Proceedings of the ASME 2023 International Technical Conference and Exhibition on Packaging and Integration of Electronic and Photonic Microsystems
InterPACK2023
October 24-26, 2023, San Diego, California

IPACK2023-111525

EXPLAINABLE MACHINE LEARNING APPROACH TO YIELD AND QUALITY IMPROVEMENTS USING DEEP TOPOLOGICAL DATA ANALYTICS

Janhavi Giri
Intel Corporation
Santa Clara, CA

Attila Lengyel
Intel Corporation
Santa Clara, CA

ABSTRACT

In wafer fabrication, data is collected and analyzed to prevent process deviations that could affect product quality and wafer yield. However, the high-dimensional, sparse, and imbalanced nature of the data poses significant challenges to yield and quality root cause analysis. Deep Topological Data Analysis (DTDA) is an unsupervised machine learning method that clusters and models the data in the form of geometric objects such as graphs and their higher-dimensional versions. This method reduces the multidimensional dataset to two-dimensional networks or graphs, where each node represents a cluster of samples with similar characteristics, and an edge represents the presence of overlapping characteristics between the connecting nodes. DTDA provides insights into the necessary data elements required to conduct accurate analysis and helps engineers identify the features contributing to yield and quality issues, enabling corrective actions. Moreover, the approach prevents the waste of engineering resources and mitigates the impact on final manufacturing cost.

Keywords: topological data analysis, feature selection, high dimensional, wafer manufacturing, big data, yield and quality improvements

1. INTRODUCTION

A vast amount of data is collected in wafer fabrication and routinely analyzed to ensure that there are no process deviations that may result in loss of product quality and wafer yield. Characterizing and determining the factors which cause these deviations is crucial to drive corrective actions and improve production cost.

Big Data shifted how we collect, store, and analyze data and enables engineers to link an infinite number of features to a single wafer/die. With the number of wafer process steps above a thousand, engineers can link tens of thousands of features together to describe a wafer/die, and it is not even close to what maximally possible. A dataset that has more features (columns) than data points (rows) is often referred in the literature as high dimensional data. This high dimensionality is more critical at the new product introduction (NPI) phase where we have limited number of wafers as data points and among those wafers even lower number of wafers with quality or yield loss issues. The low number of data samples with less than 5% of positive labels adds significant complexity to the analytics and described as highly imbalanced. Due to the high variations of wafer workflows, especially in NPI phase, we also have features that are not present for a handful of wafers increasing the sparsity of the analyzed data.

To describe yield and quality root cause analysis in term of data analytics, we can state that the challenge is to find a set of features in a highly dimensional, imbalanced, and sparse dataset that provide statistical confidence in predicting positive labels. Any one of these properties provide a unique challenge for data scientist, but the combination of the three is why quality and yield analytics is so extremely difficult to approach by established AI/ML approaches. The conventional methods used for dimensionality reduction, feature selection, and visualizing such high dimensional dataset include principal component analysis (PCA), partial least squares (PLS), Uniform Manifold Approximation and Projection (UMAP), and t-distributed stochastic neighbor embedding (t-SNE). However, these methods are known to be sensitive to noise and outliers, are linear, require specific probability distributions and perform poorly with sparse and imbalanced data. Another requirement for these types of analytics is that engineers are not only interested in the features that can predict a certain type of failures, but they must understand why, so they can execute corrective actions. Explainable AI/ML (XAI) is a new field in data science with several on-going research studies and of high importance in making industrial AI/ML applications successful.

In this paper, we propose a novel approach to yield and quality improvements utilizing topological data analysis. Topological data analysis (TDA) is an unsupervised machine learning approach which aims to determine the unknown topology of the high-dimensional manifold where the data resides to extract the hidden patterns [1] [2]. TDA defines the relationship between shapes through abstract topological shapes instead of traditional geometric meanings. TDA when combined with deep generative models results in a unique XAI technique referred as Deep Topological Data Analysis (DTDA) that identifies hidden structures within dataset, clusters them according to the patterns found, explains which features of the data contributed to the formation of these clusters and how these features are correlated [3]. DTDA uses the Vietoris-Rips algorithm [4] to construct nearby data points to build topological structures, and nested complexes are used to identify persistent elements of the data structure using Morse theory [5]. Finally, the manifolds of original dimensions are simplified and visualized [6].

Our main objective is to demonstrate the effectiveness of the DTDA approach in improving yield and quality in manufacturing processes while providing a clear and interpretable explanation of the decision-making process. The remainder of the paper is organized as follows. In Materials and Methods section, we introduce the proposed methodology, including the dataset used, pre-processing steps, machine learning method used, and the explainability measures employed. In the Results and Discussion section, we present the findings from the use cases investigated and discuss the performance of the chosen method in terms of yield and quality improvements and the interpretability of the model using DTDA. Finally, the conclusion section provides a summary of the study, its contribution to the field, and implications for future research.

2. MATERIALS AND METHODS

Wafer yield and quality management is a critical task for semiconductor manufacturers. A lot of effort has been conducted in both industry and academia to improve the wafer yield. However, existing yield and quality management approaches typically only consider a single process or a few processes. A more comprehensive approach is needed to fully utilize the potential of process operational data to improve wafer yield. This approach should consider all the processes involved in semiconductor manufacturing, as well as the interactions between these processes. It should also be able to identify and respond to potential yield issues in real time.

One of the very well-known challenges here is first, the sheer volume of data that needs to be collected and analyzed. Another challenge is the complexity of the semiconductor manufacturing process. Effective data mining approaches are thus required at various stages of the semiconductor manufacturing life cycle. As indicated in the review article [7] smart data mining approaches can help significantly improve yield, process control and product development. Finally, it is important to develop an approach that is cost-effective, robust, and scalable.

With these requirements and limitations understood of the conventional approaches for yield and quality analytics, we decided to look for alternative method that could address these challenges. We learned about Topological Data Analysis (TDA), an unsupervised machine learning method widely utilized to extract insights from high-dimensional data. The fundamental notion in TDA is that data has shape and shape has meaning [1]. It aims to reveal crucial connections and dependencies of the input data.

TDA when applied to high dimensional data, reduces the muti-dimensions to two dimensional networks composed of nodes (or clusters) of similar samples and edges connecting the nodes with overlapping characteristics. TDA segments and helps to determine features contributing to the given segmentation. Furthermore, this approach could be extended to determine which features are relevant to the outcome (or target). For example, Guo et. al [8] demonstrated the application of TDA in detecting faults of semiconductor manufacturing process which are known to be difficult using traditional methods. They were successful in generating topological networks that captured the intrinsic data separation demonstrating that the input data was coming from different experiments and identified the connections that existed among the clusters of samples in these networks. Furthermore, from analyzing the shape of these networks they determined the top contributing process variables or features that were responsible for the failing wafers. The predictive model built using these selected features resulted in high prediction accuracy. In our exploration of TDA, we were successful in replicating their results however, we did observe few challenges working with the Mapper open-source library [9] [10] which was utilized by Guo et. al [8] in their study. The Mapper library requires user to pick several parameters which is mostly trial and error with no defined metric to determine which one of the obtained TDA maps is the actual representation of the input data. Secondly, there is a version dependency which creates an issue with reproducibility. Furthermore, the post-processing of the resulting TDA maps is manual to obtain quantitative learnings. We, therefore, investigated other python based open-source packages besides Mapper and realized the drawbacks of working with open-source TDA packages. These packages would work fine for exploration purposes with small datasets and for POCs but were devoid of readiness for enterprise level applications. They lacked scalability, robustness, and could not handle big data i.e., order of millions of rows and columns. We therefore explored the commercially available TDA platforms. There were two enterprise ready solutions which were available in market. After thorough analysis, we picked DataRefiner [3] as the platform for conducting topological data analysis. In our collaboration with DataRefiner team, we developed several customized features in the platform that would help with the post-processing and interpreting of the TDA maps for semiconductor manufacturing yield and quality use cases.

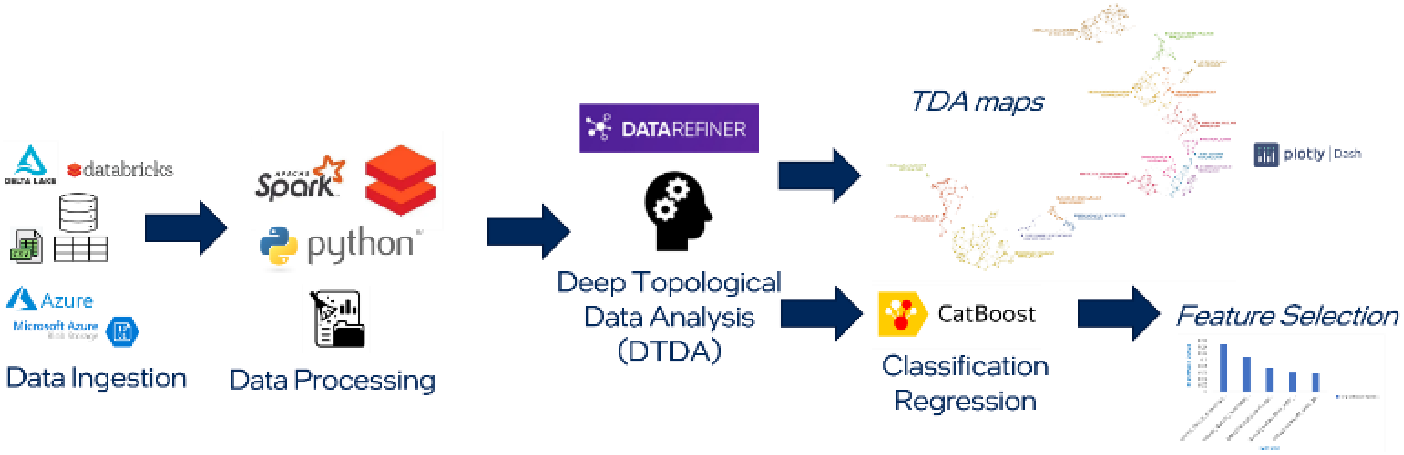

FIGURE 1: HIGH DIMENSIONAL VISUAL ANALYTICS PLATFORM DEMONSTRATING THE FRAMEWORK FOR DEEP TOPOLOGICAL DATA ANALYSIS.

2.1 Pipeline for analysis and modeling using DTDA

FIGURE 1 illustrates the overall approach for high dimensional visual analytics. DataRefiner's [3] Deep Topological Data Analytics (DTDA) engine is utilized for data mining, visualization, feature selection for predictive modeling from the high dimensional, sparse, and extremely imbalanced datasets encountered in semiconductor manufacturing.

The following sections present the detailed step-by-step process shown in FIGURE 1 consisting of four phases: (A) data ingestion, (B) data processing, (C) deep topological data analysis, and (D) post-processing (TDA maps/Feature selection).

A. Data Ingestion: The platform accepts tabular data in form of csv files which a user could upload straight to the platform, or it could be pipelined through an on-prem database or cloud storage such as Microsoft Azure blob storage and Databricks Delta Lake tables.

B. Data Processing: The input variables of the dataset comprise of various data subsets corresponding to wafer manufacturing for ex. the toolset, recipe, sensor parameters, electrical measurements, metrology measurements etc. The dimension of dataset can span from several hundreds to 10k+ columns/rows. The dataset therefore includes both categorical and numerical datatypes. To prepare these datasets for analysis, pre-processing and cleaning are performed utilizing Databricks Spark and Python libraries. Columns with 100% null values are dropped. Based on use case, rows containing one or more columns with null values could be dropped as well. Categorical features are one-hot encoded. When supervised training is included, the dataset is split into training and test sets where the training set is used for model optimization and selection. The platform itself performs many data verifications to ensure the integrity of the data as well as suggests user on the changes to improve the results. For example, distribution correction is an option that user can enable or disable based on their use case.

C. Deep Topological Data Analysis: It is a process of transforming pre-processed data on the previous step into a topological map for user analysis. This process can be presented as 11 step process as shown in FIGURE 2.

1. The initial step involves acquiring the source data, which can take various forms such as numerical matrices or raw text. In the case of raw text, the system internally performs necessary transformations.
2. A series of data verifications is conducted by the system to ensure data integrity while also providing suggestions to the user for improving the results.
3. An optional stage involves the application of a self-supervised deep generative model specifically designed for extracting high-level parameters from the data, enhancing the quality of data segmentation.
4. Employing a scalable nearest-neighbor algorithm utilizing gradient descent, this module enables efficient processing of vast datasets, even reaching hundreds of millions of records.
5. The system employs an iterative approach to perform topological optimization, aiming to identify and characterize homology groups, thereby representing the topological structure in both 2D and 3D spaces.
6. Validation of the resulting topological structure is carried out, involving the adjustment of meta-parameters and reiteration of steps 4, 5, and 6 to generate new candidates. The most optimal candidate is selected as the final result.
7. The final topological structure is presented to the user for analysis and interpretation.
8. The user engages in analysis of the structure, comprehending its intricacies, and may make alterations to enhance the quality of segmentation. The user can also define groups and additional structures for later supervised steps.
9. Once the structure is refined, the user can provide new data with a similar structure.
10. The learned model is then applied to the new data, yielding segmentation results based on the acquired knowledge.
11. Finally, a new labeled data file is generated, which can be conveniently accessed for download through a web interface or an API.

D. Post-Processing (TDA Maps/Feature Selection): The DTDA engine segments the data based on their shared characteristics outputting data maps or graphs. These maps are composed of nodes which are clusters containing samples from the data that exhibit similar characteristics. Edges connecting the nodes are clusters containing samples that have overlapping characteristics. Top features of the dataset resulting in data segmentation are found. Every cluster is clearly explained. Significant feature

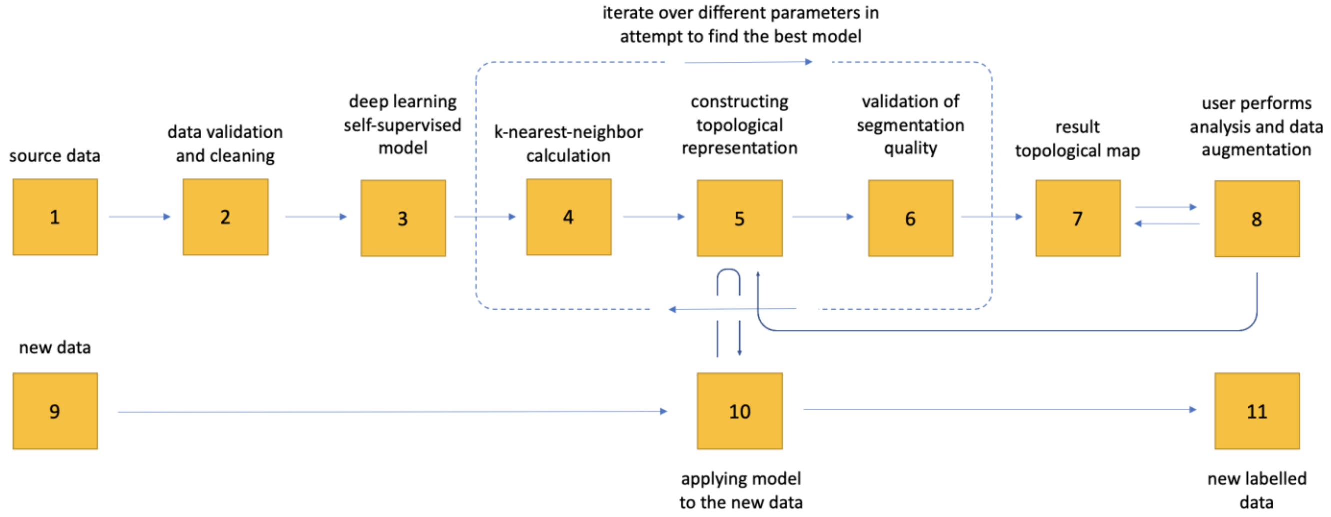
iterate over different parameters in attempt to find the best model
source data
1
data validation and cleaning
2
deep learning self-supervised model
3
k-nearest-neighbor calculation
4
constructing topological representation
5
validation of segmentation quality
6
result topological map
7
user performs analysis and data augmentation
8
new data
9
10
applying model to the new data
11
new labelled data

FIGURE 2: DATAREFINER ARCHITECTURE

correlations are determined at cluster level as well as it is possible to determine how the features are correlated at global level. Cluster to cluster comparison allows user to determine the critical features that differentiates the clusters. Precise rules that separate a selected cluster from rest of the data is also determined. The distribution of ‘target’ or ‘outcome’ is visualized by overlaying over the generated data maps. Interesting data segments could be then identified based on these overlays. Similarly, we can also determine how rest of the features are distributed
throughout the map by selecting the feature of interest in the platform giving an insight of data segments with any peculiarities. When included supervised training, key features for the outcome are also determined. Underlying supervised learning model’s performance is obtained through the learning curves, confusion matrix for classification type problems and R2-score for regression learning problems.

aim to root cause. For proprietary reasons, the exact names of variables and devices are not revealed. The input variables include process operating conditions, time spans, equipment units, and sensor parameters. These are essentially the common fields that are chosen for yield analysis. Various fabrication data are then combined with each wafer to organize the input variables in two-dimensional form such that each row corresponds to a wafer and columns correspond to the process flow variables. The data are restricted by the wafers of a specific product with ‘target’ defined for supervised learning, wafer failing/passing the pre-decided criteria are encoded as ‘1’ for fail and ‘0’ for pass. The use cases described here correspond to reported wafer failures observed end of line. The challenge here was to determine the root causes of failures considering the underlying imbalance, high dimensional, and sparsity of the datasets.

The first use case comprised of 2952 wafers (rows) and 974 wafer process flow steps (columns). Out of these 2952 wafers, there were 15 wafers that were failing which was only 0.5%. Our hypothesis was if the given dataset contains one or specific combination of process flow steps which are causing the failure then we should be able to identify those through the unsupervised

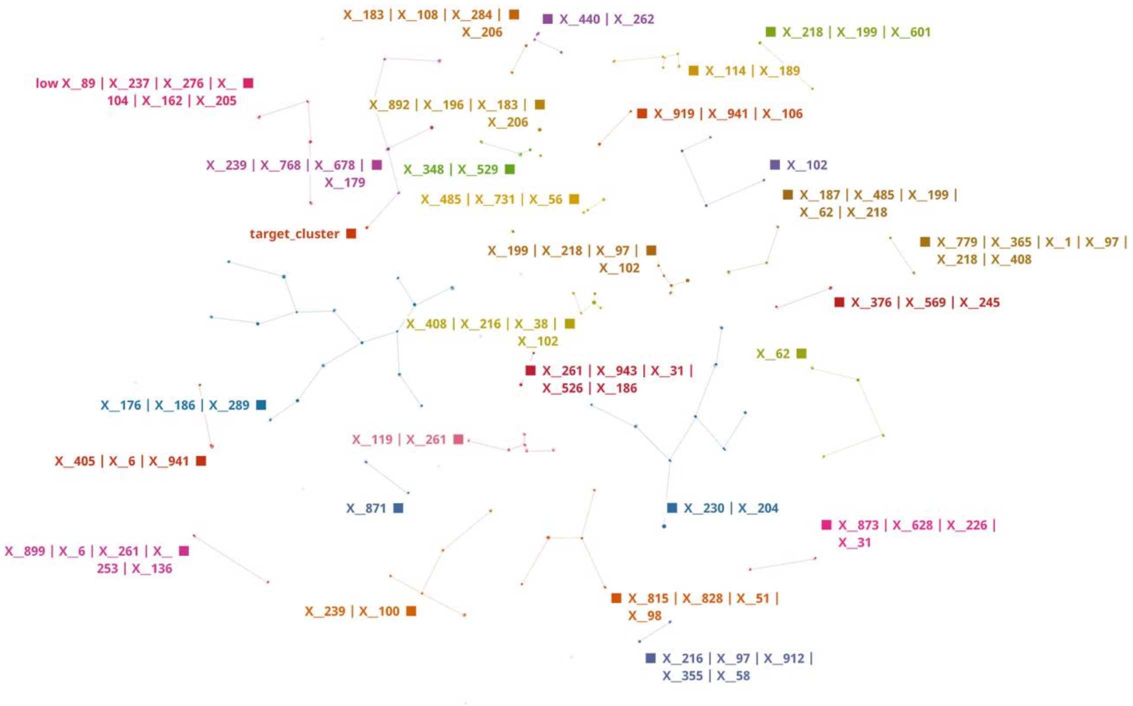
X_183 | X_108 | X_284 | X_206
X_440 | X_262
X_218 | X_199 | X_601
X_114 | X_189
low X_89 | X_237 | X_276 | X_ 104 | X_162 | X_205
X_892 | X_196 | X_183 | X_206
X_919 | X_941 | X_106
X_239 | X_768 | X_678 | X_179
X_348 | X_529
X_102
X_485 | X_731 | X_56
X_187 | X_485 | X_199 | X_62 | X_218
target_cluster
X_199 | X_218 | X_97 | X_102
X_779 | X_365 | X_1 | X_97 | X_218 | X_408
X_376 | X_569 | X_245
X_408 | X_216 | X_38 | X_102
X_62
X_261 | X_943 | X_31 | X_526 | X_186
X_176 | X_186 | X_289
X_119 | X_261
X_405 | X_6 | X_941
X_871
X_230 | X_204
X_873 | X_628 | X_226 | X_31
X_899 | X_6 | X_261 | X_ 253 | X_136
X_815 | X_828 | X_51 | X_98
X_239 | X_100
X_216 | X_97 | X_912 | X_355 | X_58

FIGURE 3: TOPOLOGICAL MAP DERIVED FOR THE USECASE #1 DATASET. EACH NETWORK IS NAMED BASED ON THE RELEVANT FEATURES FOR THAT NETWORK. NODES CORRESPOND TO WAFER SAMPLE CLUSTERS WITH CONNECTING EDGES INDICATING OVERLAPPING CHARACTERISTICS. THE ‘TARGET’ CLUSTER COMPRISES OF ALL THE 15 FAILED WAFERS.

3. RESULTS AND DISCUSSION

We intent to test the hypothesis with empirical data that the given dataset contains one or specific combination of process flow variables that are resulting in the observed failure that we

DTDA approach. Our approach successfully clusters all failed wafers into a single cluster fully unsupervised. The resulting TDA map from this dataset is shown in FIGURE 3, where the cluster 'target' contains all the 15 failed wafers.

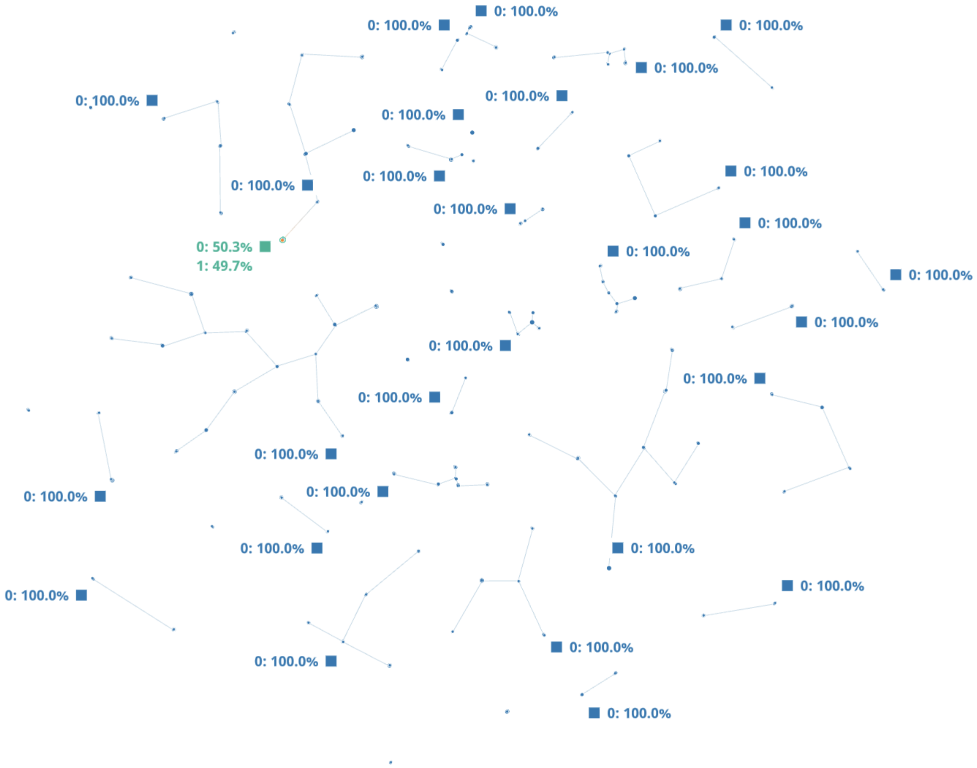

FIGURE 4: DISTRIBUTION OF TARGET IN EACH CLUSTER. TARGET CLUSTER (IN GREEN) CONTAINS THE 15 FAILED WAFERS.

On overlaying the 'target' feature values we can clearly see in FIGURE 4 that 100% of the failed wafers aggregate into one cluster. We also identified the topmost contributing feature for the observed data separation, as seen in the FIGURE 5 where the 'target' cluster corresponds to the maximum value of the X_470 feature.

On further zooming into the 'target' cluster, we were able to determine the specific features and their combinations that were resulting into the clustering which were X_101, X_156, X_517, X_452, X_679 etc. To determine which features of these identified as relevant for the overall TDA map are important to the outcome i.e., failed vs normal wafer, we further applied supervised machine learning method CatBoost which is an open-source library that uses gradient boosting on decision trees [11] for both regression and classification tasks. The feature selected as important for the outcome are shown in FIGURE 6. These features were further confirmed to be important by the respective process owners as top contributors to the observed failures.

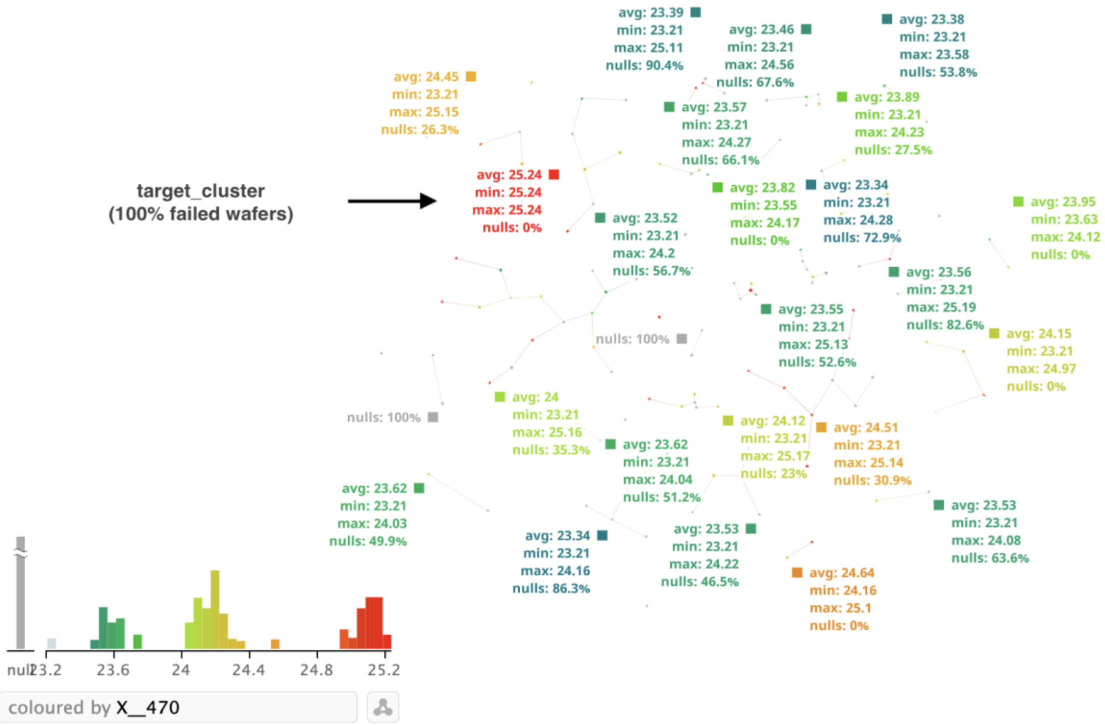

FIGURE 5: DISTRIBUTION OF THE FEATURE X_470 IDENTIFIED AS THE TOP CONTRIBUTOR TO THE OBSERVED DATA SEPARATION. THE TARGET CLUSTER WITH ALL THE FAILED WAFERS CORRESPOND TO HIGHEST RANGE OF VALUE FOR X_470. COLORS ARE IN SYNC WITH THE RANGE OF VALUES FOR X_470, FOR EX. GREY COLOR REPRESENTS NULL VALUES AND RED COLOR IS USED TO REPRESENT MAXIMUM VALUE.

The second use case that we investigated using this DTDA comprised of 11,468 wafers with 962 process flow steps. Out of this 11k+ wafers, there were 1137 wafers that were failing their metrics. The challenge was to determine out of these 900+ process steps which one was contributing to the failed wafers. The hypothesis we were testing was same as use case 1 i.e., if the given dataset contained one of specific combination of process flow steps which are contributing to the failure, then we should be able to capture those through DTDA. In FIGURE 7, the resulting TDA map from the given dataset is shown. Interestingly, when the 'target' values are overlaid on this map in FIGURE 8 no distinct cluster which is composed of all the failed wafers is found.

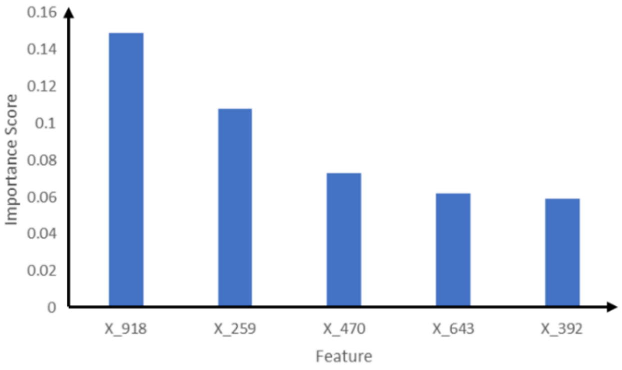

FIGURE 6: FEATURES SELECTED FROM DTDA AND SUPERVISED LEARNING APPROACH. HIGHER THE SCORE LARGER THEIR IMPACT WOULD BE ON THE OUTCOME.

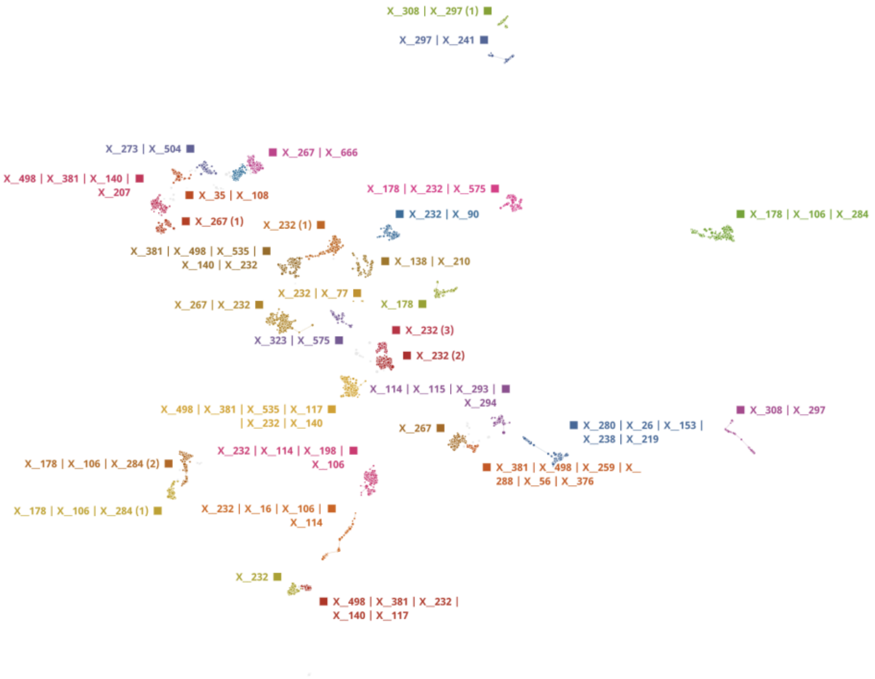
X_308 | X_297 (1)
X_297 | X_241
X_273 | X_504
X_267 | X_666
X_498 | X_381 | X_140 | X_207
X_35 | X_108
X_178 | X_232 | X_575
X_232 | X_90
X_267 (1)
X_232 (1)
X_178 | X_106 | X_284
X_381 | X_498 | X_535 | X_140 | X_232
X_138 | X_210
X_232 | X_77
X_267 | X_232
X_178
X_232 (3)
X_323 | X_575
X_232 (2)
X_114 | X_115 | X_293 | X_294
X_498 | X_381 | X_535 | X_117 | X_232 | X_140
X_308 | X_297
X_280 | X_26 | X_153 | X_238 | X_219
X_267
X_232 | X_114 | X_198 | X_106
X_381 | X_498 | X_259 | X_288 | X_56 | X_376
X_178 | X_106 | X_284 (2)
X_178 | X_106 | X_284 (1)
X_232 | X_16 | X_106 | X_114
X_232
X_498 | X_381 | X_232 | X_140 | X_117

FIGURE 7: TOPOLOGICAL MAP DERIVED FOR THE USECASE #2 DATASET. EACH NETWORK IS NAMED BASED ON THE RELEVANT FEATURES FOR THAT NETWORK.

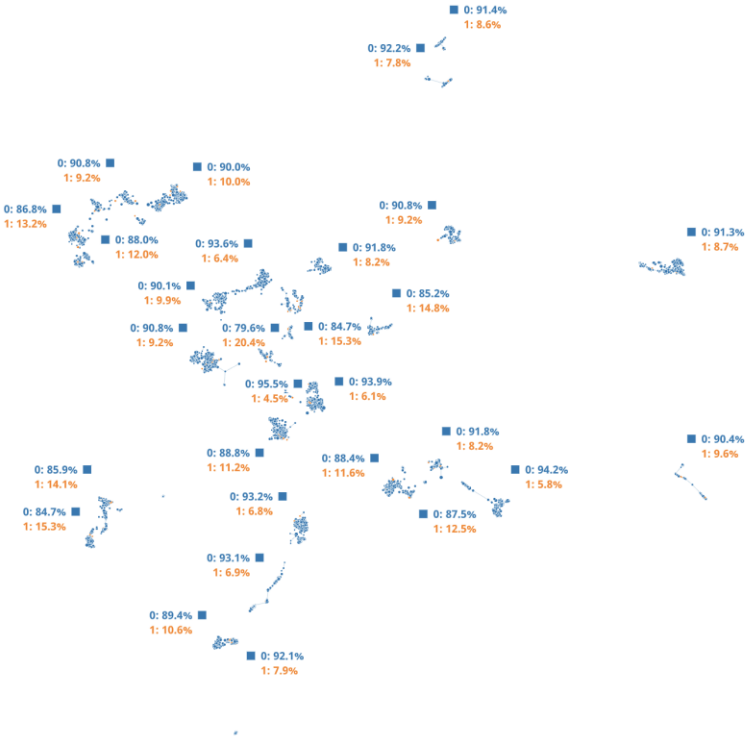

FIGURE 8: DISTRIBUTION OF TARGET VALUES THROUGHOUT THE TDA MAP, '1' INDICATES FAILED WAFER AND '0' INDICATES NORMAL WAFER.

There is an even distribution of the ‘target’ values as shown in which conveys that the dataset does not contain features that have the strongest impact and thus, fail to explain the observed wafer failures. Thus, to determine what part of the manufacturing flow resulted in these failures it needs to examine if the dataset is complete or additional data collection is required. Nevertheless, with this approach we were able to visualize the various segments of the dataset and how the ‘target’ values were distributed in those segments which would not have been possible otherwise with the classical approaches.

4. CONCLUSION

Semiconductor manufacturing is a long and complex manufacturing process through which silicon wafers are turned into electronic devices. Considering the complexity and the enormous amount of data that the manufacturing process generates, it has been becoming increasingly challenging to determine the root cause of failures. The wafer manufacturing yield and quality are significantly impacted if a single or combination of those process steps malfunction. In semiconductor manufacturing we are thus, posed with a high dimensional data problem where the datasets could have more features than the samples especially during the new product introduction phase where limited wafers are available. As a result, these datasets could be extremely imbalanced, and at times sparse. In such a situation, classical and established AI/ML approaches turn out to be ineffective in identifying the root causes for yield and quality improvement.

Deep Topological Data Analysis (DTDA) an unsupervised machine learning comes to rescue with its unique capability to extract useful patterns and determine relevant features from such high dimensional datasets. We have successfully demonstrated how our proposed framework based on DataRefiner’s DTDA engine enabled our yield & quality teams to conduct data mining, visualization, and predictive modeling for their high dimensional, sparse, and extremely imbalanced datasets. The discussed approach is applicable to structured data i.e., in form of rows and columns. In future, we intent to further refine our approach and make it applicable to unstructured data i.e., images as they are known to be major source of data in semiconductor manufacturing and are extremely valuable for various yield improvement tasks that could be automated with our approach.

ACKNOWLEDGEMENTS

The authors would like to thank Edward Kibardin, CEO and Chief Data Scientist at DataRefiner for their collaboration and technical discussions. They would also like to thank their colleagues and collaborators at Intel Corporation for their feedback and suggestions. Additionally, they would like to thank the reviewers for taking their time and effort to review the manuscript.

REFERENCES

[1] G. Carlsson, "Topology and data," Bull. Amer. Math. Soc, vol. 46, no. 2, pp. 255-308, 2009

[2] L. Wasserman, "Topological data analysis," Ann. Rev. Stat. Appl, vol. 5, pp. 501-532, 2018.

[3] "https://www.datarefiner.com/," [Online].

[4] M. K. S. P. D. &. W. C. Ahmed, "Map construction algorithms," Map construction algorithms, pp. 1-14, 2015.

[5] E. M. J. W. S. M. W. R. &. M. J. N. Stein, Morse theory, Princeton University Press, 1963.

[6] E. Kibardin, "AI for AI (artificial insemination)," 2020. [Online]. Available: https://datarefiner.com/feed/ai-for-ai.

[7] R. G. ,. a. E. M. G. R. Pedro Espadinha-Cruz, "A Review of Data Mining Applications in Semiconductor Manufacturing," Processes, vol. 9, no. 305, 2021.

[8] A. G. B. Wei Guo, "Identification of key features using topological data analysis for accurate prediction of manufacturing system outputs," Journal of Manufacturing Systems, vol. 43, no. 2, pp. 225-234, 2017.

[9] N. S. D. E. &. S. W. M. Hendrik Jacob van Veen, "Kepler Mapper: A flexible Python implementation of the Mapper algorithm (Version 1.4.1)," 2020.

[10] v. V. e. al., "Kepler Mapper: A flexible Python implementation of the Mapper," Journal of Open-Source Software, vol. 4, no. 42, p. 1315, 2019.

[11] L. G. G. V. A. D. A. V. &. G. A. Prokhorenkova, "CatBoost: unbiased boosting with categorical features," arXiv preprint, 2019.