Employee Promotion with Machine Learning Approaches

Prosper Nosa Obayangbon | D3424757

School of Computing and Digital Technologies, Teesside University, United Kingdom

Abstract

Employee promotion is an essential part of effectively managing human resources. One should be right when making the decision to promote an employee based on the employee's previous and current results. This study developed models to forecast employee promotion by using machine learning algorithms and data sourced from Kaggle. The dataset contains 54,808 employees and a total of 12 features of the dataset that contain one of the target variables. The four models, namely, Random Tree Classifier, Decision Tree Classifier, Gradient Boosting, and Logistics Regression, were tested, and the best model among them was identified as the predictor. We used the Synthetic Minority Over-sampling Technique and a 70:30 data set division into testing and training, respectively, to manage the data imbalance. The Gradient Boosting model led the other models, giving the best results in accuracy (97%), precision (97%), recall (97%), and F1-Score (97%).

Introduction

Employee promotion is a critical aspect of organisational growth, but evaluating candidates can be time-consuming and complex. To streamline this process, data-driven techniques, particularly machine learning, are being explored to predict promotion suitability. This study aims to create a classification model using a Kaggle dataset to forecast promotion eligibility, revealing hidden patterns that can guide decisions and improve organisational performance. This approach can optimise resource allocation and enhance overall effectiveness.

Literature Review

Previous studies that show the intersection of human resource management and machine learning have emphasised the importance of predictive analytics in talented employee management and development. These studies have showcased the effectiveness of machine learning models in foreseeing who should or should not be promoted.

Keawwiset et al. (2021) used machine learning techniques like decision trees, random forests, and support vector machines to predict employee promotions, achieving high accuracy rates through validation of the Random Forest model with SMOTE oversampling.

Ibrahim et al. (2020) utilised machine learning models like gradient boosting, random forests, catboost, and extreme gradient boosting to predict employee promotions, highlighting the effectiveness of Catboost and XGBoost in high prediction accuracy.

Additionally, Lui et al. (2019) studied employee promotion prediction using algorithms like Random Forests, Logistic Regression, and AdaBoost. They found the Random Forest Classifier to be the optimal model, demonstrating high accuracy, recall, and precision metrics. These studies highlight the potential for machine learning in forecasting employee promotions.

Methodology

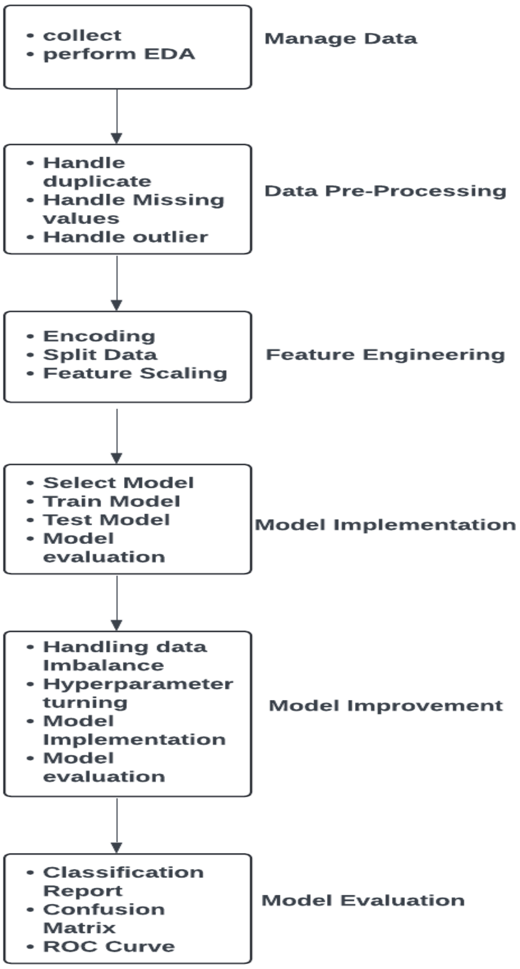

Figure 1: Methodology

Dataset

"HR Analytics: Employee Promotion Data" was sourced from Kaggle.com, created by a data scientist from a multinational company. This dataset comprises 54,808 instances and encompasses 13 attributes. Notably, the dataset incorporates both numeric and categorical data types across columns, utilising various hybridization sampling techniques and features. Among these attributes, 12 serve as input attributes, while one acts as the target attribute denoted as "is_promoted," featuring binary labels (0=No, 1=Yes). Table 1 below provides a detailed description of these features.

Table 1: Description of the dataset used in this project

Feature Name	Description
Employee_id	Unique identifier for each employee
department	Department in which the employee works
region	Region where the employee is employed (unordered)
education	Education level of the employee
gender	Gender of the employee
recruitment_channel	Channel through which the employee was recruited
no_of_trainings	Number of other trainings completed by the employee in the previous year
age	Age of the employee
previous_year-_rating	Rating received by the employee in the previous year
length_of_services	Length of service(in years) of the employee
awards_won	Indicator of whether the employee won any awards during the previous year (1 if yes, 0 if no)
avg_training_score	Average score in current training evaluations
Is_promoted	Target variable indicating whether the employee was promoted (1 if promoted, 0 if not promoted)

Exploratory Data Analysis (EDA)

The objective of this exploratory data analysis (EDA) is to uncover the underlying reasons behind employee promotions and identify the factors that are frequently considered significant in the promotion process. Utilising graphical techniques, researchers delve into the aspects influencing employee promotions to discern whether these factors play a pivotal role in the promotion decision-making process. Descriptive statistics were employed to compute, clarify, and summarise the datasets. Additionally, a correlation matrix was utilised to ascertain the relationships between numerical variables. The findings of this analysis are depicted in the figures below.

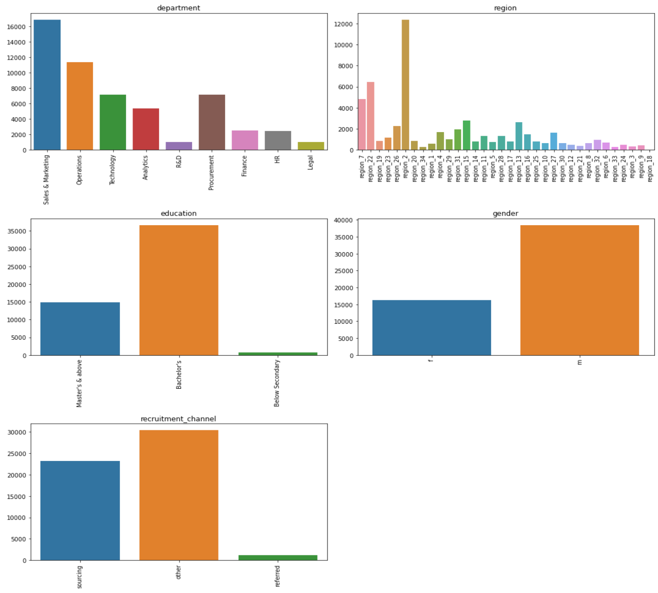

Figure 2: EDA for Categorical Features

Visualising the categorical variables through counts reveals, notably, that the organisation exhibits a higher proportion of male employees (70.24%) than females (29.76%). Sales and marketing dominate departments, with Region 2 housing the most employees. Most employees hold bachelor's degrees, and recruitment mainly favours “other“ channels.

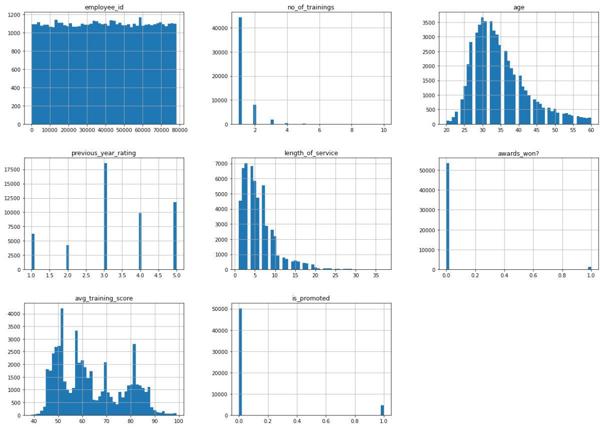

Figure 3: EDA of Numerical Features

From visualising numerical features, most employees received a rating of 3 out of 5, and length of service shows a right-skewed distribution, with most falling within 1 to 7 years. A small percentage of employees (2.32%) received awards, while 8.32% were promoted last year.

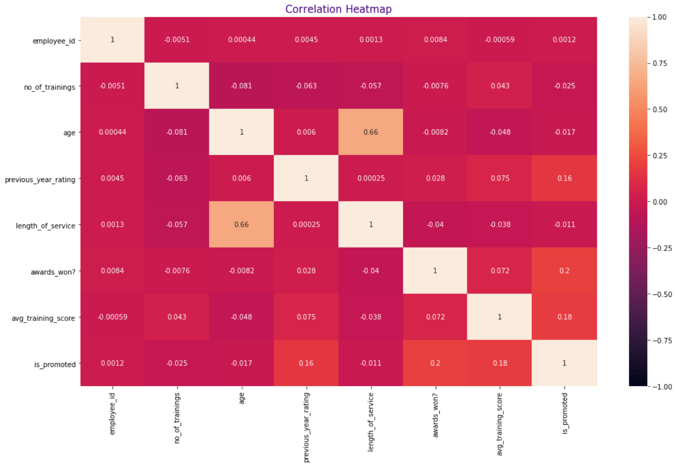

Figure 4: Correlation matrix for numerical variables

We found out that the awards won, average rating, and previous year rating all have a positive correlation with them, and the target variable (’is_promoted’)

Data Preprocessing

Check Missing Values

Missing values significantly affect classification model performance (Zahim et al., 2018). Addressing missing values is crucial before dataset analysis. The "isnull().sum()" function is used to identify them. Reports show "education" and "previous_year_rating" features contain random missing values, as depicted in Figure 5.

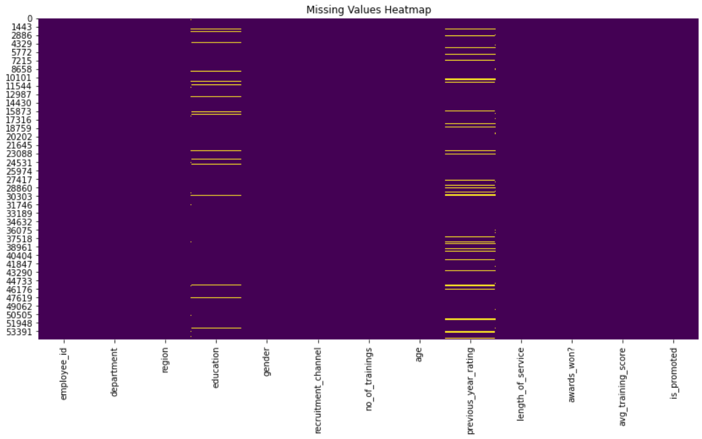

Figure 5: Rows with missing values in the dataset

To handle missing values, we investigated and found that employees with missing previous-year ratings were newly hired with a service length of 1. Consequently, we imputed these missing values with 0. For the "education" feature, being categorical, we imputed missing values with the most frequent value (Shumeiko & Rozora, no date).

Handling Outliers

Outliers are typically observed in numeric data, and in this dataset, outliers were only plausible in the average training score and length of service columns due to their continuous nature (Meynard et al., 2019). While no outliers were detected in the average training score, some were observed above the upper fence in the length of service column, as shown in Figure 6 below. Consequently, we capped these outliers at their maximum value (Fernández et al., 2022).

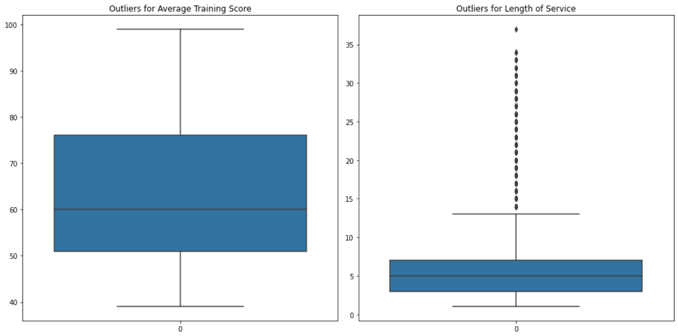

Figure 6: Outlier of numeric data

Check Duplicate Data

The “duplicated().sum()” function is used to identify duplicate data. Fortunately, in this dataset, no duplicate data was found.

Feature Engineering

Label Encoder

The scikit-learn package used for machine learning algorithms mandates numerical input and output. However, certain columns such as "department," "region," "education," "gender," and "age" in our dataset contain textual or categorical data. Thus, prior to training and testing machine learning models, this textual or categorical data must be encoded into integer values. The LabelEncoder package from the Scikit-learn preprocessing library accomplishes this task. It transforms all categorical values into a range between 0 and n-1, where n represents the total number of classes.

Split Data

Before constructing the machine learning model, it's essential to divide the dataset into training and testing sets using the “train_test_split()” function. Following the convention, we adhere to the 70:30 ratio (train:test) (Ayoubi et al., 2018). 70% of the data will be allocated for training the machine learning model, while the remaining 30% will be reserved for evaluating the models.

Feature Scaling

In the feature scaling stage, the standard scaler() function from the scikit-learn preprocessing package is employed to scale X_train and X_test. This process aims to standardise features by removing the mean and scaling to unit variance (Pedregosa et al., 2011).

Model Implementation

In this study, we employ four machine learning algorithms: Logistic Regression Classifier (LRC), Decision Tree Classifier (DTC), Random Forest Classifier (RFC), and Gradient Boosting Classifier (GBC). Each algorithm is trained using the training data to create a predictive model.

Logistic Regression Classifier (LRC)

Logistic Regression is a common classification method that utilises linear discriminants. It's a fundamental tool in modelling, particularly for forecasting project analyses (Jaffar et al., 2019).

Decision Tree Classifier (DTC)

The Decision Tree Classifier constructs a tree model using gain ratio, making it a preferred choice for training modules by human resource teams (Saxena et al., 2021). While decision trees offer good accuracy and robustness, they may suffer from overfitting due to repeated subtrees (Zhou et al., 2021).

Random Forest Classifier (RFC)

The Random Forest algorithm, a popular ensemble learning technique, builds multiple decision trees using subsets of predictors chosen randomly. It primarily employs bootstrap aggregation, or bagging, for tree learning. In bagging, each model utilises a bootstrapped dataset, aggregated to make predictions via a simple majority vote (Punnoose & Ajit, 2016).

Gradient Boosting Classifier (GBC)

Gradient Boosting (GB) is a machine learning method that combines decision trees to improve prediction accuracy. It iteratively trains these trees to minimise errors, making it useful for handling categorical data and nonlinear relationships. (Breiman 1997).

Model Evaluation

Following the selection of machine learning algorithms, we evaluated their performance on the split dataset using various metrics such as accuracy, precision, recall, and F1-score. These metrics provided valuable insights into the algorithms' ability to accurately predict which employees should or should not be promoted.

Furthermore, we employed confusion matrices to delve deeper into the performance of our models. These matrices provided a detailed breakdown of true positives, false positives, true negatives, and false negatives for each class in the predicted data. By analysing these components, we gained a comprehensive understanding of the models' proficiency in identifying positive and negative instances of employee promotion. Notably, the Gradient Boosting algorithm emerged as the top-performing model, exhibiting high true positive and true negative values (15999 and 444, respectively) and low false positive and false negative values (0 and 0, respectively).

Model Improvement

This step is to validate if the highest-performing model is not a result of overfitting and to ensure that the performance of the low-performing model is also improved.

Handling Imbalanced Data

The dataset exhibits an imbalance in the target variable, with about 91.48% of employees labelled "No" and only 8.52% labelled "Yes" for promotion. To address this issue, we employed SMOTE resampling, which synthetically generates minority-class samples to balance the dataset. Figure 7 shows visualisation before and after resampling.

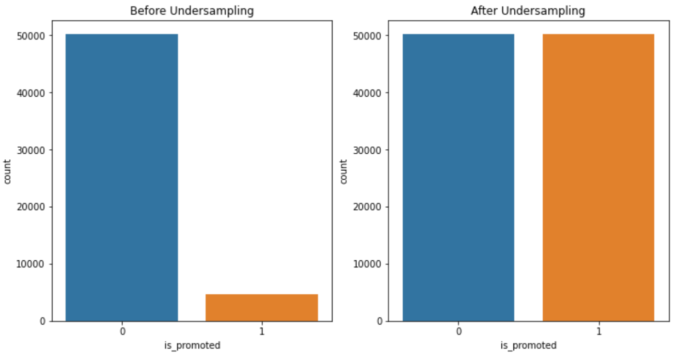

Figure 7: Visualisation of the target variable before and after data balancing

Hyperparameter Tuning

To improve the performance of our classification models by conducting hyperparameter tuning using grid search and K-fold cross-validation. The training dataset was split into five folds, each iteratively used for training and testing. The algorithms were retrained and evaluated using bar charts, confusion matrices, and ROC curves. The Gradient Boosting model emerged as the top-performing model, achieving 97% and 96% accuracy on both training and test datasets. Other models included random forest, decision tree, and logistic regression.

Results and Discussion

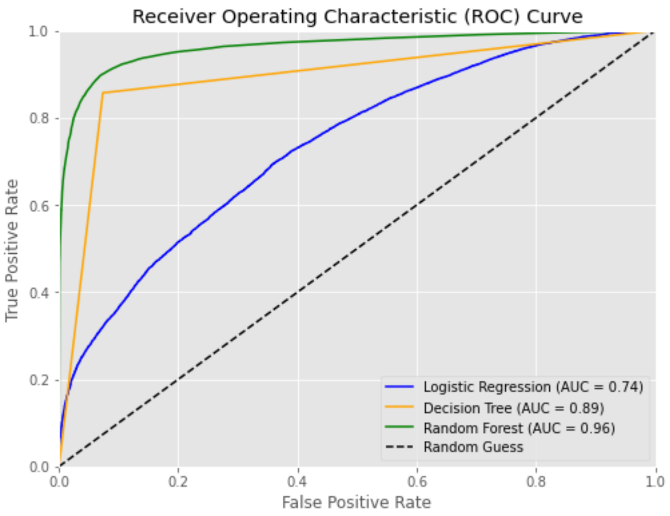

Figure 8: ROC Curve Detailing the Models

Table 2: Result of the Models

Model	Train Accuracy	Test Accuracy	Precision	Recall	F1-score
LR	0.66	0.66	0.66	0.66	0.66
DT	0.98	0.89	0.90	0.90	0.90
RF	0.97	0.90	0.91	0.90	0.90
GB	0.97	0.96	0.97	0.97	0.97

Discussion

Logistic regression, while simple and interpretable, may not capture the complex interactions among features effectively, potentially limiting its predictive power. Decision trees, on the other hand, are capable of capturing complex patterns but are prone to overfitting, which could lead to poor generalisation on unseen data.

Random forests address the overfitting issue by aggregating multiple decision trees, resulting in improved performance and robustness. However, they may still struggle to capture subtle relationships in the data.

Gradient boosting, with its iterative approach to learning, excels at capturing intricate relationships and minimising errors. This makes it well-suited for the task of predicting employee promotions, where subtle interactions among employees may play a significant role.

The superior performance of gradient boosting, as evidenced by its high test accuracy and AUC, suggests that it effectively captures the underlying patterns in the dataset and generalises well to unseen data. This makes it the preferred choice for this project, as it offers the best balance of predictive accuracy and generation capacity.

The work by Alqahtani et al. (2022) supports this decision by demonstrating the effectiveness of gradient boosting ahead of other algorithms in predicting employee promotions, with an accuracy of 93%. By achieving high predictive accuracy and effectively capturing complex relationships in HR datasets, gradient boosting emerges as a reliable and robust choice for promotion prediction tasks.

Future Work and Conclusion

In the future, this project could be expanded to include additional features or data sources that may further enhance the predictive accuracy of the model. For example, incorporating performance metrics such as project completion rates or client satisfaction scores could provide valuable insights into employees' overall contributions and potential for promotion.

Furthermore, exploring more advanced techniques such as deep learning or natural language processing could uncover hidden patterns in textual data such as employee reviews or feedback, which may offer valuable insights into employees' strengths and areas for improvement.

In conclusion, this project demonstrates the potential of machine learning in predicting `employee promotions and offers valuable insights into the factors that influence advancement within an organization. By leveraging advanced techniques and carefully curated data, organisations can make more informed decisions about talent management and workforce planning, ultimately leading to improved employee satisfaction, retention, and overall organisational performance.

References

Alqahtani, F.A. and Almaleh, A. (2022) *Analysis and Prediction of Employee Promotions Using Machine Learning.* IEEE

Ayoubi, S.*et al.* (2018a) 'Machine Learning for Cognitive Network Management', *IEEE Communications Magazine,* 56(1), pp. 165. Available at: https://doi.org/10.1109/MCOM.2018.1700560

Breiman, L. *arcing-the-edge. Comparative Analysis of Handling Missing Values and Imputation Techniques on Gene Expression Data.*

Fernández, Á, Bella, J. and Dorronsoro, J.R. (2022) *Supervised outlier detection for classification and regression.* Elsevier BV

Ibrahim, A. *et al. Performance of CatBoost classifier and other machine learning methods.*

Jaffar, Z., Noor, W. and Kanwal, Z. *Predictive Human Resource Analytics Using Data Mining Classification Techniques.*

Jeon, Y. and Lim, D. (2020) 'PSU: Particle Stacking Undersampling Method for Highly Imbalanced Big Data', *IEEE Access,* 8, pp. 131920-131927. Available at: https://doi.org/10.1109/ACCESS.2020.3009753

Kaewwiset, T., Temdee, P. and Yooyativong, T. (2021a) *Employee Classification for Personalized Professional Training Using Machine Learning Techniques and SMOTE.* IEEE

Kaewwiset, T., Temdee, P. and Yooyativong, T. (2021b) 'Employee Classification for Personalized Professional Training Using Machine Learning Techniques and SMOTE', IEEE Available at: 10.1109/ECTIDAMTNCON51128.2021.9425754.

Liu, J.*et al.* (2019) 'A Data-driven Analysis of Employee Promotion: The Role of the Position of Organization', IEEE Available at: 10.1109/SMC.2019.8914449.

Meynard, C.N., Kaplan, D.M. and Leroy, B. (2019) *Detecting outliers in species distribution data: Some caveats and clarifications on a virtual species study.* Wiley

Punnoose, R. *Prediction of Employee Turnover in Organizations using Machine Learning Algorithms A case for Extreme Gradient Boosting.*

Saxena, M., Bagga, T. and Gupta, S. (2021) *Fearless path for human resource personnel through analytics: a study of recent tools and techniques of human resource analytics and its implication.* Springer Science and Business Media LLC

Shumeiko, D. and Rozora, I. *Handling Missing Values in Machine Learning Regression Problems.*

Zhou, H.*et al.* (2020) *A feature selection algorithm of decision tree based on feature weight.* Elsevier BV