Makine Öğrenimi Algoritması Kullanarak Kişisel Göstergelere Dayalı Çalışan Teşviklerinin Tahmini

***

Forecasting Employees' Promotion Based on Personal Indicators by Using a Machine Learning Algorithm

Yasmine Aya IBRIR[1]

Mahmut ÇAVUR[2]

DOI:10.33461/uybisbbd.1471499

Makale Bilgileri

Makale Türü:
Araştırma Makalesi

Geliş Tarihi:
21.04.2024

Kabul Tarihi:
18.07.2024

Öz

Terfi, çalışanın kendini geliştirmesi ve işin yükünü ve sorumluluğunu, kendisine yüklenen pozisyonla birlikte taşıma isteği için motive etmenin bir aracı olarak hareket eder. Geleneksel yöntemler ile yapılan terfilerin hakkaniyeti ve ölçülebilirliği nicel olarak ölçülemediği için farklı yöntemlere ihtiyaç duyulmaktadır. Son yıllarda şirketlerde bilgi sistemlerinin kullanımın yaygınlaşması ile çalışanlara ait performans bilgileri gibi birçok bilgi dijital ortamda tutulmaya başlandı. Yine ver bilimlerinin gelişmesi ve birçok alana uygulanması ile birlikte çalışanlara ait bu verilerin değerlendirmesinde makine öğrenmesi ve yapay zekâ algoritmalarının kullanımı yaygınlaştı. Bu çalışma, çeşitli özelliklere dayalı olarak bir kuruluş içindeki çalışanların terfilerini tahmin etmek için sağlam bir çerçeve oluşturmayı amaçlamaktadır. Bu özellikler, eğitim sayısını, önceki yıl derecelendirmelerini, hizmet süresini, kazanılan ödülleri ve ortalama eğitim puanını içermekle birlikte bunlarla sınırlı değildir. Çalışmanın amacı, kuruluşların bilinçli terfi kararları almaları için güvenli bir araç sağlamak ve bu çerçevenin diğer tahmin problemlerine genelleştirilebileceğini göstermektir. Deneysel sonuçlar XGBoost modelinin doğruluk açısından en verimli model olduğunu göstermektedir. XGBoost, %94 doğruluk ve ROC AUC, %94 duyarlılık ve %94 hassasiyetle bellek kullanımı verimliliği, doğruluk ve çalışma süresi açısından üstün bir algoritma olarak kabul edilmektedir.

Anahtar Kelimeler: *Çalışan Terfisi, Çalışan Terfi Tahmin Çerçevesi, XGBoost, Makine Öğrenimi, Denetimli Öğrenme.*

Article Info

Paper Type:
Research Paper

Received:
21.04.2024

Accepted:
18.07.2024

Abstract

Promotion is a tool to motivate employees to improve themselves and take on the burden and responsibility of the position assigned to them. Due to the fairness and measurability of promotions conducted by traditional methods needing to be quantifiable, different methods are required. In recent years, with the widespread use of information systems in companies, much information, such as performance data of employees, has started to be stored digitally. Additionally, with the development of data sciences and their application in many fields, machine learning and artificial intelligence algorithms in evaluating this data have become widespread. This study aims to establish a robust framework to predict employee promotions based on various features. These features include but are not limited to the number of training sessions attended, previous year ratings, tenure, awards received, and average training scores. The study aims to provide organizations with a reliable tool to make informed promotion decisions and demonstrate that this framework can be generalized to other prediction problems. Experimental results show that the XGBoost model is the most efficient in terms of accuracy. XGBoost is considered a superior algorithm with 94% accuracy, 94% ROC AUC, 94% sensitivity, and 94% precision, excelling in memory usage efficiency, accuracy, and runtime.

Keywords: *Employee Promotion, Prediction, XGBoost, Machine Learning, Supervised Learning.*

Atıf/ to Cite (APA): Ibrir, Y.A. and Çavur, M. (2024). Forecasting Employees' Promotion Based on Personal Indicators by Using a Machine Learning Algorithm. International Journal of Management Information Systems and Computer Science, 8(2), 75-98. DOI: 10.33461/uybisbbd.1471499

[1] Kadir Has University, Management Information System Department, yasmineaya.ibrir@stu.khas.edu.tr, Istanbul, Türkiye.

[2] Dr. Kadir Has University, Management Information System Department, mahmut.cavur@khas.edu.tr, Istanbul, Türkiye.

1. INTRODUCTION

Promotion is regarded as one of the most important issues in any organization because it is essential for administrative development and a means of motivating employees to pursue self-development. Promotion has always been an important research point in several areas, including human development. Nowadays, many organizations need help with job promotion and professional stability. The ability of institutions to achieve their goals depends largely on the extent to which the administration succeeds in providing sufficient satisfaction and setting up a promotion program according to objective criteria that permit achieving organizational effectiveness. Businesses must be able to forecast what will happen to their client base and staff so that they can take appropriate actions before the "promotion" process. The term "promotion" is defined as a means of an employee's career advancement and development and is linked to the employee's level of performance.

The issue that this study attempts to solve is the difficulty organizations have in accurately forecasting and overseeing employee promotions. Promotions may cause unhappiness and inefficiency without an impartial and reliable prediction framework, which lowers employee morale and corporate performance. Therefore, we clarify how machine learning (ML) models enhance the ability of enterprises to forecast employee promotions. Not only the type of ML but also the parameters are critical to predicting the promotions. Therefore, defining, deciding and explaining those parameters affecting the promotions is essential. Finally, choosing, adopting, and/or developing ML algorithms for promotion prediction is necessary. Compared to conventional techniques, machine learning models may greatly improve the accuracy of employee promotion forecasts. Reliability, years of service, practical efficiency, and credentials are important indicators of when an individual will be promoted. The models' forecast accuracy will increase with new criteria like total score, work percentage, and years to retirement. Since there can be one or more machine learning models that perform better in terms of prediction accuracy and reliability than the baseline models among the numerous models, we selected and adopted several ML algorithms to compare their performances with several metrics.

In this thesis study, we aimed to set up a robust framework that can be used and generalized to predict problems in the business, not just the problem of predicting employee promotion. Employees are promoted based on their practical efficiency and loyalty in performing their jobs, as well as their years of service and qualifications. The essential idea is to promote the right man to the right place, thus being the path to success for the company. We believe that our framework can be successfully used in choosing the appropriate employee according to the organizational hierarchy without fraud or prior knowledge. The primary goal of the learning models is to anticipate the individual's promotion within a particular period reliably. We will reward workers based on their performance and workplace behavior, utilizing these aspects from the HR Analytics Vidhya data. We create a framework for predicting employee promotion using machine learning algorithms. We chose existing features and added new several features, including total score, work fraction, work start year, years remaining to retire, and performance. We compare the experimental results with different baseline models using evaluation performance metrics.

2. RELATED WORKS

This section outlines the related research on predicting employee advancement and turnover.

2.1. Predicting Employee Promotion

Employee promotion refers to an employee's upward progress within the organization to a new or higher job position, tasks, and responsibilities. Promotion is an important step in the life cycle of both employees and organizations. Choosing the correct candidate for promotion at the right moment is a critical issue. According to many studies, several strategies rely on machine learning to overcome real-world challenges, particularly in human resource management, and employee promotion is one of them.

Managers spend a great deal of time recruiting capable employees. Promotion is an issue that both businesses and employees are concerned about. On the one hand, promotion is a strategy used by businesses to pick exceptional people and boost their competitiveness. Employee promotion, systems, and organizational performance have a good relationship (Chen, Hsu and Wu, 2012). On the other hand, advancement prospects have a higher impact on employee performance, as do leadership, job promotion, and work environment. These components work together to improve employee performance (Febrina, 2017). It is regarded as an excellent policy to replace gaps in higher-level positions through internal promotions since such advancements give encouragement and motivation to employees while also removing sentiments of stagnation and discontent (Li et al., 2021).

According to certain research, internal promotion in organizations is influenced by a variety of factors, including age (Li et al., 2021; Long et al., 2018; Machado and Portela, 2021) gender, education background (Jantan and Hamdan, 2010; Long et al., 2018), and job experience (De Pater et al., 2009; Long et al., 2018). Categorization is one of the most important tasks in data mining, which is used to extract knowledge from massive amounts of data. This technique is frequently utilized in a variety of sectors, although it has received less attention in HRM. Using an employee's performance data, an experiment was carried out to illustrate the practicality of recommended classification techniques (Jantan and Hamdan, 2010).

Higher compensation is always followed by a job promotion and increased experience in problem-solving, loyalty, honesty, and responsibility at work, according to Febrina's (2017) research. Based on his findings, it is possible to infer that leadership, job advancement, and job environment all positively and substantially influence staff performance at the bank. These components work together to improve employee performance (Febrina, 2017).

Businesses must prioritize human capital in the age of big data and Industry 4.0. Liu and colleagues (2019) emphasize that people should work in various places and divisions to broaden their experiences. Working in jobs where mobility is more reliable, resources are available, or experience is accessible can help the worker advance. Their study used supervised learning to predict staff advancement and build models using logistic regression, random forests, and AdaBoost. In the end, RF performs better and has a reasonable time consumption (Liu et al., 2019).

Long and colleagues (2018) have used machine learning to predict employee advancement using data from a Chinese state-owned firm. Two types of features were created based on five methodologies by extracting personal basic information and position information from this data. Using correlation analysis, they validate the efficiency of attributes in estimating employee advancement. According to the findings of the study, the influence of post features on promotion is stronger than that of personal basic characteristics (Long et al., 2018).

Job classifications are frequently established using the k-means clustering technique, which is a common method of job classification establishment. Sarker (2018) and colleagues used a decision tree algorithm to swiftly identify employees and make suitable decisions. Employees are divided into three groups based on their level of performance. According to the results, support vector machines outperform the other classifiers in terms of accuracy (Sarker et al., 2018).

Although substantial progress has been made using big data analytic technologies in human resource management, research on the mining of promotion characteristics is limited, and further research is needed. Thus, using data from Analytics Vidhya, we build various promotion attributes and predict using machine learning methods.

2.2. Predicting Employee Turnover

Employee turnover is seen as a critical issue for all firms. To address this issue, organizations are now relying on machine learning approaches to forecast employee turnover. Employee turnover can be viewed as a defacement of the organization's intellectual capital. The literature study focuses on the strategies and techniques provided by various researchers for forecasting employee attrition.

A team of researchers has proposed a new model for forecasting employee attrition based on machine learning using XGBoost. It is recognized as a superior algorithm in terms of memory use efficiency, accuracy, and running time. The model provided in this research has a very low rate of less than 30% and an accuracy of around 90%. A total of 14 factors have a greater effect on the attrition rate than any other component - frozen Promotions and Salary Hikes, Imbalance of Work-Life, Employee Misalignment, Unsuitable Behavior, and Inadequate Professional Skills (Jain and Nayyar, 2018).

In this study1, various machine learning techniques have been implemented DT, RF, and SVM, it is possible to infer that RF outperforms. In the case of employee attrition, an estimate was made as to whether a person would quit the organization. Using this approach, the business may choose the individuals who have the highest likelihood of leaving the organization and then provide them with specific incentives (Jain, Jain and Pamula, 2020).

In another study, a strategy for selecting features to reduce the dimension of the feature space was described. The recommended feature selection improves the predictor's performance. This paper offers a three-stage method for constructing an accurate employee attrition prediction model. The parameters of the logistic regression model are validated by assessing their fluctuations when trained through several bootstraps. The results suggest that the "max-out" feature selection strategy improves the F1-score performance metric (Najafi-Zangeneh et al., 2021).

A research study insists that employee turnover in the IT industry is significant. Often, their early attrition is the result of company-related or personal concerns. A correlation matrix in the form of a heatmap was developed to determine the essential factors that may affect the attrition rate. The Random Forest classifier was revealed to be the best model for predicting IT staff attrition (Bandyopadhyay and Jadhav, 2021).

Employee promotions are a crucial component of keeping a motivated and skilled workforce. Prior to the "promotion" process, firms must be able to predict what will happen to their client base and workforce. With machine learning, we may identify employees who are most likely to be promoted based on prior data such as degree, experience, age, ratings, and overall score, using primarily the XGBoost model and other models. This framework can be used and generalized to all prediction problems, not just our problem of predicting employee promotion.

We believe that, without fraud or prior information, our approach may be effectively utilized to select the proper individual according to the organizational hierarchy. We consider several elements, including traditional employee traits and work history. Favoritism and family must be excluded from these accounts to prevent the problem from being passed on to someone else.

3. MATERIAL AND METHOD

3.1. Data Description

Analytics Vidhya Data Analysis provided the data (Table 1) used in this study. The dataset has 14 characteristics 54808 records for train data and 23490 records for test data. Not all the 14 features are considered in our work for employees' predictions of promotions. We will choose the relevant aspects and add new ones that impact the employee's promotion as an important indication for the promotion process. Data exploration is a two-step process that involves identifying the data type and category of the variables used to make up an empirical data set.

The dataset contains a target feature identified by the variable that is promoted. "No" denotes an employee who did not receive the promotion, and "Yes" represents an employee who did. If this training process is repeated over time and conducted on relevant samples, the predictions generated in the output will be more accurate.

In this paper, we explore the Analytics Vidhya dataset step-by-step. The methodologies of variable identification, univariate analysis, bivariate analysis, and multivariable analysis were applied to the Data Exploration phase of the data processing process for the first time.

Table 1: Attributes description and identification

Attributes	**Description**	**Data Type**	**Variable Category**	**Type of Variable**
Employee ID	Unique ID for the employee	Numeric	Continuous	Predictor
Department	Department of the employee	Character	Categorical	Predictor
Region	Region of employment (unordered)	Character	Categorical	Predictor
Education	Educational level	Character	Categorical	Predictor
Gender	Gender of the employee	Character	Categorical	Predictor
Recruitment channel	Channel of recruitment for the employee	Character	Categorical	Predictor
No. of training	No of other trainings were completed in the previous year on soft skills, technical skills, etc.	Numeric	Categorical	Predictor
Age	Age of the employee	Numeric	Continuous	Predictor
Previous year rating	Employee rating in the prior year	Numeric	Categorical	Predictor
Length of service	Length of service in years	Numeric	Continuous	Predictor
Awards won	If awards were won during the previous year, then 1 else 0	Numeric	Categorical	Predictor
Avg training score	The average score in current training evaluations	Numeric	Continuous	Predictor
KPIs met >80%	If the percentage of KPIs (Key Performance Indicators) >80%, then 1, else 0	Numeric	Categorical	Predictor
Is promoted	(Target) Recommended for promotion	Numeric	Categorical	Target variable

3.2. Descriptive Statistics

3.2.1. Uni-variate analysis

Univariate analysis is the most basic type of statistical analysis. Continuous and categorical variables are investigated in the univariate analysis. We investigated various techniques and statistical measures for categorical and continuous variables individually. Several statistical metrics and visualization approaches describe this type of relationship.

We constructed the descriptive statistics of the dataset. We considered the following variables: count, mean, standard deviation (std), minimum and maximum values (min/max), and 25%/50%/75% 95% percentile. **Hata! Başvuru kaynağı bulunamadı.** is an excerpt from the full d ataset.

Count

The count of a dataset is simply the number of observations, denoted as n.

$$Count = n \qquad (1)$$

Mean

A dataset's mean (average) is the sum of all observations divided by the number of observations.

$$Mean\ (\mu) = \frac{1}{n}\sum_{i}^{n} 1x_i \qquad (2)$$

where x_i represents each observation in the dataset.

Standard Deviation (Std)

The standard deviation measures the amount of variation or dispersion in a dataset.

$$\text{Standard Deviation } (\sigma) = \sqrt{\frac{1}{n}\sum_{i}^{n} = 1(x_i - \mu)\,2} \qquad (3)$$

where μ is the mean of the dataset.

Minimum and Maximum Values (Min/Max)

The minimum value is the smallest observation in the dataset, and the maximum value is the largest observation in the dataset.

$$\text{Minimum} = min(x_1, x_2, \ldots\ldots, x_n) \qquad (4)$$
$$\text{Maximum} = max(x_1, x_2, \ldots\ldots, x_n) \qquad (5)$$

Percentiles (25%, 50%, 75%, 95%)

Percentiles are values below which a certain percentage of observations in a dataset fall.

- 25th Percentile (First Quartile, Q1): The value below 25% of the observations falls.

$$\text{Q1} = P_{25} \qquad (6)$$

- 50th Percentile (Median, Q2): The value below 50% of the observations falls.

$$\text{Q2} = P_{50} \qquad (7)$$

- 75th Percentile (Third Quartile, Q3): The value below 75% of the observations falls.

$$\text{Q3} = P_{75} \qquad (8)$$

Figure 1 represents the bar chart of the variables; these are the value ranges of all features. Every feature has a different distribution of values. We looked at each one independently to get a better and deeper understanding of the characteristics.

Using machine learning models with imbalanced classes often leads to very poor results that are completely biased towards the class having a higher distribution. Clearly, the data needs to be balanced; a 91% and 9% ratio between promoted and non-promoted employees is very unbalanced.

Figure 1: Numeric features distribution

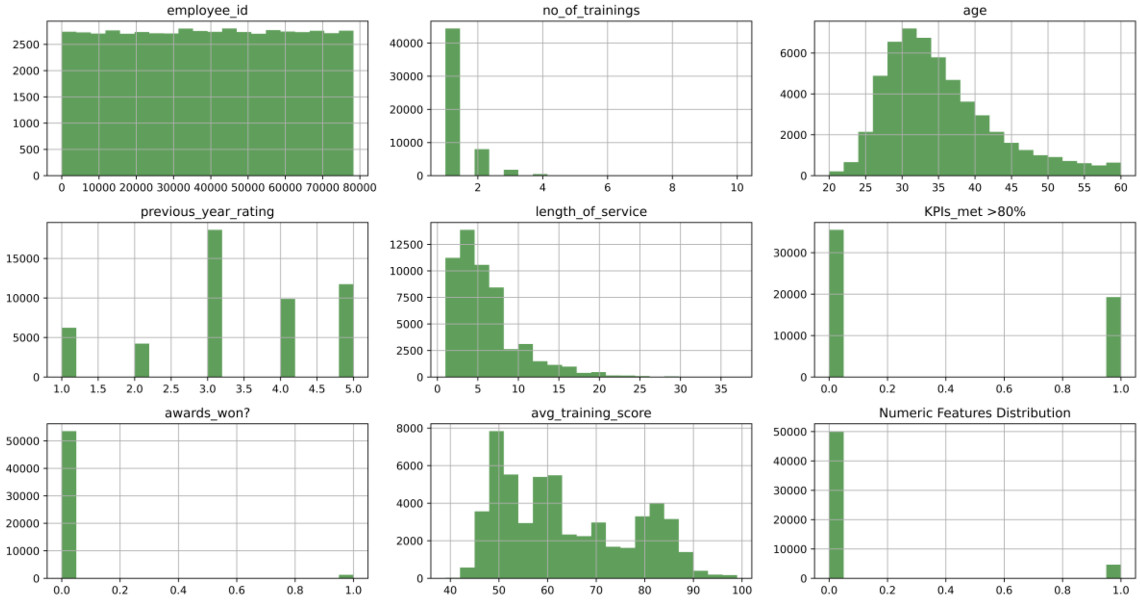
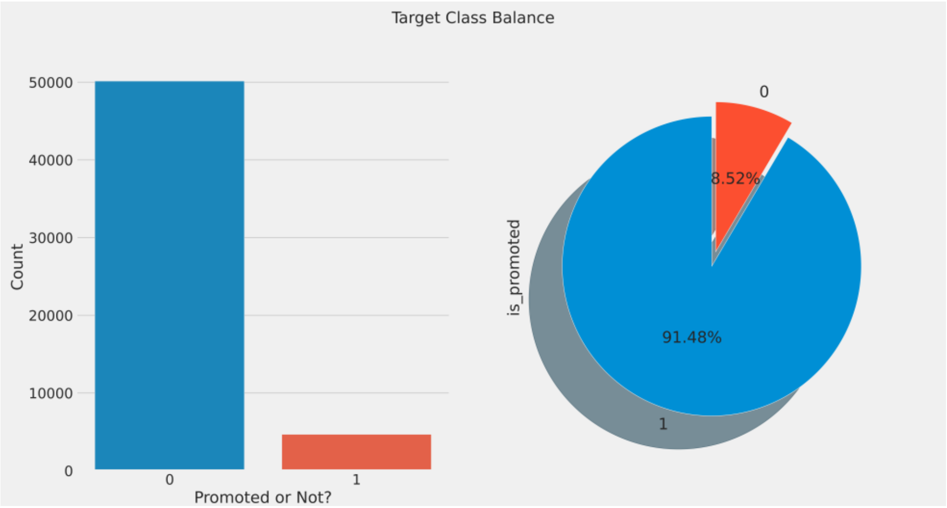


Figure 2: Is Promoted Distribution



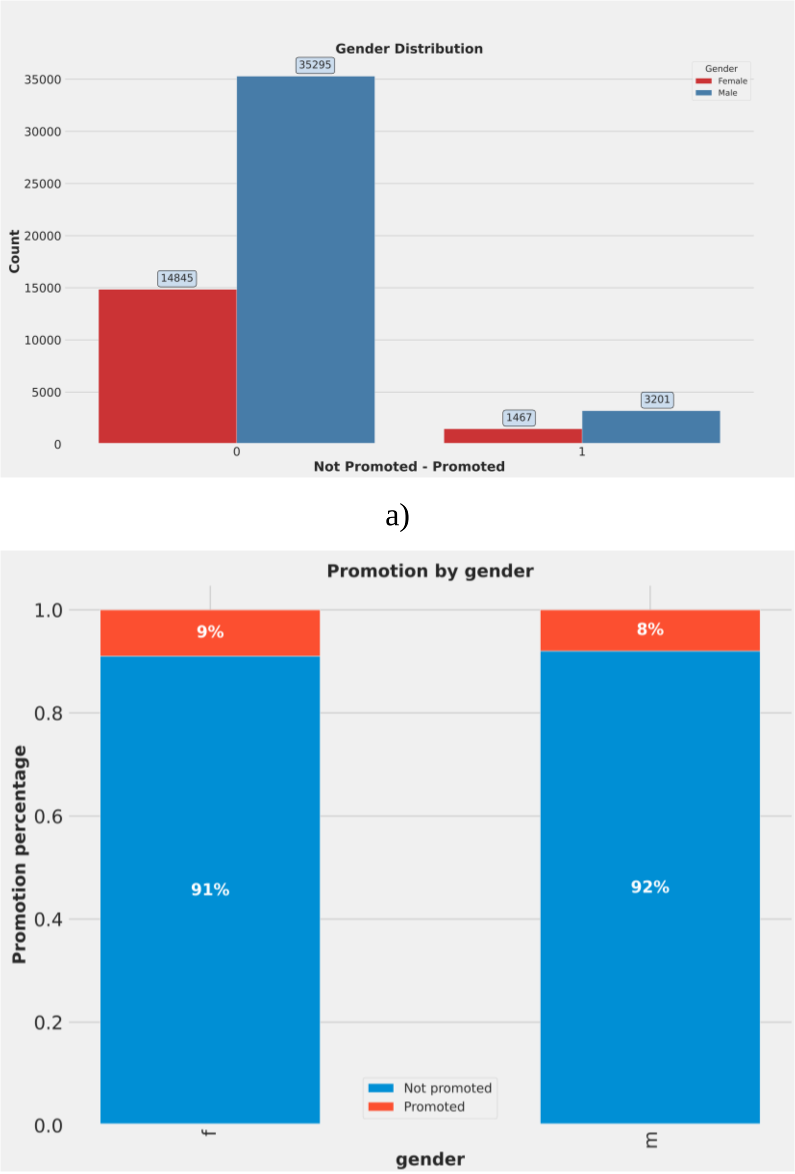
3.2.2. Bi-variate analysis

In this section, we examine variables at a predetermined significance threshold. Bivariate analysis may be used for any grouping of absolute, categorical, and continuous variables. During the analytic process, many strategies are utilized to manage these groups. The following are some examples of the specific combinations that are possible.

- ***Gender versus Employee Promoted***

According to **Hata! Başvuru kaynağı bulunamadı.**, male employees are promoted at a higher r ate than female employees. Furthermore, male employees continue to be promoted more than female ones. As previously stated, women are in the minority, but when it comes to promotion, they compete head-to-head with their male counterparts.

Figure 3: Gender versus Employee Promoted a) in number, b) in percentage



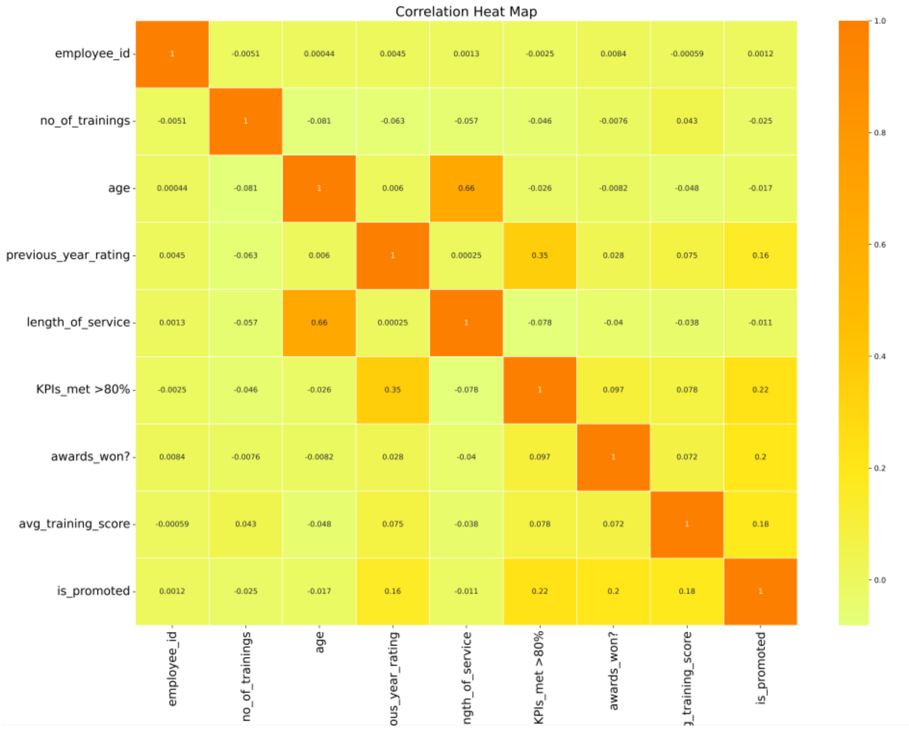
a)

b)

3.2.3. Multivariate analysis

Multivariate analysis is based on multivariate statistics concepts, simultaneously involving observing and analyzing several statistical result variables. First, we will examine the association between the numerical columns using the Correlation Heatmap.

Figure 4: Correlation Heat map



KPIs and the previous year's ratings are correlated to some degree, implying some relationship. However, we will do feature engineering before modeling to avoid multicollinearity. This heatmap shows the correlation between the columns, which is highly beneficial for regression issues.

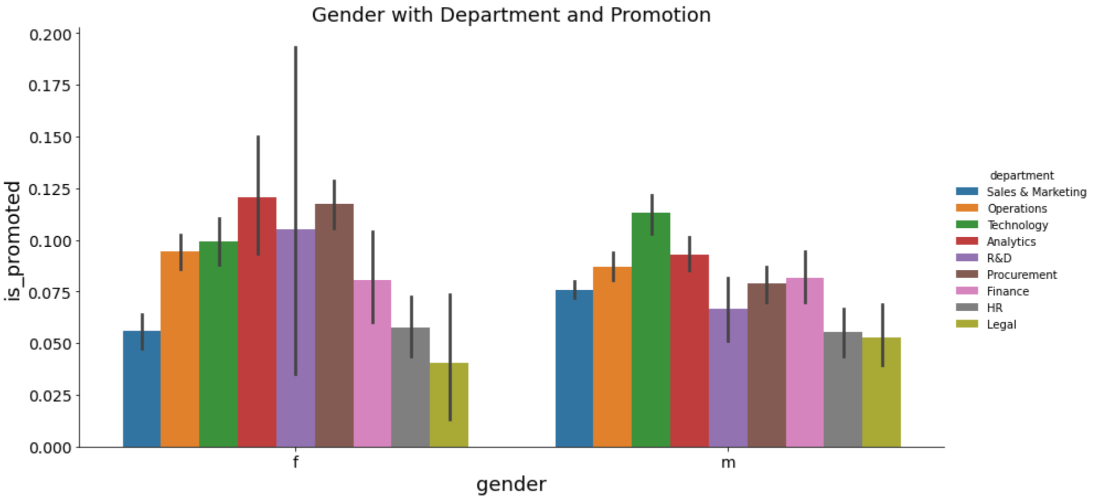
3.3. Data Visualizations

In this part, data visualization is conducted on continuous and categorical variables to understand the link between these variables and our objective variables. Graphs are the most effective way to comprehend the behavior of characteristics and the relationships between them. Here is one example:

- ***Gender with Department and Promotion***

Hata! Başvuru kaynağı bulunamadı. shows the gender breakdown by department and p romotion. This figure represents the distribution of females and males in the department section. There are varying percentages in the departments, and for females, there are higher promotions in the two departments of analytics and procurement. Unlike males, technology and analytics are the two sections with the highest percentages.

Figure 5: Gender with Department and Promotion



3.4. Machine Learning Algorithms

Machine learning is a branch of computer science with significant links to statistics and optimization. In this thesis, we use supervised learning approaches for binary classification. The experiment is carried out within the Scikit-learn library, and the code is written in Python using supervised algorithms.

3.4.1. XGBoost

XGBoost is a boosted tree approach based on the gradient boosting principle. It employs more precise approximations by applying second-order gradients and enhanced regularization. It is a fast approach based on parallel tree creation that is designed to be fault-resistant in a distributed situation (Jain and Nayyar, 2018).The classifier accepts data in the form of DMatrix (Saradhi and Palshikar, 2011). During the research, the following characteristics were investigated and incorporated(9):

$$\mathrm{Obj}(\theta) = \sum_{i=1}^{n} L(y_i, \widehat{y_\iota}) + \sum_{k=1}^{K} \Omega(f_k) \qquad (9)$$

where:

- θ represents the parameters of the model.
- L is the loss function (e.g., mean squared error for regression, log loss for classification).
- y_i is the true value of the i-th instance.
- $\widehat{y}_i$ is the predicted value of the i-th instance.
- $\Omega(f_k)$ is the regularization term for the k-th tree.
- n is the number of instances.
- K is the number of trees

Regularization: This is the primary advantage of XGBoost. It also aids in reducing overfitting. This technique is used in linear and tree-based models to prevent overfitting (Aarshay, 2020).

Parallel Processing: XGBoost uses parallel processing and is much quicker than GBM. XGBoost now supports the Hadoop implementation (Aarshay, 2020).

High Flexibility: Custom optimization targets and assessment criteria can be defined by users in XGBoost. This gives the model a new dimension, and there are no restrictions on what we may accomplish (Aarshay, 2020).

XGBoost has a procedure for dealing with missing values. The user must offer a value that differs from the other observations and pass it as a parameter. It tries different things and learns which path to follow for missing values in the future (Aarshay, 2020).

Tree Pruning: When a GBM encounters a negative loss in the split, it will cease dividing the node. On the other hand, XGBoost divides up to the max depth set before pruning the tree backwards and removing splits beyond which no positive benefit is obtained (Aarshay, 2020).

Cross-validation is supported by XGBoost at each iteration of the boosting process, making it straightforward to acquire the precise optimal number of boosting rounds in a single run. In contrast to GBM, we must execute a grid search, and only a restricted number of parameters may be examined (Aarshay, 2020).

Sklearn allows users to begin training an XGBoost model from the previous run's last iteration - this can be a substantial benefit in some situations. This capability is also available in the GBM implementation of sklearn, so they are on the same page (Aarshay, 2020).

3.4.2. Random Forest (RF)

Random Forest is a classifier that uses several decision trees on different subgroups of a given dataset and averages them to enhance the predicted accuracy. The larger the number of trees in the forest, the higher the accuracy and the lower the risk of overfitting (Jaiswal, 2022).

where p_i is the proportion of instances of class i at node t.

$$\text{Gini}(\boldsymbol{t}) = 1 - \sum_{\boldsymbol{i=1}}^{c} P_i^2 \qquad (10)$$

3.4.3. Decision Tree (DT)

A decision tree chart may help us examine alternatives and their outcomes before committing to a solution. It provides a stylized universe where we may play out a sequence of actions and see where they go without devoting unnecessary real-world time and resources (Jaiswal, 2022).

where p_i is the proportion of instances of class i at node t.

$$\text{Gini}(\boldsymbol{t}) = 1 - \sum_{\boldsymbol{i=1}}^{c} P_i^2 \qquad (11)$$

3.4.4. Logistic Regression (LR)

Logistic Regression is an important machine learning technique because it can offer probabilities and classify new data using continuous and discrete datasets. It seeks to calculate the likelihood that the output variable belongs to a certain class. Logistic Regression may be used to categorize observations using many forms of data and can quickly discover the most efficient factors for classification (Jaiswal, 2022).

The sigmoid function transforms the linear combination of inputs into a probability value between 0 and 1.

$$\boldsymbol{\sigma}(\boldsymbol{z}) = \frac{1}{1 + e^{-z}} \qquad (12)$$

where:

- σ(z) is the sigmoid function.
- z is the linear combination of inputs.

3.4.5. AdaBoost

AdaBoost is a classifier that uses ensemble boosting to improve classifier accuracy. The AdaBoost classifier creates a strong classifier by merging numerous low-performing classifiers. AdaBoost's main assumption is to build classifier weights and train the data sample in each iteration (Navlani, 2022).

$$W_i(1) = \frac{1}{m} \qquad (13)$$

where:

- $w_i^{(1)}$ is the initial weight for the i-th instance.
- m is the total number of training instances.

3.4.6. Gradient Boosting

One of the most successful machine learning algorithms is the gradient boosting strategy. Gradient Boosting's distinguishing feature is that it instead fits a new predictor to the residual errors created by the preceding prediction (Tarbani, 2021), rather than fitting an algorithm to the data at each iteration.

$$\boldsymbol{F}(\boldsymbol{x}) = \text{arg}min_{\gamma} \sum_{i=1}^{m} L(y_i, \gamma) \qquad (14)$$

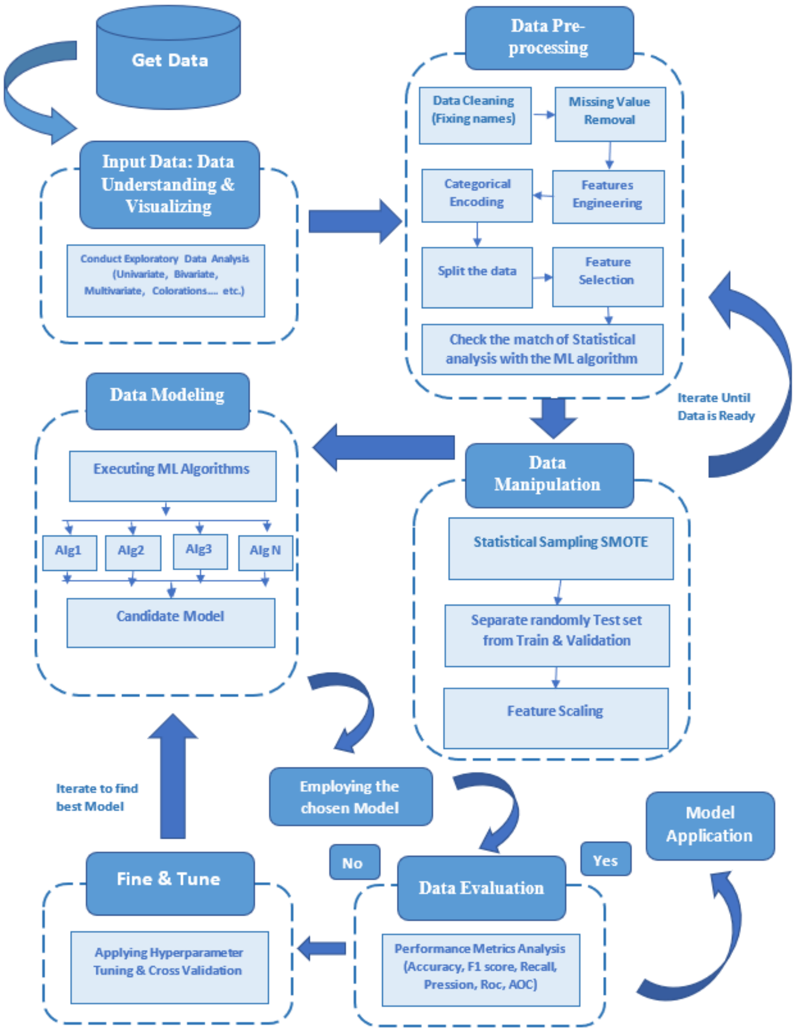
where:

- $F_0(x)$ is the initial prediction.
- $L(y_i,\gamma)$ is the loss function for the iii-th instance.
- y_i is the true label of the i-th instance.

3.5. Methodology

Our methodology is mainly composed of five phases: Input Data (which includes Data Understanding & Visualizing), Data Pre-processing (which provides for Data Cleaning, Data Preparing, & Data Splitting), Data Manipulation (which provides for Data Preprocessing and Manipulation), Data Modeling, and Data Evaluation (Fine & Tune). **Hata! Başvuru kaynağı b ulunamadı.** depicts a high-level overview. The following sections go through the specifics of each step. At the outset, the dataset will be described.

Figure 6: The general structure of the proposed employee promotion prediction framework


Get Data
Input Data: Data Understanding & Visualizing
Conduct Exploratory Data Analysis (Univariate, Bivariate, Multivariate, Colorations.... etc.)
Data Pre-processing
Data Cleaning (Fixing names)
Missing Value Removal
Categorical Encoding
Features Engineering
Split the data
Feature Selection
Check the match of Statistical analysis with the ML algorithm
Iterate Until Data is Ready
Data Modeling
Executing ML Algorithms
Alg1
Alg2
Alg3
Alg N
Candidate Model
Data Manipulation
Statistical Sampling SMOTE
Separate randomly Test set from Train & Validation
Feature Scaling
Iterate to find best Model
Employing the chosen Model
Model Application
No
Yes
Fine & Tune
Applying Hyperparameter Tuning & Cross Validation
Data Evaluation
Performance Metrics Analysis (Accuracy, F1 score, Recall, Pression, Roc, AOC)

To give some brief explanations about the implementation of our study, algorithms are used through the Scikit-learn library, and the experiment is carried out within Python. These phases are described as follows:

Input Data: Data Understanding & Visualizing

We use publicly available data provided by Analytics Vidhya Data Analysis. This dataset has 14 characteristics 54808 records for train data and 23490 for test data. We will choose the relevant aspects and add new ones that impact the employees' promotion as an important indication of the promotion process. More details of the dataset were explained in the Background section.

Data Pre-processing: Data Cleaning, Data Preparing & Data Splitting

The data preparation stage is critical for our investigation to acquire clean and usable data. There were instances in the raw data that were not appropriate. This was due to mistakes and abnormalities that had to be removed. Data cleaning and filling-in of missing values in the dataset were conducted. Analysis of the dataset is critical for our investigation to acquire clean and usable data. Treating missing values is important in any machine learning model's creation. Various reasons, such as incomplete forms, unavailable values, data entry errors, and data loss, can cause missing values. We do not have to delete any missing values; we can impute the values using mean, median, and mode.

Extracting features from raw data using domain expertise and data mining tools is known as "feature engineering." Feature engineering may be thought of as applied machine learning. Many in the industry believe it to be the most crucial step in improving model performance. It involves removing unnecessary columns, binning the numerical and categorical features, and aggregating some features.

***Agg*regating Multiple Features:**

The variables need to be categorized so that the impact of making groupings can be seen more clearly. Many of the variables are either continuous in nature or have many discrete values that peak at specific places. New features—such as Metric of sum, Total score, Work fraction, Work start year, Years remaining to retire, and Performance—are calculated because of the following assumptions:

The metric of sum: this feature is the sum of awards won, KPIs met, and the previous year's rating.

Total score: training and workshops are essential for employees since they are conducted to help employees grow their skills. These training ratings assist the organization in determining whether staff are progressing. The columns "number of trainings" and "average training score" describe the number of company-organized workshops and training the employee attended and the average training score. The total score field is numeric and on the ratio scale. The goal is to see whether there are any correlations between the total score and the "is promoted" column. The total score column is separated into three bins (categories) for this purpose: Low (65 or below), Mediocre (65 to 145 points), and High (145 or higher).

Work fraction: this new feature was created to represent the fraction of work done with their age.

Work start year: this was another feature that represents the starting age of the employee.

Years remaining to retire: this is another new feature representing the remaining years for the employee until retirement.

Performance: For ease of analysis, KPIs_met and awards_won are combined into a single-column performance using the any() function. Any employee who has either won an award or has met KPIs has shown good performance. Most employees who were promoted demonstrated excellent performance, but those who were not promoted had a high rate of non-performance. Many employees who have worked hard yet have not been promoted. This might be related to a variety of different factors and provides a solid reason to investigate the other aspects as well.

Binning the Numerical and Categorical Features:

To perform well in this phase, we combine the levels of 'no_of_trainings', which has fewer observations in train and test data. For the age feature, we bin ‘age’ data into groups (every five years as a bin).

Removing Unnecessary Features:

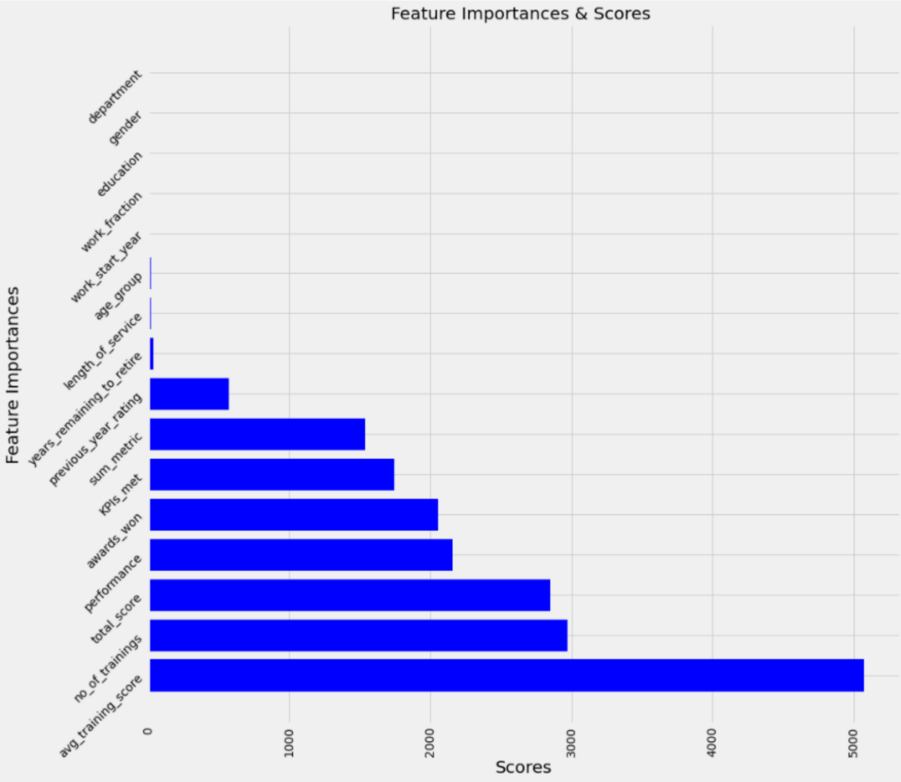
In the next stage of training, we will delete some of the columns that are not relevant for forecasting promotions. In addition, we will encrypt our object data and convert it to numeric form for the machine learning model to accept it. Categorical variables are well-known for hiding and masking important information in data collection. It is critical to understand how to cope in such situations. Otherwise, we lose out on discovering the most essential variables in a model. The train-test split is a method of assessing the performance of a machine learning system. It may be used for supervised learning and classification or regression tasks.

By eliminating the target column from the data, we store the target variable in y and the remainder of the columns in x. The Label Encoder will then encode the Department, Gender, and Number of Training Columns.

When creating a predictive model, the first and most critical phases in constructing our model should be feature selection and data cleansing. To conduct research, the dataset must contain many characteristics that impact employee advancement directly or indirectly. Irrelevant attributes in our data can reduce model accuracy and cause our model to train based on irrelevant information.

Machine learning feature selection strategies may be roughly categorized into the following. Wrapper, filter, and intrinsic supervised techniques are the three types of supervised methods. Using the model's feature importance attribute, we can determine the feature importance of each feature in our dataset. We will use SelectKBest to extract the top features from the dataset.

Figure 7: Feature importance and scores



We next wrap the model in a SelectFromModel instance using the feature importance derived from our training dataset. This is used to pick features on our training dataset, train a model using the XGBoost classifier using the selected subset of features, and then assess the model on the test set using the same feature selection strategy. We may test several thresholds for picking features based on feature relevance for interest. The feature importance of each input variable allows us to rank each subset of features in order of significance, starting with all features and ending with the most significant feature (Brownlee, 2022). According to the results of this procedure, 'department,' 'education,' and 'gender' are omitted. These are necessarily the least important features for promotion prediction in our model. From the above conclusions, it can be stated with confidence that no factor alone is responsible for the promotion of an employee.

Table 2: Factors considered for predictive modeling

Factors considered for predictive modeling
no_of_trainings
previous_year_rating
length_of_service'
KPIs_met
awards_won
avg_training_scor

sum_metric
total_score
work_fraction
work_start_year
years_remaining_to_retire
Performance
age_group

Data Manipulation: Preprocessing and Manipulate Data

Machine learning algorithms perform best when the number of samples in each class is about equal. This is because most algorithms are intended to enhance accuracy while minimizing mistakes. A class imbalance develops when observation in one class exceeds observation in other courses. They are often divided into two classes: the majority (negative) class and the minority (positive) class.

Resampling is a nonparametric statistical inference approach. Several statistical methods are available for resampling data, including oversampling and undersampling. The Synthetic Minority Oversampling Technique (SMOTE) oversamples the minority class by manufacturing "synthetic" cases rather than oversampling using replacement. After balancing the data, we are separating/splitting the entire dataset into training and testing data. Feature scaling is a strategy for lowering the values of all the independent characteristics of the dataset on the same scale. It is also known as data normalization and is performed during the data preparation step (SagarDhandare, 2022).

There must be a clear rule to determine when to normalize or standardize our data. We, therefore, have decent performance using standardization rather than normalization. It is best to fit the scaler to the training data before using it to change the testing data. This prevents data leaks during the model testing procedure.

Data Modeling

Classification has two separate implications in machine learning. Multiple predictive models such as XGBoost, Random Forest, Decision Tree, Logistic Regression, AdaBoost, and Gradient Boosting were applied in this scenario. The objective is to find the best classifier for the problem under consideration. Each classifier must be trained on the feature set, and the classifier with the best classification results is used to forecast. The original dataset was partitioned into two portions with an 80:20 ratio—one for training and another for testing.

Data Evaluation (fine & tune)

The data mentioned above set includes features such as "performance", "no_of_training", "previous_year_rating", and so on. Based on these values, the learning algorithm will anticipate whether the employee will be promoted to the organization. A typical confusion matrix may determine the number of cases properly categorized by a model. The confusion matrix visualizes a classifier's performance, providing a complete analysis with data on the number of true positives, false positives, true negatives, and false negatives. A classification report would show the model's accuracy, recall, Roc, AOC, and F1-Score. Precision and recall are based on the measure of relevance, with precision being the proportion of relevant samples found among the retrieved samples. When the datasets are divided, it is critical to maintain the same distribution of target variables across both the training and test datasets.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \qquad (15)$$

where:

- TP = True Positives
- TN = True Negatives
- FP = False Positives
- FN = False Negatives

$$\text{Precision} = \frac{TP}{TP + FP} \tag{16}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{17}$$

$$\text{F1} - \text{Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{18}$$

The goal "is_promoted" attribute is a binary variable with 91% "No" and 9% "Yes", with both datasets retaining the same percentage after splitting. The classifiers are assessed using the evaluation criteria in order to choose the best model for the issue (Jain, Jain and Pamula, 2020).

Cross-validation is a strategy for preventing over-fitting and simplifying the model. This research utilized a five-fold CV. The training-set was randomly divided into five parts (k), with one serving as a validation-set and the other k-1s serving as training-sets. Each iteration used a different section as the validation set, and the average prediction error was calculated by averaging the average errors from both sets.

XGBoost is a powerful machine learning method, particularly in terms of speed and accuracy. A grid search must be performed for all relevant model parameters before hyper-parameter adjustment is required. There are numerous settings, especially in the case of XGBoast, and it may be CPU-expensive at times (Aarshay, 2020). XGBoost settings have been classified into three groups by the creators of XGBoost (Aarshay, 2020):

General parameters: Direct the overall operation.

Booster parameters: Direct the specific booster (tree/regression) at each stage.

Learning Task Parameters: Direct the optimization process.

The next step is using a general approach for parameter tuning. The various steps to be performed are:

Selecting a fast-learning rate: Generally, a learning rate of 0.1 is adequate, although values ranging from 0.05 to 0.3 should suffice for various problems. Determine the best number of trees to use for this learning rate. XGBoost has a very handy function called "cv" that does cross-validation at each boosting iteration, delivering the optimum number of trees needed. In our study, the learning_rate is fixed to 0.1 and cv to 5.

Tree-specific parameters (max depth, min child weight, gamma, subsample, colsample by tree) should be fine-tuned for the chosen learning rate and number of trees. Here, after many iterations and tuning with changing in different values and looking at the performance, we finally fixed these values: max_depth=4, min_child_weight=6, gamma=0.1, subsample=0.8, colsample_bytree=0.8.

Regularization settings (lambda, reg_alpha=0.01) for XGBoost can be adjusted to minimize model complexity and improve performance.

'scale pos weight' is one of the most critical factors people frequently overlook when dealing with an unbalanced dataset. This parameter should be fine-tuned with caution since it may result in overfitting the data.

4. RESULTS AND DISCUSSIONS

This phase assessed the suitability of the models used. But first and foremost, we had to select the appropriate variables for our work. Hence, as we mentioned earlier in section 3 about the importance of choosing the feature, we will display the results and emphasize using the XGBoost classifier in selecting those features because we deemed it more important. In our scenario, we train and test an XGBoost model on the whole training and test datasets.

Table 3. Evaluation procedure with several selected features

Threshold	Features Number	Accuracy
0.414	n =1	77.28%
0.150	n =2	77.28%
0.073	n =3	77.63%
0.073	n =4	78.66%
0.070	n =5	80.11%
0.040	n=6	80.08%
0.037	n=7	83.68%
0.026	n=8	84.04%
0.024	n=9	83.87%
0.018	n=10	83.92%
0.018	n=11	84.10%
0.014	n=12	83.93%
0.013	n=13	83.96%
0.012	n=14	84.16%
0.012	n =15	84.19%
0.006	n =16	84.21%

Table 3 demonstrates that the model's performance typically improves as the number of selected characteristics increases, starting with feature number seven, which has an accuracy of 83.68%. This problem has a trade-off between features and test set accuracy. We could decide to take a complex model (larger attributes such as n = 13) and accept a modest decrease in estimated accuracy from 84.21 percent to 83.96%, which is likely to be more useful based on the importance of the variables used, and, of course, the accuracy will improve more using grid search as the model evaluation scheme.

Following that, after selecting the 13 features to be used in the model, but this is insufficient for us, the next phase is to run the model and compare the results without the features that correlate, such as length of service and age being highly associated, as we discovered earlier in section 3. It can also be noticed that KPIs and previous year's ratings are correlated to some extent, signaling some linkage. Thus, we eliminate those two characteristics of age and KPIs to avoid multicollinearity. As a result, the following ten characteristics will be used in running the six models: 'the number of trainings', 'previous year rating', 'length of service', 'awards won', 'avg training score', 'sum metric', 'total score', 'work fraction', 'work start year', and 'years remaining.

Comparing the results, clearly seen that having high accuracy with the 13 features rather than ten features. Therefore, we decided to choose 13 features to be employed in the model. Hence, the outcomes of the prediction phase judgments were initially gathered in the relative "confusion matrix," without using a grid search and then a grid search for each method. This is a matrix in which the classifier's predicted values are given in the columns, and the actual values of each instance of the test set are shown in the rows. To start with the performance evaluation, we used the confusion matrix to generate a set of essential metrics to quantify the efficiency of each algorithm: accuracy, precision, recall, and F1-score.

Table ***4*** summarizes these measures, which are based on the number of mistakes and accurate responses generated by the classifier.

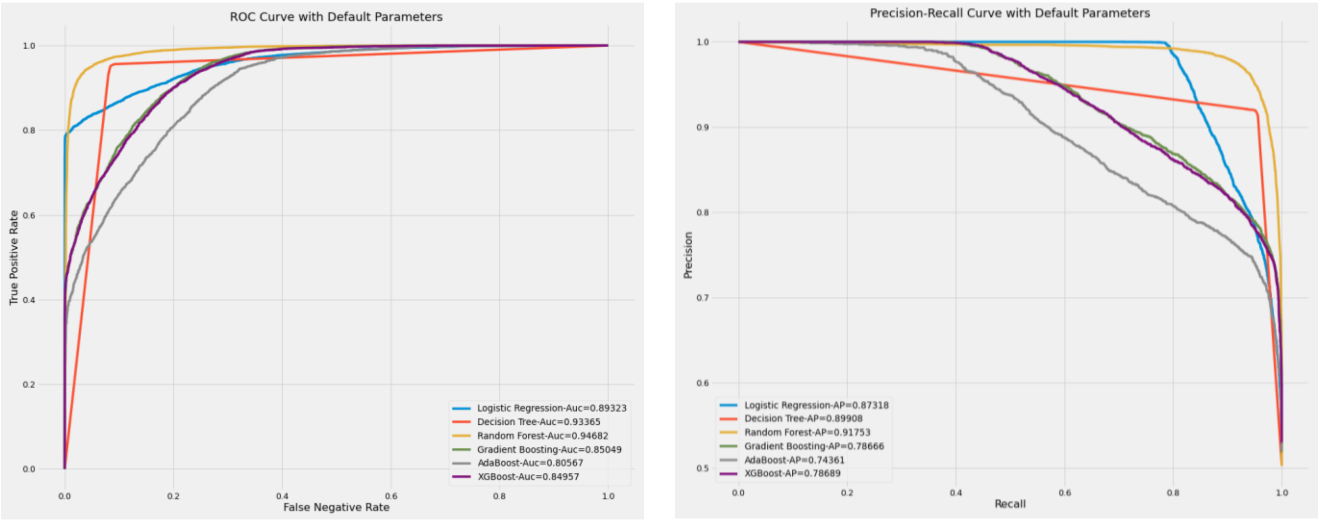
Table 4. Evaluation metrics with default parameters of different classifiers

Classifiers	Accuracy	Precision	Recall	F1-score	ROC AUC
Logistic Regression	0.88%	0.88%	0.88%	0.87%	0.87%
Decision Tree	0.92%	0.91%	0.91%	0.91%	0.91%
Random Forest	0.93%	0.92%	0.92%	0.92%	0.92%
AdaBoost	0.80%	0.79%	0.79%	0.79%	0.79%
Gradient Boosting	0.82%	0.82%	0.82%	0.82%	0.82%
XGBoost	0.82%	0.77%	0.77%	0.77%	0.77%

The results of this experiment revealed that all the classifiers had acceptable accuracy, which is greater than 80%. In many situations, this level of accuracy is seen as adequate. The dataset yielded acceptable models for this experiment's specified classification techniques. The model's accuracy is used to select the most acceptable classifier for the dataset to choose the appropriate classifier. As demonstrated in

Table ***4***, the Random Forest classifier has the best accuracy, with 93%, the highest among the chosen classifiers. Similarly, compared to other classifiers, Random Forest scores well on metrics such as precision or recall, F-Measure, and ROC AUC. However, the AdaBoost Classifier model is less accurate than others, with just over 80% accuracy. With 92% accuracy, the decision tree also performed well. **Hata! Başvuru kaynağı bulunamadı.** shows the same conclusion in terms of the R OC AUC graph. According to **Hata! Başvuru kaynağı bulunamadı.**, the random forest has the highest average precision, i.e., true positive rate.

Figure 8: Default parameters of different classifiers for a) ROC b) Precision-Recall Curve



a) b)

The accuracy of DT and RF is substantially higher, and these classifiers may be used to forecast whether an individual will be promoted within the organization. However, in our research, we will focus more on the XGBoost classifier because it is the uniqueness of our study and the first time using this classifier to forecast such a problem; thus, we will use grid search to improve the accuracy of XGBoost beyond 82%.

Considering model hyper-parameters impact performance, we alter the parameters of five models using grid search and cross-validation, with a heavy emphasis on the XGBoost classifier. The basic concept behind this approach is to select numerous parameter combinations in advance and run cross-validation for each set of parameters to discover the ideal parameter combination for XGBoost using five-fold cross-validation.

Classifiers	Accuracy	Precision	Recall	F1-score	ROC AUC
Logistic Regression	0.88%	0.883%	0.8832%	0.88%	0.8835%
Decision Tree	0.92%	0.91%	0.91%	0.91%	0.91%
Random Forest	0.93%	0.933%	0.9328%	0.93%	0.9327%
Gradient Boosting	0.86%	0.858%	0.8582%	0.86%	0.8582%
XGBoost	0.94%	0.94%	0.94%	0.9398%	0.9397%

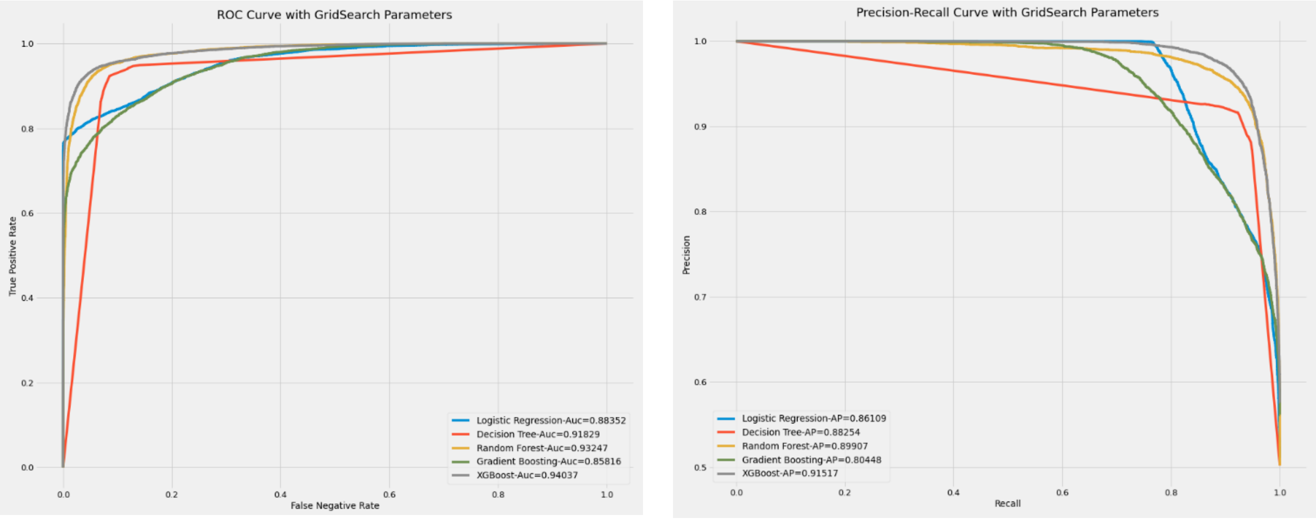
and

Table 5**Hata! Başvuru kaynağı bulunamadı.Hata! Başvuru kaynağı bulunamadı.** show that the XGBoost model outperforms other models in terms of decile performance. It also consistently outperforms a random estimate, with XGBoost outperforming Random Forest. In terms of accuracy, memory consumption, and time-consuming, the XGBoost classifier surpasses the other classifiers.

Table 5. Evaluation metrics with GridSearch parameters of different classifiers

Classifiers	Accuracy	Precision	Recall	F1-score	ROC AUC
Logistic Regression	0.88%	0.883%	0.8832%	0.88%	0.8835%
Decision Tree	0.92%	0.91%	0.91%	0.91%	0.91%
Random Forest	0.93%	0.933%	0.9328%	0.93%	0.9327%
Gradient Boosting	0.86%	0.858%	0.8582%	0.86%	0.8582%
XGBoost	0.94%	0.94%	0.94%	0.9398%	0.9397%

Figure 9: Result with GridSerach parameters of different classifiers for a) ROC b) Precision-Recall Curve

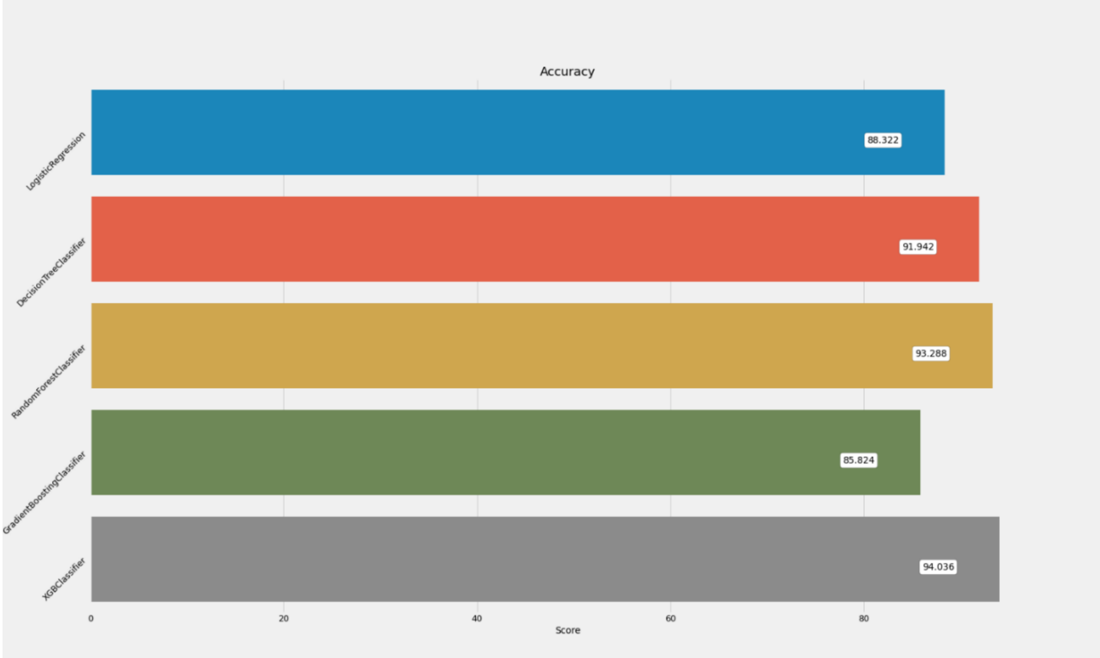


a) b)

As the results show, while random forests rely on their randomization steps to help them achieve higher generalization, more is needed to prevent over-fitting in this scenario. On the other hand, XGBoost attempts to create new trees that complement the existing ones. Boosting improves training for difficult-to-classify data points. Another significant thing to consider is the over-fitting experienced by classifiers other than XGBoost, notwithstanding regularization or the addition of randomness. Because of its superior intrinsic regularization, XGBoost solves this issue and works wonderfully in our scenario.

The XGBoost classifier is also designed to be fault-tolerant in a distributed setting and is optimized for fast, parallel tree construction. The XGBoost classifier accepts DMatrix data. DMatrix is an XGBoost internal data structure designed for memory economy and training speed. DMatrixes were created here by combining numpy arrays containing features and classes. Due to those reasons, the XGBoost classifier was selected as the best classifier for the dataset.

Figure 10: Final accuracy of the classifiers



Furthermore, XGBoost surpasses the competition, and its time consumption is reasonable. As a result, the XGBoost classifier, based on 13 features, is chosen as the final prediction model, with accuracy and AUC of 94.036%, a recall of 94%, and a precision of 94%. It outperformed the baseline model in terms of accuracy, increasing it by up to 94%, as shown in

Classifiers	Accuracy	Precision	Recall	F1-score	ROC AUC
Logistic Regression	0.88%	0.883%	0.8832%	0.88%	0.8835%
Decision Tree	0.92%	0.91%	0.91%	0.91%	0.91%
Random Forest	0.93%	0.933%	0.9328%	0.93%	0.9327%
Gradient Boosting	0.86%	0.858%	0.8582%	0.86%	0.8582%
XGBoost	0.94%	0.94%	0.94%	0.9398%	0.9397%

.

In a comparable study conducted by Analytics Vydha, the participant rated first (faizankshaikh, 2022) received an accuracy of 92.88%, a recall of 42.13%, and a precision of 63.13% for promotion prediction, with a synthesized value (F1 score) of roughly 50.54% (faizankshaikh, 2022) whereas that of our experiment was 93.98%. Similarly, in the realm of employee turnover, papers (Punnoose and Ajit, 2016; Jain and Nayyar, 2018; Yedida et al., 2018) used the XGboost classifier to obtain accuracy values of 88%, 89%, and 90%, respectively. As a result, the suggested framework and technique perform well and are equal to the other published methods.

5. CONCLUSION

In this thesis, we attempt to forecast whether an employee will be promoted at their present company. To solve this categorization problem, we employ supervised machine learning methods.

Our approach's primary contributions are the application of machine learning algorithms and a framework for forecasting employee promotion.

XGBoost is recognized as a superior algorithm in terms of memory use efficiency, accuracy, and running time, with an accuracy and ROC AUC of 0.94036%, a recall of 0.94%, and a precision of 0.94%. It is a robust and scalable approach for handling all types of noise from large data sets. The suggested automated predictor's results show that the important promotion factors are average training score, number of training, total score, and performance.

The data analysis results constitute a beginning point for creating increasingly efficient employee promotion classifiers. Using extra datasets or simply updating them regularly, feature engineering to uncover new relevant qualities in the dataset and the availability of more information. Management may use this project to estimate the likelihood of promotion, allowing managers to choose the best conditions for someone to be promoted.

We want to deploy the suggested model in real-world firms soon so that organizations may learn about employee promotion variables. The study can be expanded by integrating features such as Scope of Development, Views on Workload Distribution, Career Goal Discussion, and Issues of Unhealthy Work Ethics. For example, we can estimate promotion speed or investigate whether a promoted person is qualified for a higher-level role and then provide appropriate management recommendations. It is advised to examine the use of deep learning models for forecasting promotion. A well-designed network with enough hidden layers may enhance accuracy, but scalability and practical implementation must also be considered. Instead of eliminating the region column, we may divide the 32 columns into two sections. We may also eliminate the gender column because there is only a small difference in the likelihood of males and females receiving a promotion..

REFERENCES

Aarshay (2020). XGBOOST parameters: XGBoost parameter tuning. Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/. (Accessed April 11, 2022).

Bandyopadhyay, N. and Jadhav, A. (2021) 'Churn Prediction of Employees Using Machine Learning Techniques.', Technical Journal / Tehnicki Glasnik, 15(1), pp. 51–59. Available at: http://icproxy.khas.edu.tr/login?url=https://search.ebscohost.com/login.aspx?direct=true&db=edb&AN=149158643&site=eds-live.

Brownlee, J. (2022). Feature Importance and Feature Selection With XGBoost in Python. [online] Machine Learning Mastery. Available at: https://machinelearningmastery.com/feature-importance-and-feature-selection-with-xgboost-in-python/ (Accessed 14 April 2022).

Chen, K.-Y., Hsu, Y.-L. and Wu, C.-C. (2012) Num 2 Fall 2012 1 The International Journal Of Organizational Innovation Volume 5 Number 2 Fall 2012 Information Regarding The International Journal Of Organizational Innovation 4 IJOI, The International Journal of Organizational Innovation. Available at: http://www.iaoiusa.org (Accessed: 1 March 2022).

Chen, K.-Y., Hsu, Y.-L. and Wu, C.-C. (2012) Num 2 Fall 2012 1 The International Journal Of Organizational Innovation Volume 5 Number 2 Fall 2012 Information Regarding The International Journal Of Organizational Innovation 4 IJOI, The International Journal of Organizational Innovation. Available at: http://www.iaoiusa.org (Accessed: 1 March 2022).

De Pater, I. E. et al. (2009) 'Employees' Challenging Job Experiences And Supervisors' Evaluations Of Promotability', Personnel Psychology, 62(2), pp. 297–325. doi: 10.1111/j.1744-6570.2009.01139.x.

Faizankshaikh (2022). wns-analytics-wizard-2018/Rank 1: Siddharth3977 at master · analyticsvidhya/wns-analytics-wizard-2018. [online] GitHub. Available at:

https://github.com/analyticsvidhya/wns-analytics-wizard-2018/tree/master/Rank%201:%20Siddharth3977 (Accessed 13 March 2022).

Febrina, S. C. (2017) 'Predicting Employee Performance by Leadership, Job Promotion, and Job Environmental in Banking Industry', Jurnal Keuangan dan Perbankan, 21(4), pp. 641–649. doi: 10.26905/jkdp.v21i4.1630.

Jain, P. K., Jain, M. and Pamula, R. (2020) 'Explaining and predicting employees' attrition: a machine learning approach', SN Applied Sciences, 2(4). doi: 10.1007/s42452-020-2519-4.

Jain, R. and Nayyar, A. (2018) 'Predicting employee attrition using xgboost machine learning approach', in Proceedings of the 2018 International Conference on System Modeling and Advancement in Research Trends, SMART 2018. (1)Department of Computer Science and Engineering (CSE), Bharati Vidyapeeth's College of Engineering: Institute of Electrical and Electronics Engineers Inc., pp. 113–120. doi: 10.1109/SYSMART.2018.8746940.

Jaiswal, Logistic regression in machine learning - javatpoint. www.javatpoint.com. Available at: https://www.javatpoint.com/logistic-regression-in-machine-learning (Accessed April 11, 2022).

Jaiswal, Machine learning decision tree classification algorithm - javatpoint. www.javatpoint.com. Available at: https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm (Accessed April 11, 2022).

Jaiswal, Machine learning random forest algorithm - javatpoint. www.javatpoint.com. Available at: https://www.javatpoint.com/machine-learning-random-forest-algorithm (Accessed April 11, 2022).

Jantan, H. and Hamdan, A. (2010) 'Applying Data Mining Classification Techniques for Employee's Performance Prediction', Knowledge …, pp. 601–607. Available at: http://www.kmice.cms.net.my/ProcKMICe/KMICe2010/Paper/PG601-607.pdf (Accessed: 29 November 2021).

Li, M. G. T. et al. (2021) 'Employee performance prediction using different supervised classifiers', in Proceedings of the International Conference on Industrial Engineering and Operations Management, pp. 6870–6876.

Liu, J. et al. (2019) 'A data-driven analysis of employee promotion: The role of the position of organization', in Conference Proceedings - IEEE International Conference on Systems, Man and Cybernetics. National University of Defense Technology, College of Systems Engineering: Institute of Electrical and Electronics Engineers Inc., pp. 4056–4062. doi: 10.1109/SMC.2019.8914449.

Long, Y. et al. (2018) 'Prediction of employee promotion based on personal basic features and post features', in ACM International Conference Proceeding Series. Association for Computing Machinery, pp. 5–10. doi: 10.1145/3224207.3224210.

Machado, C. S. and Portela, M. (2021) 'Age and Opportunities for Promotion', SSRN Electronic Journal. doi: 10.2139/ssrn.2367639.

Najafi-Zangeneh, S. et al. (2021) 'An improved machine learning-based employees attrition prediction framework with emphasis on feature selection', Mathematics, 9(11). doi: 10.3390/math9111226.

Navlani, A., AdaBoost classifier algorithms using python Sklearn tutorial. DataCamp. Available at: https://www.datacamp.com/tutorial/adaboost-classifier-python (Accessed April 11, 2022).

Punnoose, R. and Ajit, P. (2016) 'Prediction of Employee Turnover in Organizations using Machine Learning Algorithms', International Journal of Advanced Research in Artificial Intelligence, 5(9). doi: 10.14569/ijarai.2016.050904.

SagarDhandare (2022). Feature Scaling In Data Science!. [online] Medium. Available at: https://medium.datadriveninvestor.com/feature-scaling-in-data-science-5b1e82492727 (Accessed 13 April 2022).

Saradhi, V. V. and Palshikar, G. K. (2011) 'Employee churn prediction', Expert Systems with Applications, 38(3), pp. 1999–2006. doi: 10.1016/j.eswa.2010.07.134.

Sarker, A. et al. (2018) Employee's Performance Analysis and Prediction using K-Means Clustering & Decision Tree Algorithm Mawlana Bhashani Science and Technology University Employee's Performance Analysis and Prediction using K-Means Clustering & Decision Tree Algorithm, Type: Double Blind Peer Reviewed International Research Journal Software & Data Engineering Global Journal of Computer Science and Technology: C.

Tarbani (2021). Gradient boosting algorithm: How gradient boosting algorithm works. Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2021/04/how-the-gradient-boosting-algorithm-works/. (Accessed April 11, 2022).

Yedida, R. et al. (2018) 'Employee Attrition Prediction', IJISET-International Journal of Innovative Science, Engineering & Technology, 7(9). Available at: www.ijiset.com (Accessed: 13 January 2022).