International Journal of Electrical and Computer Engineering (IJECE)
Vol. 99, No. 1, June 2023, pp. 1∼1x
ISSN: 2088-8708, DOI: 10.11591/ijece.v99i1.pp1-1x

Enhancing the Accuracy of Manufacturing Process Error Detection through SMOTE-based Comprehensive Techniques Using Machine Learning and Deep Learning

Sai Venkat Boyapati[1], Guduru Bhanu Sri Rakshitha[1], Maryada Rishitha Reddy[1], Dr. Salati Sumalatha[2]
[1]Department of Computer Science and Engineering, SRM University, AP, India
[2]Assistant Professor Department of Computer Science and Engineering, SRM University, AP, India

Article Info

Article history:

Received month dd, yyyy
Revised month dd, yyyy
Accepted month dd, yyyy

Keywords:

Semiconductor
Manufacturing process errors
SECOM dataset
SMOTE technique
Machine Learning
Deep Learning

ABSTRACT

A production competency study leads to a rise in the manufacturing sectors' strategic emphasis. Developing Semiconductors is a highly complex approach that necessitates numerous evaluations which are stated in the SECOM Dataset. There are imbalanced statistics in this dataset, so our methodology incorporates SMOTE functionality that is introduced to mitigate the imbalance of the dataset by leveling off any unbalanced attributes. Detecting faults in the manufacturing process improves semiconductor quality and testing efficiency is used to validate Machine Learning and Deep Learning algorithms. Prior to implementing the algorithms, we were able to reduce the features 591 to 9, resulting in a training time of 2.94 seconds. However, after applying the balancing techniques, the number of features increased to 19. Despite this increase, the training time was significantly reduced to 4.41 seconds although it led to a longer training time due to the increased number of cases or samples. The random oversampling balancing technique was particularly effective in achieving the reduction of training time and also consistently produced the highest accuracy. Additionally, it is important to highlight the remarkable accuracy of the Random Forest in predicting semiconductor performance, as it consistently achieved an impressive accuracy rate of 96.37%.

Corresponding Author:

Sai Venkat Boyapati
Department of Computer Science and Engineering, SRM University, AP, India
Email: saivenkat_boyapati@srmap.edu.in

1. INTRODUCTION

The corporate setting of modern times is constantly evolving. The development of semiconductors has significantly altered our world [1]. The primary objective of semiconductor manufacturers is to raise their quality on an annual basis. Because semiconductors form the foundation of every hardware system, the demand for them has increased tremendously along with both the personal and business use of all technologies [2]. Machine learning, data mining, and deep learning offer a multitude of possibilities for the efficient management of industrial processes. By means of distributed data collection, data cleansing, extracting useful data from noisy data, and updating optimization ideas from flowing data in real-time, these technologies can provide a wealth of opportunities for industrial applications. These changes produced enormous volumes of data that need to be evaluated [3]. This article suggests machine learning and deep learning methods for evaluating manufacturing process failures. The production of semiconductors is the industry that is taken into account for the validation of these processes. Predicting process faults is crucial for reducing failure rates [4]. The SECOM dataset serves as support for comparing our suggested methodologies and is indicative of semiconductor manufacturing operations. This dataset acts as the norm for assessing if the deliverables of a series of manufacturing activities are incorrect or not. It conducts numerous tests on the semiconductor and assesses its functional capability to see if it functions properly. In our dataset, we have a significant quantity of data from several semiconductors that were undertaking these checks. The results of the tests indicate whether the semiconductors were successful or not. Below is a brief outline of this article's workflow and key contributions [5]:

1. Determining an approach that utilizes the following processes for spotting faults in the manufacturing process which are data preprocessing which involves data cleaning and it corrects the noisy data, features selection, and dataset will be splitting into train and test data [6].
2. Before doing these data preprocessing techniques SMOTE operation is applied to correct the class imbalance data.
3. The modification, applying as well as assessment of two deep learning and machine learning algorithms.

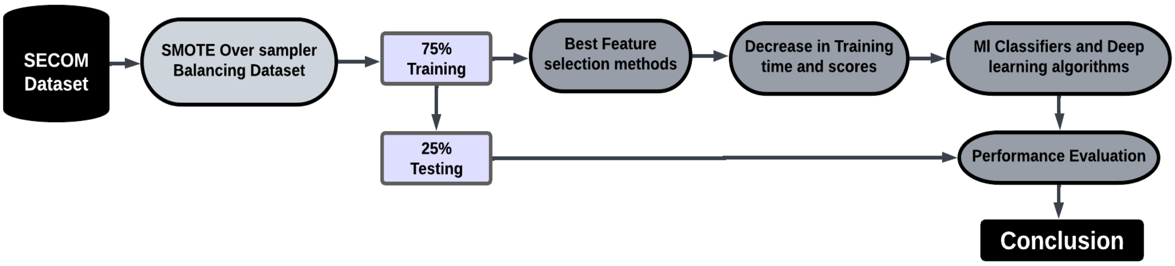
SECOM Dataset
SMOTE Over sampler Balancing Dataset
75% Training
25% Testing
Best Feature selection methods
Decrease in Training time and scores
Ml Classifiers and Deep learning algorithms
Performance Evaluation
Conclusion

Figure 1. Proposed Methodology for SECOM Dataset workflow.

Additionally, the document is structured as follows: Section 2 presents the Dataset description, Section 3 takes us through the complete workflow which is the proposed methodology, Section 4 tells about Feature Elimination Section 5 companies all the results while Section 6 presents the conclusions and at the end, our paper ends with mentioning all the references used throughout the research.

2. REVIEW OF LITERATURE

The initial literature papers about our topic actually assist us in first comprehending the importance, manufacturing techniques, and testing procedures for semiconductors. Using earlier articles as a guide, Let's now talk about the methodologies that numerous other published studies use. These studies highlight the continuous efforts to develop effective models for product quality prediction and fault detection in the semiconductor manufacturing industry.

Balancing the Dataset as the dataset is faulty data balancing is the main part. Kim et al [7] and Salem et al [8] authors employed the SMOTE technique to balance the dataset. Their results indicated that SMOTE-based oversampling could help improve fault detection in semiconductor manufacturing processes. Salem et al [8] authors tested 288 approaches to classifying the SECOM dataset using various stages for data imputation, data imbalance, feature selection, and classification. The results showed that LR was the best classification model, and SMOTE was the best technique for synthetic data generation. Moldovan et al [10] tested their method on SECOM and SETFI datasets, and found that using the CSO algorithm to determine the number of nodes in each hidden layer increased the weighted precision of the MPC.

Feature Elimination enhances model performance by selecting pertinent features and noise, resulting in improved accuracy and efficiency. Salem et al [8] authors used SELECTFDR to be effective for feature selection, and ”In-painting KNN-Imputation” was the best data imputation method. These discoveries have the potential to improve the accuracy and reliability of fault detection in semiconductor manufacturing processes. Extensive analysis of research papers revealed a focus on reducing training time by identifying relevant features. Increased feature count was consistently linked to longer training times. Consequently, feature elimination techniques were utilized to select informative features, reducing training time while maintaining model performance. Furthermore, apart from the aforementioned techniques, there exists a range of other methods for feature elimination, such as Principal Component Analysis (PCA), Multivariate Adaptive Regression Splines (MARS), and NOVA [13]. These approaches provide alternative strategies to select important features, thereby enhancing the performance of classification models across diverse domains and industries [14].

Classification is the crucial stage of evaluating product quality, and classification models have been extensively utilized. Various studies have explored different approaches, Kim et al [7] authors developed fault detection prediction models using logistic regression (LR), artificial neural networks (ANN), decision trees (DT), and random forests (RF). Kerdprasop and Kerdprasop [9] researchers have also proposed data mining-based fault detection methods, with Naive Bayes achieving a 90% fault detection rate but an 80% false alarm rate. To address this, boosting techniques were employed, resulting in improved precision for tree-based models while maintaining a low false alarm rate. Multilayer Perceptron Classifier (MPC) and Chicken Swarm Optimization (CSO) were employed in other studies by Moldovan et al [10]. Additionally, classification performance has been explored using support vector machines, logistic regression, artificial neural networks, and decision tree algorithms by Munirathinam [11] and Karthigaikumar [12].

3. EVALUATING THE DATASET

SECOM Dataset was discovered by authors Michael McCann and Adrian Johnston and was donated to the UCI Machine Learning Repository [19] on the 19th of November, 2008. This research utilizes the SECOM (Semiconductor Manufacturing) dataset, which includes both manufacturing operation data and semiconductor quality data, to determine the quality of semiconductors produced in the industry [18]. The dataset comprises 1567 instances with two classes, 104 fails, and some missing values. The SECOM dataset is composed of 1567 examples that originate from a wafer manufacturing production line [20]. Each example in the dataset is represented by a vector of 590 sensor measurements and includes an identification for pass-fail testing. Out of the total examples, only 104 are marked positive as failed cases, coded as 1, while the majority of examples pass the test and are marked negative, coded as -1 [21]. This significant class imbalance poses a challenge in achieving a good balance between the precision and recall of the classifier. The SECOM Dataset poses a two-class problem, but there is an imbalance in the distribution with a 14:1 skew of the pass to fail. With a total of 590 features, this dataset presents a significant number of variables [22]. The dataset contains prevalent missing and incomplete feature information, along with constant values in some columns, making it unique. It also includes a date-time stamp and varying intensity of null values corresponding to unrecorded measurements [23].

4. PROPOSED METHODOLOGY

This research paper looks at some of the most prominent machine learning and deep learning algorithms for sensor-based manufacturing approaches [15]. The proposed methodology is outlined, and the corresponding implementations are summarized in Figure 1. The dataset we worked with comprises 1567 records and 592 properties. The msno function was used to check for missing values once we discovered them when cleaning the data [16]. A graphical depiction of missingness patterns in a dataset, known as a matrix plot for missing values, enables you to quickly and accurately identify trends in missingness across several variables. The primary advantage of using a matrix plot for missing values is that it can aid researchers in developing better-informed decisions regarding handling missing data, resulting in more reliable and accurate data analyses. Missing values were found in the dataset, affecting conclusions. Pass/fail predictions were displayed to identify imbalances, and the dataset was balanced using the smote library, with the results shown [17]. As we know, there are two methods for balancing a dataset: oversampling and undersampling. We initially used the undersampling technique, specifically the random undersampling, to balance our data by reducing the number of case conditions. However, we found that this approach was not particularly helpful since fewer case conditions result in lower prediction accuracy. Therefore, we decided to try out oversampling techniques and tested various options such as

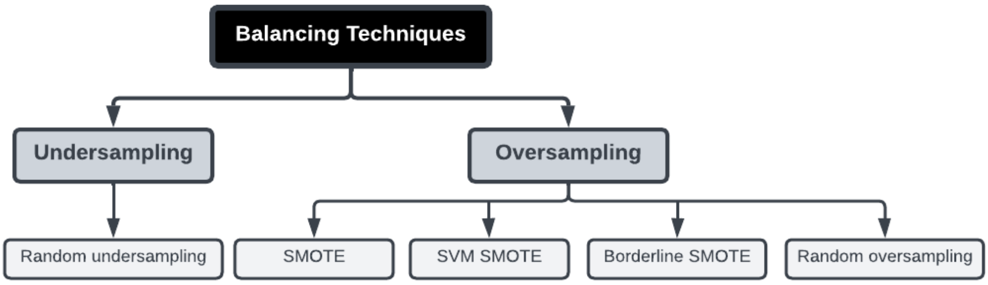

Figure 2. Different Types of Balancing Techniques.

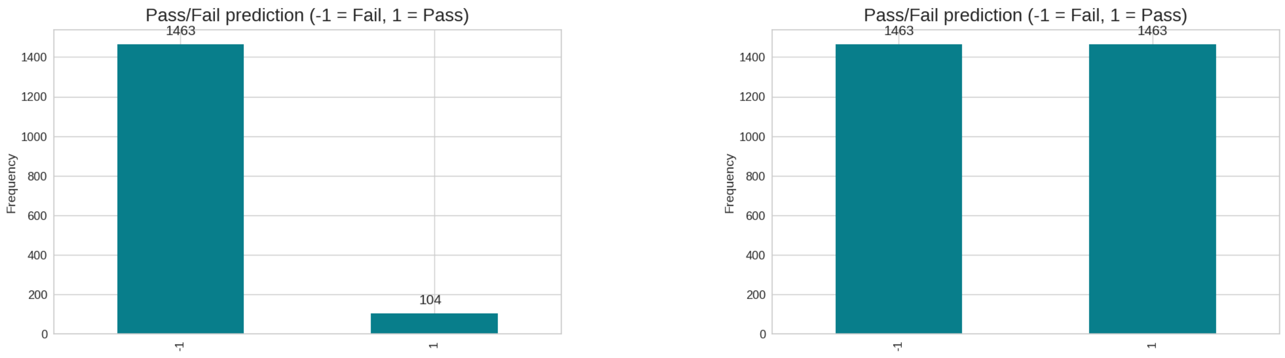

Figure 3. Before and After employing Balancing Techniques.

To prepare the data for analysis or modeling, we performed some preprocessing steps on the dataset. One of these processes was to divide the ”timestamp” column, which enabled us to turn the timestamp values into a numerical format that could be expressed as a decimal integer. Moreover, we discovered and deleted some unnecessary columns, such as ”start time.” Using a unique function, we were able to retrieve a special weekday, year, month, and date. After importing the dataset into a pandas data frame and analyzing its rows and columns, significant imbalances in the target values were found. To resolve this issue, the target values were balanced using the SMOTE method. The data was then split into 75:25 training and testing sets using the train-test split approach. We used a dataset containing a variety of characteristics, the bulk of which were numerical. The information in a few columns, such as ”year”, ”month”, ”date”, ”weekday”, ”hour”, and ”min,” however, was not numerical. We carefully considered these features and determined that they were not necessary for our modeling or analysis, therefore we removed them from the dataset. By implementing the Synthetic Minority Over-sampling Technique (SMOTE), multiple data points are added to the dataset, which leads to a significant enhancement in the efficiency and accuracy of the model. This is primarily attributed to the fact that the additional data points aid in improving the feature engineering process. It has been observed that incorporating SMOTE has resulted in a reduction in training time compared to not utilizing SMOTE. In our dataset, we encountered the challenge of missing values, which can hinder data analysis and modeling. To overcome we have implemented the KNN imputer.

- **KNN Imputer:**
  We handled missing values in our dataset by using the K-nearest neighbor (KNN) imputation method. KNN imputation utilizes the values of the nearest neighboring data points to fill in missing values, but it was subsequently removed after imputation. Initially, we removed columns with more than 30% missing values. For the remaining missing samples (less than 30%), KNN imputation was applied. To prevent data leakage, the imputed data was modified only in the testing dataset. This approach led to a reduction in training time [24].

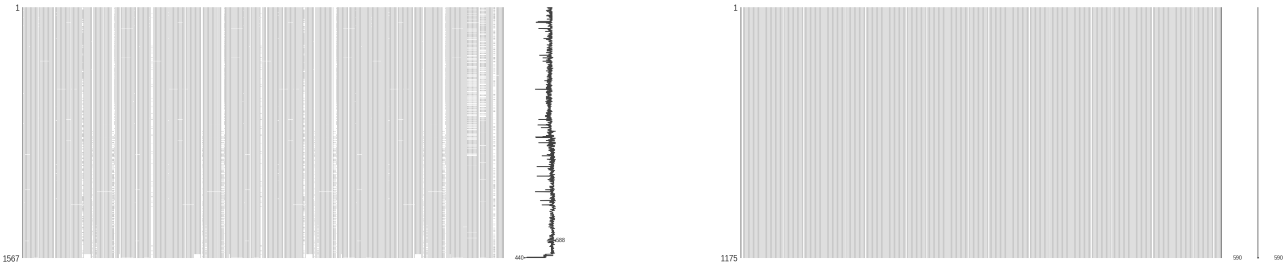

Figure 4. Caption

5. FEATURE ELIMINATION

Our main aim was to significantly shorten the training period, using several techniques described below. As part of our plan, the number of characteristics dropped, which significantly reduced training time [26].

- **Normalization:**
  Normalization was applied as a preprocessing step before implementing the variance threshold. The purpose of normalization is to adjust the scale of numerical data, reducing its sensitivity to different feature scales. It rescales the data to fit within the range of 0 to 1. Normalization plays a crucial role in enhancing the performance of machine learning models. Standardizing the feature scale ensures that no single feature dominates the learning process based solely on its magnitude. This helps prevent biased learning and improves the overall efficiency and accuracy of the models. This is particularly important when using distance-based algorithms, such as KNN or clustering algorithms, where the calculation of distances is highly influenced by the scale of features [27].
- **Variance Threshold:**
  We noticed repeated values in corresponding rows, which hindered progress. To address this, we used the variance check method with a specific threshold (e.g., zero) to eliminate low-variance features. We imported a normalizer from sklearn and applied it to both the training and testing data to prevent data leakage. Features with a zero variance, indicating constant values that remained the same in each row, were removed. After this process, we were left with 590 features. We then utilized KNN imputation and reevaluated training time and outcomes.
- **Correlation:**
  To evaluate the correlations between features, we examined correlation coefficients. Weakly correlated features (those close to 0) were deemed unimportant, whilst strongly correlated features (those near 1 or -1) were eliminated as redundant. We will exclude independent characteristics with a high absolute correlation since they are identical to other features and do not enhance our model. This left 264 features remaining after 214 features were removed, which then caused 214 further features to be removed. After this procedure, 264 important qualities were left [28].

- **Correlation With target:**
  Following the previous steps, we assessed the correlation between the remaining features and the target column. Features with a correlation below 5% were dropped. As a result, we identified 40 correlated features, while removing 224 features from the dataset.
- **Recursive Feature Elimination:**
  The RFE method recursively removes features until the desired number is achieved for feature selection. It starts by building a model with all components, then removes less crucial features and evaluates performance. RFE is efficient for high-dimensional datasets with few observations. We used RFE to obtain a reduced feature set. Applying KNN imputation significantly reduced training time compared to the ”correlation with target” approach [29].

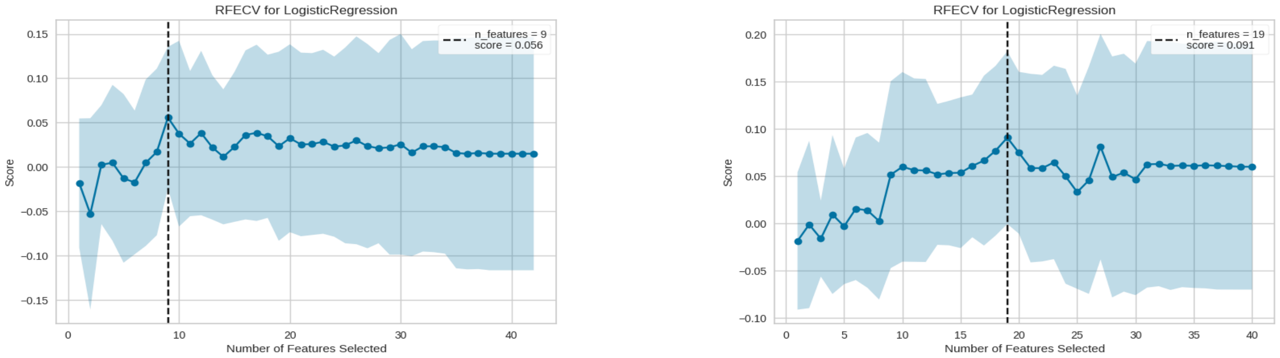

Figure 5. Caption

The graph visually represents the recursive feature selection process and identifies the optimal number of features for the logistic regression model. Initially, the analysis concentrated on examining 9 attributes prior to incorporating SMOTE. However, after the integration of SMOTE, the selection process identified a total of 19 features. Upon implementing the undersampling technique, the feature count remained unchanged compared to the scenario without undersampling, attributed to the reduced case conditions. Likewise, upon employing various oversampling techniques and enhancing the case conditions, the feature count remained constant across all oversampling techniques. A visual representation is provided above, outlining the associated training time at each stage of the feature engineering procedure. Despite the reduction in training time, the model's MCC Score and F1 Score have remained unchanged.

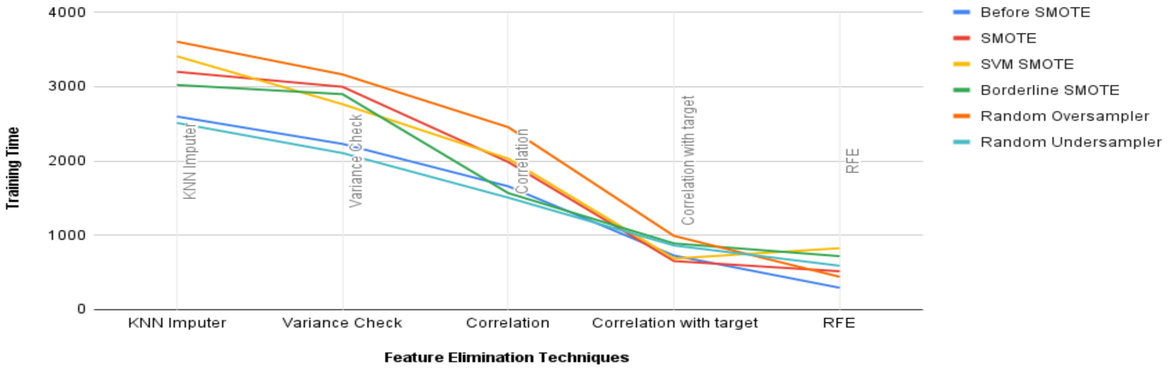

Figure 6. Training Time with respect to different SMOTEs and Feature Elimination Techniques

6. IN-DEPTH EXAMINATION OF RESULTS

The main aim of the paper is to examine and forecast the manufacturing process of semiconductor devices using several machine learning and deep learning techniques and compare the results to select the best-performing model. After the implementation of the machine learning and deep learning techniques to the SECOM dataset, we presented the results in a tabular and graphical format which makes it easy to understand in a concise manner [31]. The graph provided above illustrates the evaluation of multiple classifiers, both before and after employing the SMOTE. Initially, only one classifier demonstrated a significant improvement in performance compared to the others. Specifically, the Random Forest achieved the highest accuracy rate of 93%.

Table 1. Accuracy of different classifications for different SMOTE techniques

Accuracy	**Under sampler**	**SMOTE**	**SVM SMOTE**	**Random Oversampler**	**Borderline SMOTE**	**Before SMOTE**
Logistic Regression	63.27%	68.11%	77.81%	78.88%	69.39%	69.17%
Decision Tree	62.55%	80.61%	84.18%	88.01%	83.40%	86.99%
XGBoost	69.44%	89.54%	90.05%	92.35%	88.27%	89.86%
Random Forest	81.05%	88.01%	94.60%	96.37%	91.80%	92.88%
SVM	64.29%	46.68%	55.36%	65.41%	49.23%	48.60%
Neural Networks	71.78%	70.15%	84.73%	87.48%	66.58%	80.36%

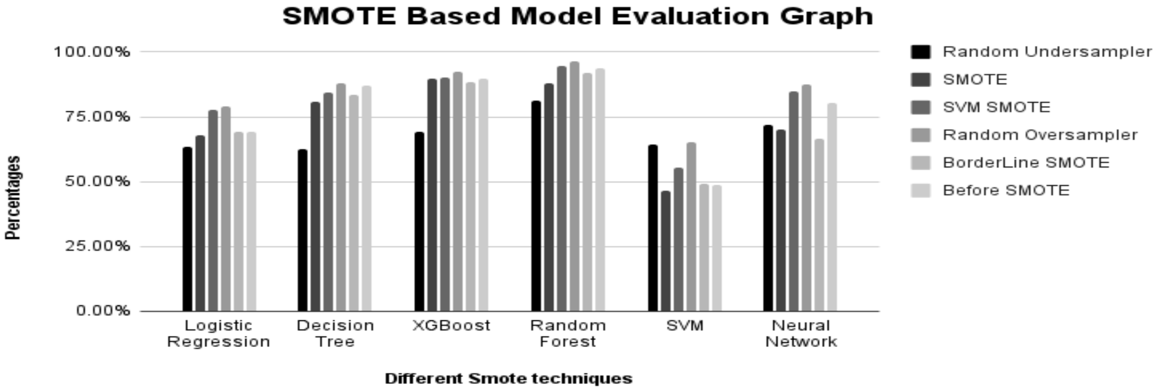

Figure 7. Result Analysis of Performances.

However, the introduction of SMOTE led to a substantial improvement in the number of well-performing classifiers. Among the various SMOTE methods tested, random oversampling yielded the best results. Additionally, machine learning surpassed deep learning in terms of accuracy. Notably, the Random Forest exhibited superior performance, achieving accuracy rates of 93.37%, outperforming the other classifiers.

Table 2. SMOTE Vs Models

SMOTE Vs Models	**Best Models**	**Accuracy**
Undersampler	Random Forest	81.05%
SMOTE	XGBoost	89.54%
SVM SMOTE	Random Forest	94.60%
Random Oversampler	Random Forest	96.37%
Borderline SMOTE	Random Forest	91.80%
Before SMOTE	Random Forest	92.88%

Implementing SMOTE techniques showed a slight improvement in accuracy, though not significantly impacting neural network performance. The comparison highlights the best-performing machine learning technique against deep learning, emphasizing the potential of SMOTE to enhance model performance [30].

Table 3. Best scores in Machine Learning and Deep Learning

ML Vs DL	Random Forest	Neural Network
Accuracy	96.37%	87.48%
Recall	100.00%	94.86%
Precision	94.31%	92.71%
F1 Score	93.59%	90.82%

7. CONCLUSION

This study investigates how the combination of different SMOTE techniques and machine learning methods can effectively detect defects in manufacturing processes. The research findings highlight that employing various SMOTE methods significantly enhances the accuracy, precision, recall, and f-score [25] of the classifiers by creating superior features. Prior to implementing SMOTE, only the Random Forest demonstrated satisfactory results, even after applying SMOTE techniques exhibited the most promising performance in predicting manufacturing defects. These findings emphasize the crucial role of SMOTE and feature selection methodologies in improving the effectiveness of machine learning algorithms for detecting faults in manufacturing processes, leading to greater efficiency and accuracy. The incorporation of feature selection approaches successfully decreased the number of features, which resulted in a successful reduction in training time. By integrating various SMOTE techniques and feature selection methods, it becomes possible to develop more precise and efficient monitoring systems for manufacturing processes, thereby reducing defects and enhancing product quality. Additionally, the study suggests exploring alternative feature selection techniques to further enhance the algorithm's performance in detecting faults during manufacturing, offering new avenues for future research.

REFERENCES

[1] Dogan, A., Birant, D. (2021). Machine learning and data mining in manufacturing. Expert Systems with Applications, 166, 114060.

[2] Stuart, T. E. (2000). Interorganizational alliances and the performance of firms: a study of growth and innovation rates in a high-technology industry. Strategic management journal, 21(8), 791-811.

[3] G. Busch, "Early history of the physics and chemistry of semiconductors – from doubts to fact in a hundred years", Eur. J. Phys., vol. 10, no. 4, pp. 254–263, 1989.

[4] F. Laeri, F. Schüth, U. Simon, and M. Wark, Host-Guest-Systems Based on Nanoporous Crystals. Weinheim: Wiley, 2003,pp. 435–436.

[5] H. Bauer, P. Ranade, and S. Tandon, "Big data and the opportunities it creates for semiconductor players," McKinsey on Semiconductors, pp. 46–55, 2012.

[6] J. Dietz and C. Knepfler, "New controller extends the lifetime of 200mm tools," Nanochip Fab Solutions, vol. 8, no. 2, pp. 14–17, 2013.

[7] Kim, J., Han, Y., / Lee, J. (2016). Data imbalance problem solving for smote-based oversampling: Study on fault detection prediction model in the semiconductor manufacturing process. Advanced Science and Technology Letters, 133, 79-84.

[8] Salem, M., Taheri, S., Yuan, J. S. (2018). An experimental evaluation of fault diagnosis from imbalanced and incomplete data for smart semiconductor manufacturing. Big Data and Cognitive Computing, 2(4), 30.

[9] Kerdprasop, K., Kerdprasop, N. (2010, March). Feature selection and boosting techniques to improve fault detection accuracy in the semiconductor manufacturing process. In World Congress on Engineering 2012. July 4-6, 2012. London, UK. (Vol. 2188, pp. 398-403). International Association of Engineers.

[10] Moldovan, D., Chifu, V., Pop, C., Cioara, T., Anghel, I., Salomie, I. (2018, September). Chicken swarm optimization and deep learning for manufacturing processes. In 2018 17th RoEduNet conference: networking in education and research (RoEduNet) (pp. 1-6). IEEE.

[11] Munirathinam, S., Ramadoss, B. (2016). Predictive models for equipment fault detection in the semiconductor manufacturing process. IACSIT International Journal of Engineering and Technology, 8(4), 273-285.

[12] Karthigaikumar, P. (2021). Industrial quality prediction system through data mining algorithm. Journal of Electronics and Informatics, 3(2), 126-137.

[13] Doke, O. (2020). Data Mining for Enhancing Silicon Wafer Fabrication (Doctoral dissertation, Dublin, National College of Ireland).

[14] Cioara, T., Anghel, I., Moldovan, D., Tomus, M. M., Salomie, I. Prediction of Manufacturing Processes Errors: Gradient Boosted Trees Versus Deep Neural Networks.

[15] Gondalia, A., Dixit, D., Parashar, S., Raghava, V., Sengupta, A. and Sarobin, V.R., 2018. IoT-based healthcare monitoring system for war soldiers using machine learning. Procedia computer science, 133, pp.1005-1013.

[16] O. Sagi and L. Rokach, "Approximating xgboost with an interpretable decision tree," Information Sciences, vol. 572, pp. 522–542, 2021.

[17] C. Cortes and V. Vapnik, "Support-vector networks," Machine learning, vol. 20, no. 3, pp. 273–297, 1995.

[18] Godina, R., Rodrigues, E. M. (2021). A Review of Data Mining Applications in Semiconductor Manufacturing. Processes, 9(2), 305.

[19] McCann,Michael and Johnston,Adrian. (2008). SECOM. UCI Machine Learning Repository. https://doi.org/10.24432/C54305.

[20] Y. Takahashi, M. Asahara and K. Shudo, "A Framework for Model Search Across Multiple Machine Learning Implementations," 2019 15th International Conference on eScience (eScience), San Diego, CA, USA, 2019, pp. 331-338, doi: 10.1109/eScience.2019.00044.

[21] N. Makishima et al., "Independent Deeply Learned Matrix Analysis for Determined Audio Source Separation," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 27, no. 10, pp. 1601-1615, Oct. 2019, doi: 10.1109/TASLP.2019.2925450.

[22] Wanner, J., Herm, L. V., Heinrich, K., Janiesch, C. (2022). A social evaluation of the perceived goodness of explainability in machine learning.

Journal of Business Analytics, 5(1), 29-50.

[23] I. Anghel, T. Cioara, D. Moldovan, I. Salomie and M. M. Tomus, ”Prediction of Manufacturing Processes Errors: Gradient Boosted Trees Versus Deep Neural Networks,” 2018 IEEE 16th International Conference on Embedded and Ubiquitous Computing (EUC), Bucharest, Romania, 2018, pp. 29-36, doi: 10.1109/EUC.2018.00012.

[24] Harris, P. A., Taylor, R., Thielke, R., Payne, J., Gonzalez, N., Conde, J. G. (2009). Research electronic data capture (REDCap)—a metadatadriven methodology and workflow process for providing translational research informatics support. Journal of biomedical informatics, 42(2), 377-381.

[25] Shung, K. P. (2021). Accuracy, Precision, Recall, or F1? 2018. URL: https://towardsdatascience. com/accuracy-precision-recall-or-f1-331fb37c5cb9 (3.7. 2021.).

[26] Kim, J., Han, Y., Lee, J. (2016). Data imbalance problem solving for smote-based oversampling: Study on fault detection prediction model in the semiconductor manufacturing process. Advanced Science and Technology Letters, 133, 79-84.

[27] Wright, R. E. (1994). Logistic regression. In L. G. Grimm / P. R. Yarnold (Eds.), Reading and understanding multivariate statistics (pp. 217-244). Washington, DC: American Psychological Association.

[28] Jin, Z., Lim, D.D., Zhao, X. et al. Machine learning enabled the optimization of showerhead design for the semiconductor deposition process. J Intell Manuf (2023). https://doi.org/10.1007/s10845-023-02082-8.

[29] Sreejith, S., Nehemiah, H. K., Kannan, A. (2020). Clinical data classification using an enhanced SMOTE and chaotic evolutionary feature selection. Computers in Biology and Medicine, 126, 103991.

[30] Mahesh, Batta. (2019). Machine Learning Algorithms -A Review. 10.21275/ART20203995.

[31] Meyer, D. Support Vector Machines * the Interface to Libsvm in Package E1071 Basic Concept.

BIOGRAPHIES OF AUTHORS

Sai Venkat Boyapati holds a Bachelor’s degree in Computer Science and Engineering from SRM University, Andhra Pradesh, India. He further pursued his M.S. degree in Computer and Information Sciences at Texas Tech University, Texas, USA. During his academic journey, he made significant contributions by publishing research papers titled ”An Analysis of House Price Prediction Using Ensemble Learning Algorithms” in the Research Reports on Computer Science Journal and ”A Comparative Analysis of the Evolution of DNA Sequencing Techniques along with the Accuracy Prediction of a Sample DNA Sequence Dataset using Machine Learning” for IEEE Xplore. Additionally, he has gained practical experience as a Team Lead at the Entrepreneurship cell, where he demonstrated his leadership skills. Moreover, he successfully developed the official websites for the Coding Club and Hackathon Club of SRM University AP, showcasing his proficiency in web development. To enhance his skills, he undertook a two-month internship in the Machine Learning domain, utilizing Python, facilitated by the Andhra Pradesh State Skill Development Corporation. For further communication, he can be reached at the email address angela@uum.edu.my.

Bhanu Sri Rakshitha Guduru acquired the Bachelor’s degree in Computer Science and Engineering from SRM University in Andhra Pradesh, India. Building upon her academic achievements, She pursued an M.S. degree in Computer and Information Sciences from the University of Cincinnati in Ohio, USA. To further enhance her skills, she participated in a 2-month internship in the Data Analysis and Machine Learning domain, utilizing Python. This internship was conducted in collaboration with the Andhra Pradesh State Skill Development Corporation during her undergraduate studies. Throughout her academic journey, she actively contributed as a Team Member in the Entrepreneurship cell, showcasing her dedication and expertise. Additionally, she demonstrated her proficiency by developing official websites for the Coding Club at SRM University AP. She can be contacted at email: rakshithagudru@gmail.com.

Rishitha Reddy Maryada earned a Bachelor’s degree in Computer Science and Engineering from SRM University in Andhra Pradesh, India. She made significant contributions as a Team Leader in the Entrepreneurship cell, consistently demonstrating her dedication and expertise. She actively engaged in a 2-month internship in the Data Analysis and Machine Learning domain, utilizing Python. This valuable experience was conducted in collaboration with the Andhra Pradesh State Skill Development Corporation during their undergraduate studies. Notably, she worked on a project involving medical review analysis using sentiment analysis techniques. She can be contacted by email:

Sumalatha Saleti currently serves as an Assistant Professor in the Computer Science and Engineering department at SRM University located in Guntur, Andhra Pradesh, India. She has earned her Ph.D. degree from the esteemed National Institute of Technology in Warangal, India. Sumalatha Saleti’s research focus encompasses areas such as data mining, big data analytics, and machine learning. For any inquiries or correspondence, she can be contacted via email at sumalatha.s@srmap.edu.in.