application that applies ML algorithms to handle dispositions.

## Literature Review and Novelty of this work

### Literature Review

Many research papers have been published to solve the inefficiency of the manufacturing process. Industries are going through the fourth industrial revolution (Industry 4.0), where technologies like the Industrial Internet of Things, big data analytics, and ML are extensively utilized to improve the productivity and efficiency of manufacturing systems and processes [2]. Supervised/Unsupervised ML algorithms are applied in various Semiconductor processes for Fault detection and classification [3], identifying real-time factors influences in semiconductor fab [4], outlier detection [5], and yield prediction [6].

Deep Learning (DL) is also widely used to improve efficiency or solve problems in the semiconductor manufacturing process. Convolution Neural Network (CNN) models are applied for fault detection and diagnosis [7] and wafer surface defects classification [8].

In the probe process, the ML/DL algorithm is also used to classify wafer map patterns to improve wafer quality [9] and detect test-induced defects [10].

### Novelty of this project

Based on the literature review, ML/DL algorithms are applied in various semiconductor manufacturing processes, such as designing, etching, probing, etc., to analyze faults or improve the yield of wafers. Normally, when defects occur in the manufacturing line, the analysis is triggered to find the root cause of a failure and its findings, recorded in a reporting system using natural language [11]. Therefore, analyzing natural language is also helpful in finding the root cause and improving work efficiency. The novelty of this research can be stated as follows:

1. **Applying NLP technique to manufacturing equipment event logs:** In the semiconductor field, ML methodology is mainly used to analyze quantitative data. However, the method proposed in this paper is used to analyze event logs of semiconductor equipment, not quantitative data. Therefore, the NLP algorithm is directly applied to manufacturing equipment event logs to change the Natural Language format to Quantitative Vector format.
2. **Combination use of Unsupervised/Supervised ML Algorithms:** The proposed method in this paper applied both unsupervised and supervised ML algorithms. Since it is difficult for engineers to label and check massive raw error message logs individually, an unsupervised learning clustering algorithm was applied to cluster roughly similar event logs. After clustering, the subject matter expert (SME) reviewed the clustering results to see whether similar errors belong to the same clustering group. They also labeled clustering results as types of errors. After labeling the data, a supervised classification algorithm was used to classify error messages.
3. **Creating web applications for end-users:** The goal in developing web applications is to enable users to efficiently manage real-time termination cases via user-friendly website access. The website prompts users for basic information and then delivers recommended solutions corresponding to identified error groups as output. Furthermore, the web application is continually refined and improved through the insightful feedback provided by its users, aiming to increase its practicality and ease of use. The application offers more than recommendations; it also assists users in collecting necessary data from multiple sources and compiling all relevant information into an integrated solution.

# Methodology

The proposed solution involves aggregating historical data from Burn oven sessions, utilizing NLP methods, and employing ML algorithms to train the system how to categorize termination messages into distinct categories. SME uses these outcomes to investigate the categorized clusters for permanent solutions or provide dispositions to EDT. The classification outcomes are also displayed on the web application. The system has four main parts: data collection, unsupervised clustering, supervised classification, and web application. Fig.1 shows the overall flow of the BURN Termination Disposition System.

A diagram of a computer program

Description automatically generated

**Fig.1.** Overall flow of the BURN Termination Disposition System.

## Data Collection and Extraction

The first step of building a system is collecting past error message data. This project requires a large and diverse collection of termination messages that reflect the errors and issues that can occur during the Burn oven sessions. Collecting past data from 2017 is one of the challenges in data collection. To access the information dating back to before the beginning of the project, the Microsoft SQL Server is used to perform queries for termination data from the SQL database. In the database, the raw data, which is the entire event log, is stored in chronological order in an unstructured format. These data are generated in a structured format using the Perl script, and the output is used for the unsupervised ML input data. Fig.2 shows a sample of SQL data, while Table I shows input data for unsupervised ML.

A screenshot of a computer

Description automatically generated

**Fig.2.** Sample of the SQL data.

**Table I**Input data for Unsupervised ML

|  |  |  |
| --- | --- | --- |
|  | Data | Data type |
| Output | Date | Numerical |
|  | Lot Number | Category |
|  | Oven | Category |
|  | Session | Category |
|  | Termination Message | Category |
|  | Design ID | Category |
|  | Step | Category |
|  | Group | Category |
|  | Workweek | Category |

The data pool is also updated frequently to include new and emerging cases that the historical data might not cover, allowing the system to adapt to the changing patterns and scenarios of the termination messages and ensure accuracy. For updating data frequently, a data collection and extraction process is designed to automate the generation of relevant data from the production test summary. A scheduled cronjob is executed daily to gather data on termination cases from production test summaries. It formats the information and incorporates it into the current data collection. This updated dataset is then leveraged to train ML systems and refresh the list of termination cases displayed on the Tableau dashboard. Fig.3 shows the procedures for extracting and processing new data.

A diagram of a diagram

Description automatically generated

**Fig.3.** Procedures for extracting and processing new data.

There are two types of data usage. NLP technique is applied to termination messages to change Natural Language data to vector format. Other data in Table I are used for the dashboard on the main page.

*TF (t, d) = (1)*

*IDF (t) = log (2)*

*TF-IDF (t,d) = TF (t,d) IDF (t) (3)*

The oven error message data is especially crucial to our project. However, this raw termination message data was in an unstructured Natural Language format, not vector format, posing a challenge for applying ML. To overcome this, the NLP technique TF-IDF [12] was adopted to transforming Natural Language format to a more manageable vector format. It has been one of the most famous methods for changing Natural Language format, measuring how important a term is within a document relative to a collection of documents. Words within a text document are transformed into important numbers by a text vectorization process [12]. Eqs. (1)-(3) shows how to calculate TF-IDF. Term Frequency (TF) measures the frequency of a term within a document. t means specific term and d means specific document. Inverse Document Frequency (IDF) means the rarity of a term across a collection of documents. N is total number of documents which means total number of error messages in this project. Document Frequency (df) means the number of documents with term t.

## Unsupervised Clustering

After making the vector format, the sheer volume of data makes it challenging for engineers to categorize errors individually so unsupervised clustering algorithms was required. To find the best clustering model, three clustering algorithms, including k-means clustering, DBSCAN, and Agglomerative Clustering are tested using Python.

Before validating the model, hyperparameter is set based on the elbow method [13] for k-mean and Agglomerative clustering. The elbow method is to run k-means clustering on the dataset for a range of values of *k*, and for each value of *k* calculate the sum of squared errors (SSE) [14]. The best combination of Minimum Samples [15] and Epsilon [16] for DBSCAN are chosen based on Silhouette score.

After hyper-parameter setting, the best ML clustering model is selected by maximizing the Silhouette score. Silhouette score is calculated to find the best clustering model. Eqs. (4) indicates Silhouette score.

*s(i) = (-1 s(i) 1)* (4)

For each data point i, the mean distance between the observation and all other data points in the same cluster is called the intra-cluster distance, denoted by a. The mean distance between the observation and all other data points of the next nearest cluster is called the mean nearest-cluster distance, denoted by b. The silhouette score, s, considers those inner and intra-variations and shows a result ranging from -1 to 1.

## Supervised Classification

After clustering termination messages, the number of clusters is determined. Using clustering results, Burn side relevant engineers help to advise whether the clustering is correct and label each data set. After labeling data, supervised classification algorithms are utilized to classify termination messages. Four classification algorithms, including SVC, Random Forest (RF), Naive Bayes (NB), and Logistic classifier, were tested to find the best classification model. Hyper parameter for each models are set using GridsearchCV from scikit-learn library based on the highest accuracy. After setting the hyperparameters, five-fold validation is applied to find the best ML model. The best model is selected based on F1 score.

## Building Web Application

Following selecting the optimal ML model, a web-based interface is created to provide users with an accessible means of interacting with the system. The web application is structured into two primary segments: the frontend and the backend. The user provides the necessary information on the frontend through a web page. Then, the backend fetched data about the terminated lot information (which includes the termination event log) from the oven mySQL database or test summaries using the provided inputs. It then activates ML algorithms to analyze and recommend disposition.  Subsequently, the disposition advice output containing termination details is relayed from the backend to the frontend and presented to the user via an email report format. Upon receiving a trigger from the frontend to send an email, the Backend script is activated to carry out this function. Fig. 4 provides an overview of the web application.

A diagram of a computer

Description automatically generated

**Fig.4.** Overview of web application.

The project involves creating a web app via Flask, known for being an uncomplicated yet adaptable framework. It does not demand complex setups or extensive unnecessary code. Flask interacts with the HTML using routes and views. The frontend design uses CSS, JavaScript, and HTML, while the backend functionality relies primarily on Python and Perl scripts. User actions are triggered through buttons, input fields, or select options on webpages, which then prompt processes on the Backend side. Moreover, the developers employed the "Supervisor" utility to control and monitor the application's functions, ensuring continuous operation and automated restoration in case of unforeseen terminations or malfunctions of the web application. Furthermore, the application was designed to address a basic requirement for disposition, and then underwent numerous modifications and customizations to cater to the specifications of the EDT, particularly concerning the email reporting feature.

# Result and Discussion

This section discussed outcomes from data collection and extraction, unsupervised clustering, and supervised classification. Outcomes of the web application were then examined in the following section separately.

## Automatic Data Extraction

A Perl script is generated to automatically extract data from our server. The script extracted four years of NAND and mNAND data (2,935) from the Singapore site database for the initial big data crunching stage. TF-IDF is applied to error messages to change the data format to vector format, which is input data for unsupervised clustering.

## Unsupervised ML result

K-Means, DBSCAN, and Agglomerative models are considered to determine the best clustering model. Hyper-parameter tuning was conducted on each before the models were tested.

**Table II**  
Silhouette Score and Hyperparameter

|  |  |  |
| --- | --- | --- |
| Clustering  Algorithm | Silhouette  Score | Hyper Parameter |
| K-means | 0.63 | Number of Cluster = 28 |
| DBSCAN | **0.67** | **Epsilon = 0.8, minimum sample =10** |
| Agglomerative | 0.59 | Number of Cluster = 28 |

Table II shows the Silhouette score and hyperparameters. Epsilon means how close points should be to each other to be considered a part of a cluster. Minimum sample means the minimum number of points to form a dense region [16]. The best model is DBSCAN based on Silhouette score. Therefore, the DBSCAN will be used for clustering.

As a result of applying DBSCAN, the data is categorized into 31 groups, including ‘NONE’ cases that are not categorized. (Total data set: 2,935 / number of ‘NONE’ cases: 353, amount of data for clustering: 2,582). After clustering dada, this data is labeled and validated by actual TSE team users used for supervised classification.

## Supervised ML result

After labeling data from relevant engineers, supervised ML models are applied. RF, Multinomial NB, Logistic Regression, and SVC are considered to determine the best clustering model. Hyperparameter tuning was conducted on each model before the models were tested. After setting hyperparameters, fivefold validation is applied to each model to choose the best model based on F1 score.

Table III and Fig.5 show the F1 score and hyperparameters. ‘n\_estimators’ means the number of trees in the forest. ‘max\_depth’ is the maximum depth of the tree. Alpha means additive smoothing parameter. C means the inverse of regularization strength. Gamma means Kernel coefficient.

All of the models have high F1 score. SVC is selected for web application among classification models because it has the highest F1 score and the lowest variation.

**Table III**  
F1 Score and Hyperparameter

|  |  |  |
| --- | --- | --- |
| Classification  Algorithm | F1 Score | Hyper Parameter |
| RF | > 0.98 | n\_estimators=200, max\_depth=30 |
| Multinominal NB | > 0.94 | alpha = 1 |
| Logistic Regression | > 0.96 | C = 1000.0, penalty = l1, solver = liblinear |
| SVC | **> 0.98** | **C = 100, gamma = 0.1, kernel = rbf** |

A diagram of a group of colored squares

Description automatically generated with medium confidence

**Fig.5.** F1 score of models

# Disposition of Termination Assistant (DoTA)

A web application is developed to link users with a system machine using a graphical user interface (GUI) called the "Disposition of Termination Assistant" (DoTA). The DoTA application, a First-of-A-Kind (FOAK) automated disposition advisor system, has been widely utilized by both the EDT and TSE teams, providing several vital features specifically designed for their needs:

## Termination Information Query and Disposition Advice

The system supports two modes: real-time termination (labelled as RTSUMS) and historical analysis (labelled as TSUMS). EDT make extensive use of RTSUMS mode to obtain recommended dispositions to try before escalating to TSE, while TSUMS mode is supported for TSE to investigate errors in past termination cases. Once the user inserts two required inputs as shown in Fig.6, the system automatically detects the mode and carries out different query methods for each mode. The system retrieves data from the summary uploaded to backend databases for TSUMS mode, and it accesses the mySQL database for RTSUMS mode to obtain real-time outcomes.

A screenshot of a computer

Description automatically generated

**Fig.6.** Two user inputs are needed to query termination information.

After the system engine completes all queries for information needed, the query result page, as illustrated in Fig.7, is presented to the user. The result page displays test session information, errors, and termination details to assist users in diagnosing and understanding issues. The suggested dispositions on the result page can be referred to by EDT/TSE for the next step in session recovery. As shown in Fig.7, by default, errors are listed in descending order to enhance brevity, displaying only the latest error to the user to ensure page readability. However, users can take advantage of the "Show all results" option to view a comprehensive list of errors and the "Sort" feature to arrange them in ascending order, enhancing flexibility during whole session investigations and sequence analysis.

A screenshot of a computer

Description automatically generated

**Fig.7.** Query result page returns session info and termination details.

## Email Reporting for Termination Cases

The EDT team employs the DoTA application to email pertinent stakeholders (such as TSE, SIG, Process, HyCu, and MFG teams) according to label group after they carry out the suggested disposition, but the termination problem persists. DoTA automatically generates an email regarding the latest error that leads to a specific lot being terminated and includes MAM details for that lot. For greater reporting adaptability, extra configurations are available on the query results page (see Fig.8). EDT can choose which particular errors they want to report to the concerned stakeholders. Moreover, if several partner lots are processed in the same session, DoTA allows to include all the associated lot numbers in a single email.

A screenshot of a computer

Description automatically generated

**Fig.8.** Additional setting for email drafting.

The email preview page is depicted in Fig.9. There are three different options available for sending an email:

1. Send to EDT (cc TSE): EDT is the main addressee of the email, and the email's contents appear to be instructions from TSE directed to the EDT team.
2. Send to TSE (cc EDT): The email is primarily sent to TSE, and the body contains a request for TSE to look into the termination as directed by EDT.
3. Send to Me Only: The email is dispatched to a sole EDT member based on the provided username. The EDT member then revises and forwards the content to the relevant stakeholders.

A screenshot of a email

Description automatically generated

**Fig.9.** EDT can review or modify email content before sending out.

Using the email from option 2) for illustration, the recipient receives an email resembling Fig.10. In addition to lot information and termination specifics, the system verifies whether the lot contains hot lot attributes (such as priority or reason) or if it is psel-overridden. Essential details regarding the lot are included in the email's subject or body to ensure team awareness. In option 3), the intended recipients are listed at the beginning of the email body (see Fig.11). EDT replicates this list of recipients when forwarding the edited email.

A screenshot of a computer

Description automatically generated

**Fig.10.** Example of email sent out when using option 2) above.

A screenshot of a computer

Description automatically generated

**Fig.11.** Actual recipient list is inserted into email content for option 3).

## Dashboard for Termination Case Analysis

Tableau is configured to develop a dashboard focused on analyzing termination cases. Its purpose was to facilitate early detection of terminations occurring in the production line and to improve the overall process of investigation and analysis. The dashboard displays three types of data:

1. Termination count by workweek (Fig.12): To display the weekly count of termination cases for each product technology.
2. Termination count by label (Fig.13): To display the count of termination incidents within each cluster over the past six months, with an option to filter by product technology.
3. Termination count by design ID (Fig.14): To display the count of termination cases per design ID over the last 6 months.

A graph with blue lines and numbers

Description automatically generated

**Fig.12.** Termination count for each product technology by workweek.

A graph with different colored bars

Description automatically generated with medium confidence

**Fig.13.** Termination count by label for past six months.

A graph with blue and orange bars

Description automatically generated

**Fig.14.** Termination count by design ID for past six months.

## Past Cases Documentation

Users may look at the historical case termination outcomes by technology clusters and contribute the most recent cases for ongoing reference within the DoTA system as demonstrated in Fig.15.

A screenshot of a computer

Description automatically generated

**Fig.15.** View the past documentation for each cluster.

# Further Optimization

For further improvements, two areas can be addressed: 1) the data collection and processing methods for Ambyx5 devices, and 2) the ML model calibration and validation for different product types and test programs.

## AMBYX5 Legacy Devices Support

DoTA offers complete support exclusively for Ambyx6 products, while Ambyx5 devices (legacy products) is not fully supported. DoTA provides partial support to Ambyx5 by acquiring TSE contact details, automating the subject line and session particulars, as well as retrieving MAM data, which is depicted in Fig.16. However, termination specifics like label, error descriptions, proposed resolutions, and register details are not available to send back to users, so those data points remain blank or are marked with "XXXXXX" in outgoing emails. This is primarily due to discrepancies between tester and program setups, which result in differing format of termination event logs in SQL or test summaries when compared with existing models. Although ML algorithms and the email reporting system are compatible across all devices, obstacles remain in data collection, extraction, and processing methodologies. Theses discrepancies necessitated the construction of an entirely new system to achieve full support for Ambyx5.

Although the team is still developing a permanent solution, the team has set up an interim measure that features a special function enabling users to choose a label, which consequently refines the recipients list. In this scenario, EDT would choose to send the email to themselves to customize editing with termination details. Other the other hand, there is no exigency to adjust newer DID/ovens provided that their event log format remains consistent with Ambyx6 standards. Nonetheless, the team is prepared to reassess this stance and make any necessary modifications responsive to changing conditions.

A screenshot of a computer

Description automatically generated

**Fig.16.** Partially support for Ambyx5 legacy devices.

## Enhancement of ML Calibration Model

One of the clusters from the classification is labeled ‘NONE,’ indicating that the cases could not yet be classified into any valid category. The disposition for this class will be directly reported to TSE members for now. Therefore, the DoTA team is working to group these termination cases into new categories with better resolutions by conducting quarterly reviews with respective product team members to collect their feedback and inputs for new classification and disposition.

Currently, the existing system is fully equipped to handle BURN, MPTA, and CHAMBER processes for NAND, MNAND, and DRAM products. However, the setup and ML models—such as clustering and labeling — are tailored to the BURN/MPTA steps of NAND and MNAND products. Differences in programmatic requirements between DRAM products or CHAMBER steps and the BURN/MPTA steps for NAND and MNAND may lead to inaccuracies in outcomes, including termination messages, labels, test registers, alongside contact details and attributes needed for reports. At present, a majority of the disposition communications for DRAM products or steps involving CHAMBER still necessitate involvement with TSE. To ensure higher precision in data, efforts are being made to recalibrate the system using a broader dataset from various products and processes. This initiative may engage members who specialize in DRAM devices and the CHAMBER process to contribute to refining the clustering process and providing disposition recommendations.

# Conclusion

The DoTA application was formally launched for the EDT and TSE team in Micron Singapore Backend (MSB) as of February 2024. It had achieved full adoption by MSB's EDT, serving as the sole advisor for dispositions and reporting, effectively replacing the previous manual method of generating emails. Following the achievement in MSB and with intentions to apply same tool globally, this application was presented to the Micron Memory Penang (MMP) EDT team in July 2024. Now, it is undergoing a trial period for the MMP EDT team.

The deployment of the DoTA application led to notable improvements in classifying error messages from the Burn oven. Firstly, the system enhanced the efficiency of resolving Burn termination incidents by delivering preliminary solutions to the EDT, thereby boosting work efficiency and lessening the burden on TSE/EDT. By utilizing the DoTA application, the wait time for termination cases arising during weekends was cut down from 1-2 days to mere minutes. A further advantage introduced by DoTA is its capability to auto-generate detailed report emails for termination instances, reducing the workload of EDT as they no longer must compose each email manually from scratch, with only minor adjustments required for particular cases. The time required to prepare an email has been reduced from approximately 10 minutes to under one minute in most instances. On top of that, the reporting emails are now uniform and consistent throughout every team. Furthermore, the TSE team gained the advantage of monitoring the primary pareto of termination clusters or design IDs with a higher incidence of terminations directly through the Tableau dashboard. They can also assess uncategorized (NONE) cases, as these terminations could indicate new issues in the test program. This information enabled them to effectively investigate and resolve issues, thereby reducing the number of termination cases. Moreover, new members of the TSE team could understand the dispositions for each category by reviewing past case files, reducing the need for direct supervision from senior colleagues.

# Acknowledgment

Deep gratitude goes first to the EDT team which helped to provide support and feedback during the system design and testing process. Appreciation also extends to NAND/mNAND TSE team members who provided dispositions for each error cluster to the project team.

References

[1] C.-F. Chien, C.-Y. Hsu, and K.-H. Chang, “Overall Wafer Effectiveness (OWE): A novel industry standard for semiconductor ecosystem as a whole,” *Computers & Industrial Engineering*, vol. 65, no. 1, pp. 117–127, May 2013, doi: 10.1016/j.cie.2011.11.024.

[2] A. A. Nuhu, Q. Zeeshan, B. Safaei, and M. A. Shahzad, “Machine learning-based techniques for fault diagnosis in the semiconductor manufacturing process: a comparative study,” *J Supercomput*, vol. 79, no. 2, pp. 2031–2081, Feb. 2023, doi: 10.1007/s11227-022-04730-x.

[3] “Machine Learning-Based Process-Level Fault Detection and Part-Level Fault Classification in Semiconductor Etch Equipment.” https://ieeexplore.ieee.org/abstract/document/9740077 (accessed Sep. 28, 2023).

[4] I. K. Singgih, “Production Flow Analysis in a Semiconductor Fab Using Machine Learning Techniques,” *Processes*, vol. 9, no. 3, Art. no. 3, Mar. 2021, doi: 10.3390/pr9030407.

[5] “Advanced Outlier Detection Using Unsupervised Learning for Screening Potential Customer Returns.” https://ieeexplore.ieee.org/abstract/document/9325225/ (accessed Oct. 04, 2023).

[6] “A Gaussian Mixture Model Clustering Ensemble Regressor for Semiconductor Manufacturing Final Test Yield Prediction.” https://ieeexplore.ieee.org/abstract/document/9337923/ (accessed Oct. 04, 2023).

[7] C.-Y. Hsu and W.-C. Liu, “Multiple time-series convolutional neural network for fault detection and diagnosis and empirical study in semiconductor manufacturing,” *J Intell Manuf*, vol. 32, no. 3, pp. 823–836, Mar. 2021, doi: 10.1007/s10845-020-01591-0.

[8] J.-C. Chien, M.-T. Wu, and J.-D. Lee, “Inspection and Classification of Semiconductor Wafer Surface Defects Using CNN Deep Learning Networks,” *Applied Sciences*, vol. 10, no. 15, Art. no. 15, Jan. 2020, doi: 10.3390/app10155340.

[9] J. Shim, S. Kang, and S. Cho, “Active cluster annotation for wafer map pattern classification in semiconductor manufacturing,” *Expert Systems with Applications*, vol. 183, p. 115429, Nov. 2021, doi: 10.1016/j.eswa.2021.115429.

[10] “Machine Learning-Based Detection Method for Wafer Test Induced Defects | IEEE Journals & Magazine | IEEE Xplore.” https://ieeexplore.ieee.org/abstract/document/9375473 (accessed Nov. 15, 2023).

[11] “Intelligent Fault Analysis Decision Flow in Semiconductor Industry 4.0 Using Natural Language Processing with Deep Clustering.” https://ieeexplore.ieee.org/abstract/document/9551492/ (accessed Oct. 04, 2023).

[12] “TF-IDF — Term Frequency-Inverse Document Frequency – LearnDataSci.” https://www.learndatasci.com/glossary/tf-idf-term-frequency-inverse-document-frequency/ (accessed Nov. 29, 2023).

[13] “Stop Using Elbow Method in K-means Clustering | Built In.” https://builtin.com/data-science/elbow-method (accessed Apr. 08, 2024).

[14] 262588213843476, “Using the elbow method to determine the optimal number of clusters for k-means clustering,” *Gist*. https://gist.github.com/rpgove/0060ff3b656618e9136b (accessed Apr. 09, 2024).

[15] J. Sander, M. Ester, H.-P. Kriegel, and X. Xu, “Density-Based Clustering in Spatial Databases: The Algorithm GDBSCAN and Its Applications,” *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 169–194, Jun. 1998, doi: 10.1023/A:1009745219419.

[16] “DBSCAN Python Example: The Optimal Value For Epsilon (EPS) | by Cory Maklin | Towards Data Science.” https://towardsdatascience.com/machine-learning-clustering-dbscan-determine-the-optimal-value-for-epsilon-eps-python-example-3100091cfbc (accessed Aug. 23, 2024).

[17] K. S. do Prado, “How DBSCAN works and why should we use it?,” *Medium*, Jun. 03, 2019. https://towardsdatascience.com/how-dbscan-works-and-why-should-i-use-it-443b4a191c80 (accessed Apr. 08, 2024).

Biography

**Mun Sik Park,** Master’s Degree in Chemical Engineering from NUS in 2023. Joined Micron TSE team since Dec 2022 and he is currently TSE who has experience in Probe testing. Specialized in applying ML and Multi-objective optimization algorithms to solve practical problems.

**Choon Hian, Chai,** Bachelor’s Degree in Electrical and Medical Electronics Engineering from University of Technology Malaysia in 2011. Joined Micron TSE team since July 2011 and he is currently Test Solutions Engineering Manager who has experience in Probe testing and Backend Burn-In testing across multiple product technologies covering DRAM, NAND and mNAND.

**Fei Keng Tan,** Bachelor’s Degree in Electronic Engineering from University of Technology Malaysia in 2018. She started her work in Micron Semi Asia PTE Ltd, Singapore since September 2018 as a Test Engineer, and is currently Senior Test Solutions Engineer Lead who has experience in Probe testing and Burn-In testing.

**Kin Keong, Leong,** Bachelor’s Degree in Electrical Engineering from University of Malaya, Malaysia in 2007. Started working at Micron Semi Asia PTE Ltd, Singapore in June 2007 as a Test Engineer, and is currently Principal Test Solutions Engineer focusing on test solutions development at Ambyx and Axcel platforms.