Machine Learning Models for Analysing and Predicting Team Productivity in Garment Manufacturing

Zareena Begum[1*], Arukala Anjan[2], Guda Abhivardhan[2], Murahari Varun Tej[2], Eera Hashwanthratna[2]

[1]Associate Professor, [2]UG Student, [1,2]Department of Computer Science and Engineering (Data Science), Vaagdevi College of Engineering, Bollikunta, Warangal, Telangana.

*Corresponding Email: zareena@vaagdevi.edu.in

ABSTRACT

The garment manufacturing industry contributes to over 6% of global industrial employment and generates $1.5 trillion in annual revenue. However, inefficiencies in workforce productivity lead to a 20–30% loss in overall production efficiency, affecting profit margins and delivery timelines. Traditionally, team productivity is monitored manually, relying on supervisor assessments and historical averages, which are prone to inconsistencies, delays, and human bias. These manual tracking systems fail to adapt to real-time production changes, resulting in inaccurate efficiency evaluations. To address these challenges, we propose a machine learning-driven productivity prediction framework specifically designed for garment manufacturing, using the Garment dataset with "actual productivity" as the target variable. The methodology includes preprocessing techniques such as intelligent feature mapping and adaptive imputation to handle missing data and categorical variables. A comprehensive Exploratory Data Analysis (EDA) is conducted using scatter plots, KDE plots, histograms, correlation heatmaps, regression plots, violin plots, line plots, density plots, and bar plots, offering deep insights into team productivity trends. The dataset is processed through train-test splitting to ensure balanced data distribution for model training. In the predictive modeling phase, a Decision Tree Regressor serves as a baseline to capture primary productivity trends. To enhance accuracy, a CatBoost Regressor is introduced, leveraging gradient boosting techniques to capture complex non-linear relationships within the dataset. The integration of advanced preprocessing, EDA, and hybrid regression modeling significantly improves forecasting accuracy over traditional manual methods. The proposed system enables real-time productivity monitoring, minimizes inefficiencies, and enhances decision-making in garment manufacturing.

Keywords: Team Productivity, Garment Manufacturing, Machine Learning, Prediction Models, Workforce Analysis.

1. INTRODUCTION

Garment manufacturing has long constituted a cornerstone of the global textile industry, driven economic growth and providing employment for millions, particularly in developing economies such as Bangladesh, India, and Vietnam. Historically, production of garments transitioned from small-scale artisanal workshops in the late nineteenth and early twentieth centuries to mass-production assembly lines introduced by pioneers such as Isaac Singer and later Henry Ford's adaptations, which emphasized standardized tasks and mechanization. During and after World War II, technological innovations in sewing machinery, synthetic fibers, and production-planning techniques further accelerated output, giving rise to modern factory systems characterized by specialization, division of labor, and the proliferation of large-scale apparel conglomerates.

By the late 20th century, the industry had embraced Total Quality Management, lean manufacturing, and just-in-time inventory strategies to reduce lead times and minimize waste. More recently, the advent of Industry 4.0 technologies—including Internet of Things (IoT) sensors, digital twins, and real-time data analytics—has offered unprecedented opportunities to monitor machine utilization, task cycle times,

doi: 10.48047/ijiee.2025.15.4.19

and environmental factors, yet the translation of these data streams into actionable insights for workforce productivity remains nascent. Despite significant process automation, the human element continues to dominate shop-floor operations, with team productivity subject to variability arising from skill-level heterogeneity, ergonomic constraints, supply-chain volatility, and fluctuating demand patterns.

Traditional productivity assessment methods, often reliant on manual time-motion studies and basic output-per-hour metrics, fail to capture complex interactions among operators, machinery, and work processes, while lacking predictive capabilities needed for proactive management. Motivated by the imperative to optimize labor utilization and enhance factory agility, this research project explores the application of machine learning (ML) models to analyze multi-dimensional production data—ranging from individual operator motion tracking, task completion times, and equipment-downtime logs—and to predict team performance under diverse operational scenarios. By leveraging supervised learning techniques, the study aims to uncover latent patterns and non-linear relationships that traditional statistical approaches may overlook.

The primary objectives are to (1) develop and validate ML models capable of accurate and interpretable forecasting of team productivity metrics; (2) perform a comparative evaluation of different algorithmic approaches in terms of predictive accuracy, robustness, and computational efficiency. In addressing these objectives, the research seeks to bridge the gap between advanced analytics and practical shop-floor management, thereby contributing to the evolution of resilient, adaptive, and high-performing garment manufacturing systems. It empowers data-driven decision-making for continuous organizational improvement.

2. LITERATURE SURVEY

Bhatia, Arora, and Tomar 2016 [1] for presence of diabetic retinopathy, the results proved that the model could help in detecting symptoms earlier. Outperformed results were found in a study conducted. Kruppa et al. 2013 [2] for credit risk prediction using framework of machine learning algorithms such as random forests (RF), k-nearest neighbors (KNN) and bagged K-nearest neighbors (BKN). Furthermore, a study by Balla, Rahayu, and Purnama 2021 [3] proved a promising result in predicting employee's productivity which is one of the most substantial factors in any organization. The study applied three classification algorithms namely, Neural Network (NN), Random Forest (RF) and Regressi Linier (RL).

Random forest showed minimal values of correlation coefficient, MAE, and RMSE, which reflect that RF is very appropriate in predicting employee's productivity. Decision tree classification algorithms utilized by Attygalle and Abhayawardana 2021 [4] for investigating and visualizing employee productivity and any other social phenomenon with evidence. Moreover, decision tree methods and data mining tools employed by Ďurica, Frnda, and Svabova 2019 [5] to build a model for predicting financial difficulties of polish companies. The results presented prediction power around 98% and more. In addition, Mahoto et al. 2021 [6] had used three machine learning algorithms (Multiclass Random Forest, Multiclass Logistic Regression, Multiclass one-vs-all) in order to help business workers to set product pricing and discounts depending on customer behavior, the model showed outstanding results in product price prediction.

On the other hand, prediction model has been built by study Sorostinean, Gellert, and Pirvu 2021 [7] using decision tree methods and data mining tools for investigating the effect of decision tree methods and ensemble learning for improving performance prediction in assembly assistance system. The results demonstrated that the gradient boosted decision trees was the best through all the decision treebased methods. Some studies evaluated worker 's performance of textile company by using ML and ensemble

learning algorithm, such as study asSaad. [8] which applied different Machine learning algorithms including, decision tree and bagging algorithm to achieve the highest accuracy. The CHAID model produced high-level specificity and sensitivity. Four different ML algorithms including, support vector machine, optimized support vector machine (using genetic algorithm), random forest, XGBoost and Deep Learning were used. El Hassani, El Mazgualdi, and Masrour [9] for predicting the overall equipment effectiveness (OEE) which is a performance measurement of manufacturing industry. Deep learning and random forest with cross validation manifest the best results for predicting OEE. Additionally, an approach built in study De Lucia,

Pazienza, and Bartlett [10] of ML and logistic regression used for financial performance prediction by focusing on predicting the accuracy of main financial indicators such as Return of Equity (ROE) and Return of Assets (ROA). The ML algorithms were performed perfectly for predicting ROE and ROA. All studies and research work mentioned above focused on combining two or more classifiers and how this integration of different techniques and algorithms can help in prediction. This research focuses on combining classification algorithms with bagging and Adaboost. In addition, the iterations from 1 to 100 are recorded to study how these combinations influence the accuracy, RMSE, and MAE values of predicting employees' productivity [11][12].

3. PROPOSED SYSTEM

The method is not presented in existing surveys and overcomes the drawbacks of traditional manual monitoring by integrating preprocessing mapping, exploratory data analysis (EDA), a two-stage regression modeling approach using Decision Tree Regressor and CatBoost Regressor, and performance evaluation for garment manufacturing productivity prediction. The preprocessing step includes intelligent feature mapping, adaptive mean clustering for missing value imputation, and isolation forest-based outlier detection. EDA is performed using scatter plots, KDE plots, histograms, correlation heatmaps, regression plots, violin plots, line plots, density plots, and bar plots to extract valuable productivity insights. The predictive model consists of two phases: an initial Decision Tree Regressor to establish baseline trends and a fine-tuned CatBoost Regressor to capture complex non-linear relationships, improving accuracy. The hybrid combination ensures real-time adaptability, high precision, and superior performance over conventional manual tracking and single-model regression approaches.

Step-1: Data Collection and Preprocessing

The methodology begins with the collection of real-time productivity data from garment manufacturing teams, including worker efficiency, machine downtime, defect rates, shift hours, and production complexity. Preprocessing mapping is applied to clean and standardize the dataset, ensuring structured input for analysis. Missing values are handled through adaptive mean clustering, dynamically selecting the best imputation strategy. Additionally, outliers are detected and removed using an isolation forest algorithm, preventing extreme values from distorting predictions.

Step-2: Exploratory Data Analysis (EDA) and Visualization

A variety of data visualization techniques are employed to gain comprehensive insights into productivity patterns within the garment manufacturing dataset. Scatter plots are used to analyze relationships between key variables, while KDE plots and histograms help study the distribution of productivity across different data points. Correlation heatmaps reveal dependencies among features, enabling the identification of influential factors. To understand team-level performance, regression plots and violin plots are utilized to compare productivity variations across teams. Line plots are applied to track productivity trends over time, highlighting temporal patterns. Additionally, density plots and bar

plots are used to categorize and visualize efficiency levels, offering a clearer view of performance clusters within the workforce.

Step-3: Train-Test Splitting and Baseline Decision Tree Regressor

The dataset is split into training and testing subsets using stratified sampling. The Decision Tree Regressor serves as the baseline model, capturing primary productivity trends. Hyperparameter tuning, such as max depth optimization and entropy-based splitting, is applied to prevent overfitting. This phase provides an initial understanding of productivity drivers.

Step-4: Advanced Prediction with CatBoost Regressor

To enhance accuracy, a CatBoost Regressor is implemented, leveraging its efficient handling of categorical variables without requiring extensive encoding. The model is trained using gradient boosting, learning from historical productivity data. Bayesian optimization is used for hyperparameter tuning, improving generalization. This step ensures the model captures complex productivity patterns with high precision.

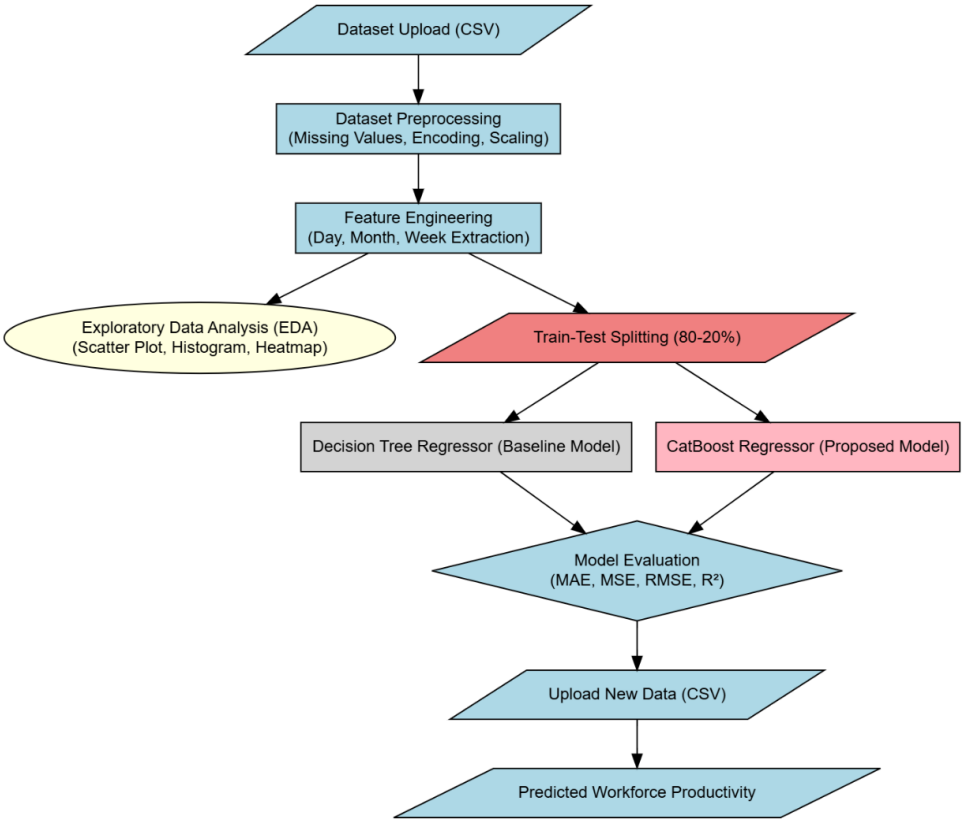

Fig. 2: Proposed system architecture of predicting team productivity in garment industry.

Step-5: Performance Evaluation and Real-Time Implementation

The model's accuracy is assessed using Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R^2 score. The hybrid Decision Tree–CatBoost Regressor model achieves superior results compared to manual monitoring and single-model regressors. The final model is deployed in a real-time monitoring system, enabling garment manufacturers to make data-driven decisions, optimize team productivity, reduce downtime, and improve operational efficiency.

doi: 10.48047/ijiee.2025.15.4.19

3.2 ML Model Building

3.2.1 Decision Tree Regressor

A Decision Tree Regressor (DTR) is a supervised machine learning algorithm used for regression tasks. It works by recursively splitting the dataset into smaller subsets based on feature values, forming a tree-like structure as shown in Fig. 4.2. The goal is to create a model that predicts numerical outcomes (such as actual productivity in garment manufacturing) by learning from training data and making rule-based decisions. The function implements DTR using x_train and y_train for training and x_test for testing, with predictions compared against y_test for evaluation.

Step 1: Model Initialization

The DecisionTreeRegressor() is initialized without additional parameters. This means the model will create a decision tree that attempts to fit the data without predefined constraints, allowing it to fully grow based on the training dataset. The tree structure will be optimized based on the feature values in x_train and their relationship with y_train (actual productivity).

Step 2: Model Training with x_train and y_train

The fit() function is used to train the model, where x_train (input features) and y_train (actual productivity values) are provided. The Decision Tree Regressor analyzes the relationships in this data by splitting it at decision nodes, finding patterns in how different features impact productivity. The tree continues to grow until an optimal structure is formed, minimizing the error in predicting actual productivity.

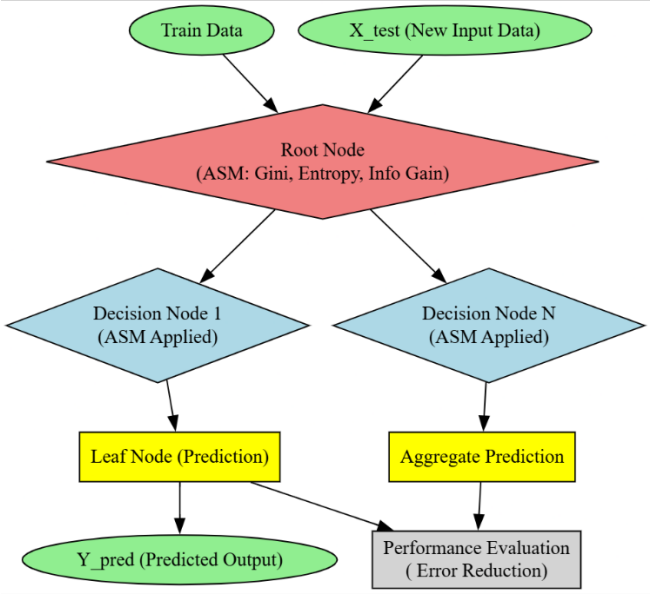

Fig.3: Block Diagram of Random Forest Regressor.

Step 3: Making Predictions with x_test

After training, the model is applied to x_test, a set of unseen data points. The predict() function is used to estimate the productivity values based on the learned tree structure. The decision tree follows the trained rules, evaluating feature values in x_test and determining predicted y_test values accordingly.

Step 4: Evaluating Performance

The predicted values from predict are compared with the actual values in y_test using the calculateMetrics() function. This evaluation helps determine how well the Decision Tree Regressor is

doi: 10.48047/ijiee.2025.15.4.19

performing. Metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), or R^2 score might be used to measure prediction accuracy.

3.2.2 CatBoost Regressor

CatBoost Regressor (CBR) is an advanced gradient boosting algorithm designed to handle categorical data efficiently while improving prediction accuracy and training speed. It is particularly useful for predicting team productivity in garment manufacturing, as it can model complex relationships in the data while reducing overfitting. In the function, CBR is implemented with a mechanism to load a pre-trained model or train a new model if none exists. It takes x_train and y_train as inputs for training and x_test as test input, with predictions compared against y_test for evaluation.

Step 1: Checking for an Existing Model

Before training a new model, the function first checks if a previously trained CatBoost Regressor model exists in the directory "model/CBR.pkl". If the file is found, the model is loaded using joblib, allowing the function to skip the training step and directly proceed with making predictions. This ensures efficiency by avoiding unnecessary re-training, which saves computational time.

Step 2: Loading or Training the Model with x_train and y_train

If a pre-trained model does not exist, a new CatBoost Regressor is initialized and trained using x_train (input features) and y_train (actual productivity values). The model applies gradient boosting techniques to iteratively improve predictions by minimizing errors. It uses an ensemble of decision trees, where each tree corrects errors made by the previous trees, resulting in a highly optimized predictive model.

Step 3: Saving the Trained Model

Once training is complete, the trained model is saved as "CBR.pkl" using joblib. This ensures that the model can be reused in future runs without needing to be retrained, making the system more efficient and scalable.

Step 4: Making Predictions with x_test

The trained CatBoost Regressor is then used to predict actual productivity values for the unseen test dataset x_test. Since CatBoost handles categorical features effectively, it ensures that productivity trends are captured more accurately compared to traditional regression models.

Step 5: Loss Optimization and Performance Evaluation

After making predictions, an additional step called loss optimization is applied using the loss_optimization() function. This step fine-tunes the predicted values to further reduce errors. Finally, the calculateMetrics() function evaluates the model's performance by comparing the optimized predictions against y_test (actual productivity values).

4. RESULTS AND DISCUSSION

The research is based on machine learning application which predicts workforce productivity in the garment industry using multiple regression models, primarily Decision Tree Regressor (DTR) and CatBoost Regressor (CBR). The implementation integrates data preprocessing, exploratory data analysis (EDA), model training, evaluation, and prediction using a Tkinter GUI.

4.1 Dataset Description

The dataset represents workforce productivity in a garment manufacturing industry. It contains various factors influencing productivity, such as team structure, work schedules, incentives, and operational

efficiency. Each row in the dataset represents a specific day's performance for a given team within a department.

The dataset used for productivity prediction in garment manufacturing includes a variety of features capturing both temporal and operational aspects of production. The date column records the specific day the data was collected, while quarter indicates the fiscal quarter of the year, ranging from 1 to 4 (or 5 in some cases). The department refers to the division within the factory, and day represents the numerical day of the week. The team column identifies the specific team working within a department. targeted_productivity denotes the planned or expected productivity level for that day. smv (Standard Minute Value) indicates the time allocated to complete a task, and wip (Work in Progress) reflects the number of unfinished garments. Operational metrics such as over_time, which captures extra work hours, and incentive, representing performance-based monetary rewards, are also included. Additionally, the dataset features idle_time and idle_men to measure periods and the number of workers not actively engaged in tasks. It also tracks no_of_style_change, showing how often garment styles were switched, and no_of_workers, indicating the total workforce involved. Finally, the actual_productivity column records the true productivity level achieved, serving as the target variable for predictive modeling.

4.2 Result Analysis

Fig.4: represents the performance analysis of the Naïve Bayes regression model, showing a comparison between the true and predicted values of workforce productivity. The scatter plot in the GUI visualizes how well the model's predictions align with the actual values. Ideally, points should closely follow a diagonal reference line, indicating high accuracy. However, since Naïve Bayes is primarily designed for classification tasks rather than regression, its predictions may show significant deviations from actual productivity values. The GUI displays key evaluation metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and R-squared (R^2) score, helping users assess the model's effectiveness.

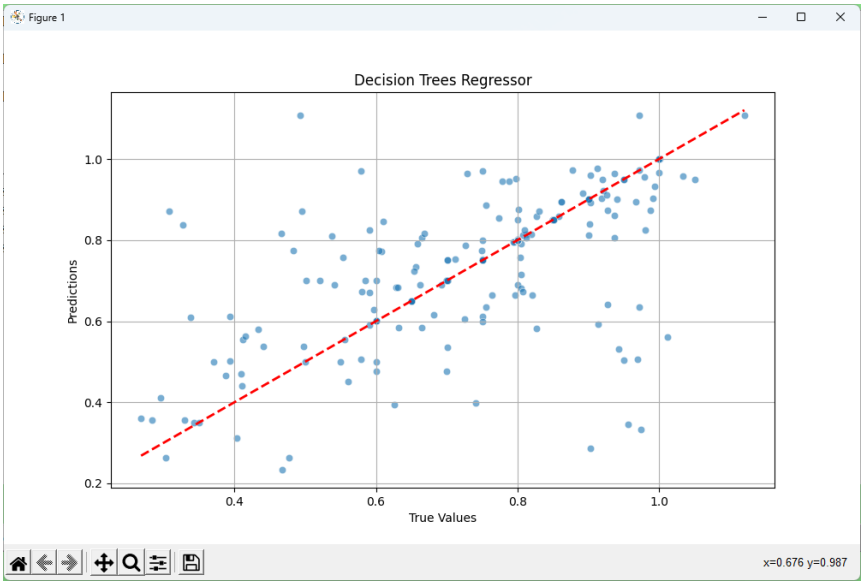

Fig 4: NaïveBayes Performance: True vs. Predicted Values.

In Fig. 5 Compared to Naïve Bayes, CatBoost typically shows a better fit due to its ability to handle categorical data efficiently and prevent overfitting. The GUI also provides evaluation metrics such as MAE, MSE, RMSE, and R^2 score, allowing users to compare the model's performance against other algorithms used in the study.

doi: 10.48047/ijiee.2025.15.4.19

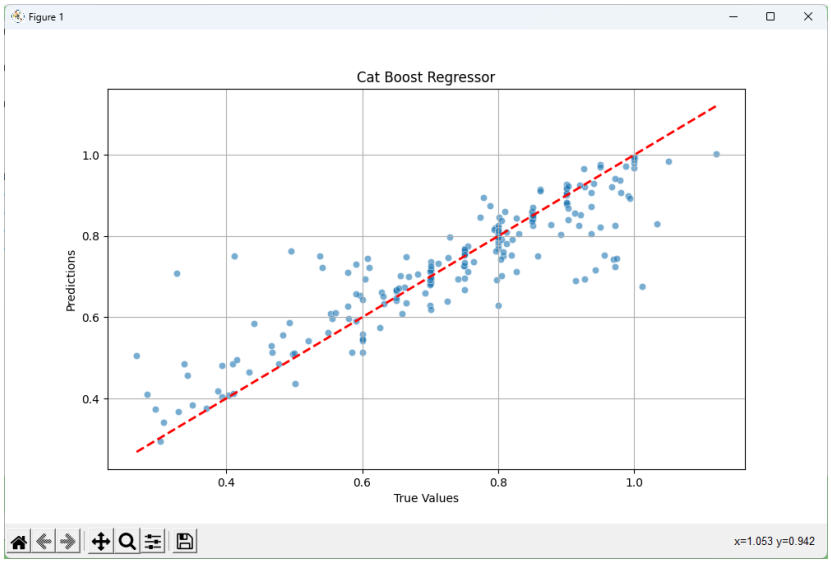

Fig 5: Catboost Regressor Performance: True vs. Predicted Values.

Table.1 Performance comparison of Algorithms.

Metric	Decision Tree Regressor	CatBoost Regressor
Mean Absolute Error (MAE)	0.10	0.05
Mean Squared Error (MSE)	0.03	0.01
Root Mean Squared Error (RMSE)	0.18	0.08
R-squared (R^2)	0.05	0.81

Table. 1 compares the performance of the Decision Tree Regressor and the CatBoost Regressor using four key regression metrics. The Mean Absolute Error (MAE) and Mean Squared Error (MSE) for CatBoost are significantly lower than those of Decision Tree, indicating that CatBoost makes more precise predictions with less deviation from actual values. The Root Mean Squared Error (RMSE), which represents the standard deviation of prediction errors, is also lower for CatBoost, confirming its superior accuracy. Most notably, the R-squared (R^2) score, which measures how well the model explains the variance in the data, is 0.81 for CatBoost compared to 0.05 for Decision Tree. This suggests that CatBoost has a strong predictive capability, while Decision Tree struggles to capture patterns in the dataset, making CatBoost the better choice for workforce productivity prediction.

REFERENCES

[1] Alam, Mohammad, Rosima Alias, and Mohammad Azim. 2018. 'Social Compliance Factors (SCF) Affecting Employee Productivity (EP): An Empirical Study on RMG Industry in Bangladesh', 10: 87-96. https://www.researchgate.net/publication/326733299

[2] Bhatia, Karan, Shikhar Arora, and Ravi Tomar. 2016. "Diagnosis of diabetic retinopathy using machine learning classification algorithm." In 2016 2nd international conference on next generation computing technologies (NGCT), 347-51. IEEE DOI: 10.1109/NGCT.2016.7877439.

[3] Kruppa, Jochen, Alexandra Schwarz, Gerhard Arminger, and Andreas Ziegler. 2013. 'Consumer credit risk: Individual probability estimates using machine learning', Expert systems with

applications, 40: 5125-31 DOI: https://doi.org/10.1016/j.eswa.2013.03.019. https://www.sciencedirect.com/science/article/pii/S0 957417413001693

[4] Balla, Imanuel, Sri Rahayu, and Jajang Jaya Purnama. 2021. 'GARMENT EMPLOYEE PRODUCTIVITY PREDICTION USING RANDOM FOREST', Jurnal Techno Nusa Mandiri, 18: 49-54 DOI: https://doi.org/10.33480/techno.v18i1.2210

[5] Attygalle, Dilhari, and Geethanadee Abhayawardana. 2021. 'Employee Productivity Modelling on a Work from Home Scenario During the Covid-19 Pandemic: A Case Study Using Classification Trees', Journal of Business and Management Sciences, 9: 92-100 DOI: 10.12691/jbms-9-3-1

[6] Ďurica, Marek, Jaroslav Frnda, and Lucia Svabova. 2019. 'Decision tree-based model of business failure prediction for Polish companies', Oeconomia Copernicana, 10: 453-69 DOI: 10.24136/oc.2019.022

[7] Mahoto, Naeem, Rabia Iftikhar, Asadullah Shaikh, Yousef Asiri, Abdullah Alghamdi, and Khairan Rajab. 2021. 'An Intelligent Business Model for Product Price Prediction Using Machine Learning Approach', 30: 147-59 DOI: 10.32604/iasc.2021.018944

[8] Sorostinean, Radu, Arpad Gellert, and BogdanConstantin Pirvu. 2021. 'Assembly Assistance System with Decision Trees and Ensemble Learning', Sensors, 21: 3580 DOI: https://doi.org/10.3390/s21113580

[9] Saad, Hamza. 2020. 'Use Bagging Algorithm to Improve Prediction Accuracy for Evaluation of Worker Performances at a Production Company', arXiv preprint arXiv:2011.12343 DOI: 10.4172/2169- 0316.1000257

[10] El Hassani, Ibtissam, Choumicha El Mazgualdi, and Tawfik Masrour. 2019. 'Artificial intelligence and machine learning to predict and improve efficiency in manufacturing industry', arXiv e-prints: arXiv: 1901.02256

[11] De Lucia, Caterina, Pasquale Pazienza, and Mark Bartlett. 2020. 'Does good ESG lead to better financial performances by firms? Machine learning and logistic regression models of public enterprises in Europe', Sustainability, 12: 5317 DOI: https://doi.org/10.3390/su12135317

[12] Ihya, Rachida, Abdelwahed Namir, Sanaa El Filali, Mohammed Ait Daoud, and Fatima Zahra Guerss. 2019. "J48 algorithms of machine learning for 58 Informatica 46 (2022) 49 –58 R. Obiedat et al. predicting user's the acceptance of an E -orientation systems." In Proceedings of the 4th International Conference on Smart City Applications, 1 -8. DOI: 10.1145/3368756.3368995