information

MDPI

Article

Leakage-Free Evaluation for Employee Attrition Prediction on Tabular Data

Ana Maria Căvescu and Alina Nirvana Popescu *

Computer Science and Engineering Department, National University of Science and Technology, Politehnica, 060042 Bucharest, Romania; ana.prisecaru@gmail.com
* Correspondence: nirvana.popescu@upb.ro

Abstract

In the context of employee attrition prediction using imbalanced tabular data, we propose a reproducible, leakage-aware evaluation protocol and validate it on the IBM HR Attrition dataset. We perform the train/test split prior to any rebalancing; SMOTE (Synthetic Minority Over-sampling Technique) is applied exclusively within the training portion of each fold in stratified 5-fold cross-validation, while the test set remains untouched. One-Hot Encoding is performed consistently using pd.get_dummies. We benchmark Logistic Regression, Random Forest, ExtraTrees, LightGBM, and XGBoost using imbalance-aware metrics: F1 for the minority class, PR-AUC reported as Average Precision (AP), and ROC-AUC reported both in cross-validation and on the held-out test set. XGBoost attains the best mean AP in cross-validation (0.556 ± 0.056). Logistic Regression achieves the highest mean F1 (0.439 ± 0.048), while LightGBM yields the best mean ROC-AUC (0.791 ± 0.026). On the test set, XGBoost achieves a precision value of 0.65 and a recall value of 0.45 at a fixed threshold of 0.5. Overall, the results highlight a trade-off between stable minority-class detection (Logistic Regression) and stronger risk ranking performance (boosting models) under class imbalance.

Keywords: employee retention; machine learning; attrition prediction; random forest; LightGBM; ExtraTree; XGBoost; logistic regression; HR analytics

1. Introduction

In an increasingly competitive and highly mobile labor market, the retention of valuable employees has become a priority for organizational stability and performance. Machine learning approaches can anticipate the risk of departure by leveraging historical data and organizational indicators, supporting proactive HR decisions. The objective of this paper is to comparatively evaluate several ML algorithms for predicting turnover and extract the relevant factors for retention decisions.

Staff turnover is a major problem in modern organizations, with negative effects such as high costs, the disappearance of accumulated organizational expertise, and a decrease in team morale. This context highlights the need for robust predictive methods, as sectors with high turnover can directly benefit from machine learning solutions for early identification of the risk of departure and prioritization of retention interventions. To combat these effects, the integration of machine learning technologies into workforce management becomes a promising solution. By analyzing historical data, algorithms can identify patterns related to job satisfaction, career opportunities, or interpersonal relationships, thus anticipating the risk of employee departure [1].

Check for updates

Academic Editor: Haridimos Kondylakis

Received: 25 January 2026
Revised: 13 March 2026
Accepted: 18 March 2026
Published: 23 March 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

The objective of this paper is the comparative evaluation of five ML algorithms on the IBM HR Attrition dataset for predicting turnover and identifying relevant explanatory factors for retention decisions.

The contribution of our paper in this field is as follows:

- Leakage-Aware Protocol: We perform the train/test split before any rebalancing and apply SMOTE only within the training portion of each fold during stratified 5-fold cross-validation.
- Reproducible Benchmarking: We compare five commonly used models (Logistic Regression, Random Forest, ExtraTrees, LightGBM, and XGBoost) under the same protocol and seed, reporting F1 (Attrition = 1), PR-AUC as Average Precision (AP), and ROC-AUC in CV (mean $\pm$ std) and on a held-out test set.
- Interpretability: We provide model-specific feature analyses to highlight the predictive factors associated with attrition risk (non-causal).

2. The Analysis of Staff Turnover Prediction Techniques Using Artificial Intelligence

In recent years, current research has examined the advantages and limitations of using machine learning (ML) and deep learning (DL) technologies in human resource management to predict employee turnover. These techniques are widely used in various industries, including marketing, distribution, and customer service.

The use of ML algorithms, such as Random Forest, Logistic Regression, Decision Tree, and XGBoost, has been shown to be useful in predicting employee turnover intentions [2]. By applying the SMOTE technique to balance the data and evaluating the models using metrics such as precision, recall, and F1 score, this study highlights the effectiveness of these methods in employee relationship management.

The use of a Cross-Component Attention Transformer model has been proposed for predicting staff turnover, taking into account external factors such as the influence of competition and the tendency to imitate departures within the organization [3]. Applied mainly in the IT industry, the model has been evaluated and compared to traditional methods and demonstrated superior performance, providing an extended perspective on the variables that influence employees' decision to leave.

With the advent of large-scale language models (LLMs), such as GPT, research in the field of staff turnover prediction has undergone a significant transformation, being complemented by recent works that explore both the advantages and limitations of these advanced technologies in the analysis of human resources data.

Models developed using supervised learning algorithms such as AdaBoost, SVM, and Random Forest have demonstrated efficiency in establishing predictive analyses regarding staff turnover rates, contributing to the early identification of the risk of employee departure and the implementation of appropriate retention strategies [4].

3. Methodology

3.1. Data Processing

In this study, the analysis of employee turnover was performed by integrating machine learning methods applied to a set of public structured data obtained from an open source [5]. The dataset contains information on 1470 employees and 35 variables relevant to professional behavior, job satisfaction, and intention to leave the job.

The data analyzed include demographic factors such as age, education, gender, organizational indicators (e.g., department, position, seniority in the company, total seniority in the job, promotions, relationship with the direct manager), as well as elements related to work–life balance (e.g., distance from the workplace, frequency of travel, level of satisfac-

tion and involvement). Also included are variables that reflect the professional behavior of employees, such as participation in training sessions, time spent in the current position, and total seniority in the field of work.

In order to provide the most accurate picture possible of the employee turnover risk, the analysis integrated both quantitative variables (numerical and categorical) and qualitative indicators, derived from the measurement of satisfaction and personal balance. This methodological framework allows the investigation and prediction of the decision to leave by correlating personal, structural, and organizational factors.

In data processing, methods of coding categorical variables (One-Hot Encoding) and the preparation of a set of relevant features for modeling were applied. The resulting dataset thus provides a solid basis for the application of machine learning algorithms and for evaluating the predictive performance of the models used in the analysis of the risk of employee turnover.

Data leakage is a frequent source of overly optimistic performance estimates in applied machine learning, especially when preprocessing or resampling steps are fitted using information from validation or test data. Sasse et al. emphasize that leakage can occur at multiple points in an ML pipeline (e.g., feature engineering, scaling, imputation, or oversampling) when these steps are performed prior to the train/test split or outside the cross-validation training loop, leading to inflated metrics and reduced reproducibility. Therefore, in leakage-sensitive settings such as attrition prediction on imbalanced tabular data, evaluation protocols should ensure that any data-driven transformations and rebalancing procedures are applied strictly on training data only, while validation and test partitions remain untouched [6].

We analyzed the distribution of labels to see if we have an imbalance in the data regarding the number of employees leaving or staying. Thus, we have 1233 employees staying and 237 employees leaving. We observe an imbalance in the data; we have 5 times more employees staying than employees leaving. The data was processed by transforming the target column 'Attrition' into a binary variable (1 for 'Yes', 0 for 'No'). Columns such as "EmployeeNumber" and "StandardHours", considered unnecessary for the analysis, were removed from the dataset.

The AUC-ROC was calculated from the continuous scores of the model using predict_proba[:, 1], and it is not from binary labels. We report the AUC-ROC both in cross-validation (mean $\pm$ std on 5 folds) and on the test set (unsampled).

To address the imbalance of data in the emp_attrition.csv file, we used the Synthetic Minority Over-Sampling Technique (SMOTE—for each example in the minority class, SMOTE randomly selects a nearest neighbor using Euclidean distance and creates a new intermediate point) to balance the dataset. We did not combine it with class_weight = 'balanced' to avoid over-compensation.

Categorical variables were encoded by One-Hot Encoding using pd.get_dummies (drop_first = True). To prevent data leakage in the evaluation, SMOTE was applied exclusively on the training subset (within each cross-validation fold), and the test set remained untouched (unsampled). Since One-Hot Encoding does not estimate statistical parameters (unlike standardization), it typically introduces less direct leakage; however, for strict leakage control, encoding should be fitted on training data only (e.g., via a ColumnTransformer). Although SMOTE is applied after One-Hot Encoding in our current pipeline, we acknowledge that interpolation in a One-Hot space may produce semantically implausible synthetic samples. We did not apply SMOTENC in this study because it requires a mixed-type (numeric + categorical) representation with explicit categorical feature indices prior to One-Hot Encoding, whereas our implementation uses pd.get_dummies to obtain a unified high-dimensional numeric matrix for consistent benchmarking across models. Evaluating

categorical-aware oversampling (e.g., SMOTENC) and cost-sensitive alternatives under the same leakage-aware protocol is left for future work.

3.2. Experimental Setup

The implementation was done in Python 3.13.9, using scikit-learn, imbalanced-learn, XGBoost, and LightGBM. We set random_state = 42 for stochastic operations and used stratified cross-validation (k = 5, shuffle = True). SMOTE was applied exclusively on the training subsets of each fold (via pipeline) to avoid data leakage. For evaluation, we reported F1(+) at a threshold of 0.5 and ROC-AUC and PR-AUC (Average Precision, AP) on continuous scores (predict_proba[:, 1]).

3.3. Evaluation and Interpretation of Machine Learning Algorithms

Several machine learning algorithms were evaluated for the prediction of job turnover, and their selection was made based on the specific advantages of each. Logistic Regression was used as a reference model, due to its interpretable and easy-to-explain character. Random Forest and ExtraTreesClassifier were used for their ability to capture nonlinear relationships and highlight the importance of variables. Also, the XGBoost and LightGBM algorithms were added to the analysis to evaluate the benefits offered by the use of boosted decision trees.

Following recent work that emphasizes the joint importance of predictive accuracy and interpretability in attrition modelling, we evaluate models with imbalance-aware metrics and report model-specific explanations to identify factors associated with attrition risk (non-causal) [7].

The Logistic Regression model is frequently used in predictive applications due to its high capability in interpreting results. However, in the context of unbalanced datasets, characteristic of situations encountered in the organizational environment, such as the case of staff turnover prediction, its performance in identifying cases in the minority class requires improvements to ensure a more efficient detection of employees at risk of leaving [8].

Logistic Regression was used as a starting point in the predictive analysis of staff turnover, being a binary classification model well known for its interpretability and robustness to low-dimensional data. We did not combine it with class_weight = 'balanced' to avoid overcompensation.

The model was trained on the balanced dataset using the SMOTE technique, which artificially generated additional examples for the underrepresented class, maintaining the distribution of existing features. After completing the training process, the model's performance was evaluated on the initial (unsampled) test set to evaluate its performance in a scenario as close as possible to its practical use. The importance of the model features is shown in the accompanying graph (Figure 1).

In Logistic Regression, the most influential variables (by absolute value of the coefficients) are the EducationField (e.g., Life Sciences, Medical) and Department (Research & Development) categories, followed by operational factors such as OverTime_Yes, YearsSinceLastPromotion, and satisfaction/involvement indicators (e.g., EnvironmentSatisfaction, JobInvolvement). The interpretation is made relative to the reference category removed by One-Hot (drop_first = True). Positive coefficients indicate an increase in the probability of departure, while negative coefficients indicate an association with retention.

For comparison, we also tested the ExtraTreesClassifier model. ExtraTreesClassifier is a bagging ensemble algorithm that builds multiple completely randomized decision trees based on random subsets of the training dataset. Compared to Random Forest, this model introduces a higher degree of randomization both in the selection of samples and

in the choice of splitting values at tree nodes, which contributes to reducing the variance of the model and combating the phenomenon of overfitting. After applying SMOTE, the ExtraTreesClassifier model was trained on the balanced set, thus benefiting from a proportional distribution of labels that allows for more robust and undistorted learning. To maintain the distribution of classes constant, we used the parameter stratify = y.

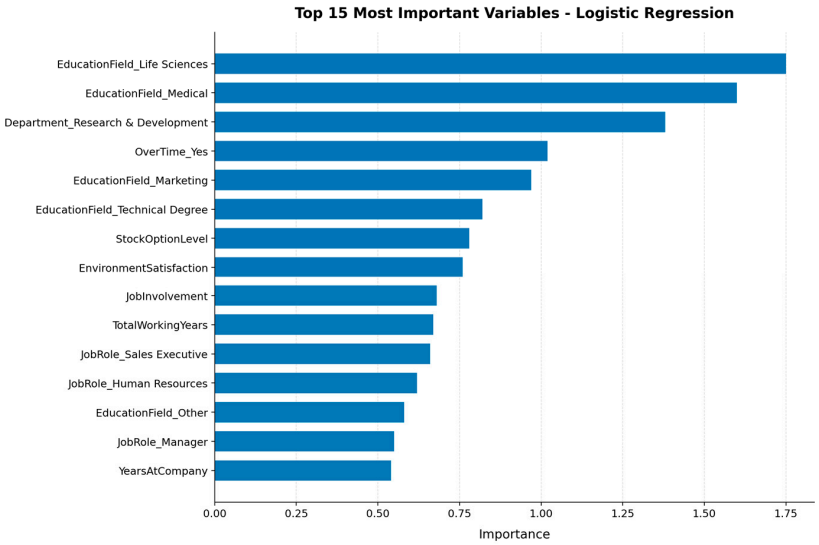

Figure 1. Logistic Regression coefficient magnitudes (One-Hot encoded features).

To ensure the replicability of the results, we used the parameter random_state = 42 in the model implementation. This parameter sets a fixed state of the internal random number generator so that stochastic operations—such as bootstrap sample selection, data shuffling, and tree node split point selection—are consistent across runs. Choosing a fixed value for random_state is standard practice in scientific machine learning experiments, as it allows for direct comparability between models and the exact replication of results. The model was tested, after training, on the original dataset to simulate real-world application scenarios. The contribution of each variable to the model's decision-making process is represented in the graph below (Figure 2).

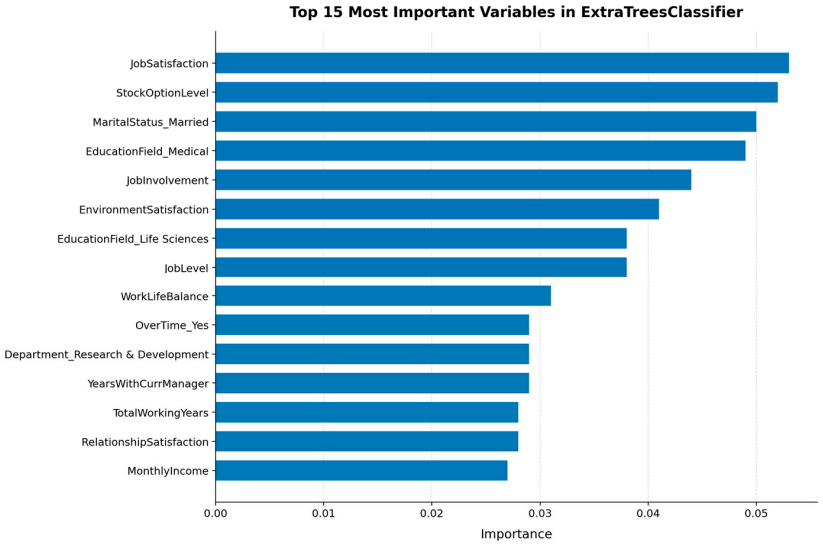

Figure 2. The importance of ExtraTreesClassifier model features.

The ExtraTreesClassifier model identifies job satisfaction, engagement, interpersonal relationships, and financial rewards as the most important predictors of employee turnover.

This analysis provides relevant information for developing retention strategies, focusing the analysis on factors that can be influenced by organizational policies, such as increasing employee satisfaction, promoting work–life balance, or strengthening relationships with management.

The model consistently highlights satisfaction factors (JobSatisfaction, JobInvolvement, EnvironmentSatisfaction, and RelationshipSatisfaction), compensation elements (StockOptionLevel and MonthlyIncome), personal status (MaritalStatus_Married), position/seniority (JobLevel, TotalWorkingYears, and YearsWithCurrManager), and organizational context (Department_Research & Development). The appearance of OverTime_Yes at the top confirms the association between workload and risk. The relative order of importance is consistent with the results on RandomForest, but ExtraTrees tends to be more variable between runs, which is also seen from the larger standard deviations of F1 in k-fold; therefore, the importances should be viewed as trends and not as absolute values.

To complete the analysis, the Random Forest model was also evaluated, and it is based on an architecture similar to that of the ExtraTreesClassifier algorithm, but with a low level of randomization in the tree construction process. Random Forest is an ensemble algorithm in the bagging family that builds multiple decision trees on random subsets of the data and selects the final prediction by majority vote. Each tree is trained on a different bootstrapped sample, and the choice of node split points is based on an optimal criterion, unlike the fully randomized approach in ExtraTrees [9].

This deterministic methodology in choosing the split points gives Random Forest its robust performance in identifying complex relationships between variables, while maintaining a good generalization capacity of the model. Similarly to the other models tested, in this case, the SMOTE technique was used for class balancing, and the dataset was divided while maintaining the proportional distribution of labels-stratify = y to ensure the validity of the evaluation.

For consistency and comparability between models, Random Forest training was performed with the same value of the parameter random_state = 42, thus ensuring the replicability of the results.

The evaluation of the importance of the variables (see Figure 3) highlights compensation and satisfaction in the foreground: StockOptionLevel is the strongest predictor, followed by JobSatisfaction, MonthlyIncome and MonthlyRate. Immediately after, involvement/environment factors appear (JobInvolvement, EnvironmentSatisfaction), as well as level and seniority indicators (JobLevel, TotalWorkingYears). Other relevant variables are HourlyRate and DailyRate (components of the salary package), Age, YearsWithCurrManager (stability of the relationship with the manager), DistanceFromHome (commuting friction), and YearsInCurrentRole. The importance profile suggests that the reward package, organizational climate, and career path are central determinants of the risk of departure.

The graph provides HR with a tool to guide their retention policies.

Designed for speed and scalability, the LightGBM model adopts a leaf-wise strategy, expanding the tree on the branch with the highest error reduction, which allows for faster convergence and increased accuracy in detecting complex relationships between variables. LightGBM also integrates techniques such as Gradient-Based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB), which significantly reduce data size and training time without compromising model performance. In HR analytics applications, the algorithm has demonstrated remarkable results in predicting turnover intention, being preferred in scenarios with large data volumes and unbalanced distributions. In this research, the Light Gradient Boosting Machine (LightGBM) algorithm was used to predict employee turnover based on a structured HR dataset. After importing the data, the target

variable Attrition was binarized (Yes—1; No—0), and categorical variables, such as Gender, were encoded using One-Hot Encoding.

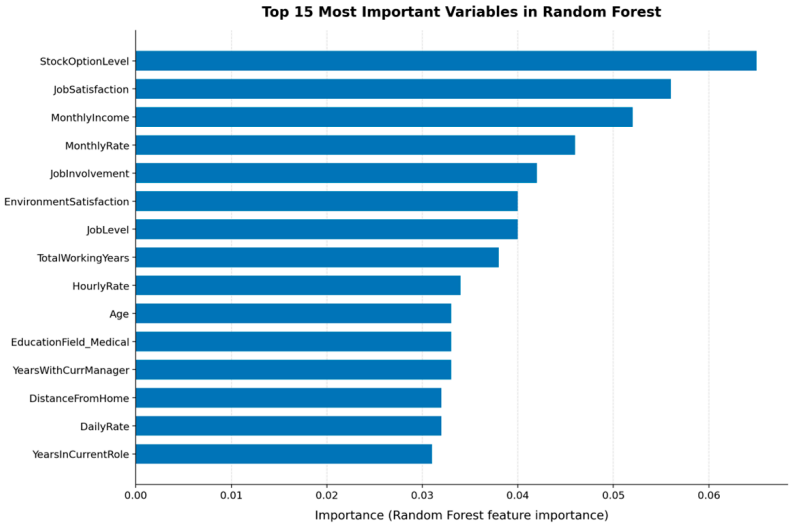

Figure 3. The importance of the characteristics of the Random Forest model.

To address class imbalance (since employees leaving the company number significantly fewer than those staying), the SMOTE technique was applied. It generates synthetic examples for the minority class based on Euclidean distances to existing neighbors, thus contributing to the creation of a balanced training set.

Analyzing the importance of variables (see Figure 4), the model indicates that compensation and satisfaction/involvement variables are major determinants. StockOptionLevel clearly comes out on top, followed by JobSatisfaction, MonthlyRate, and EnvironmentSatisfaction; JobInvolvement and context markers such as WorkLifeBalance and DistanceFromHome appear immediately after. The salary component (Monthly/Daily/HourlyRate, MonthlyIncome) and level/seniority indicators (JobLevel, YearsWithCurrManager) remain consistently relevant. EducationField categories (e.g., Medical, Life Sciences) also appear among the top positions, but their interpretation is made relative to the reference category (One-Hot with drop_first = True) and does not imply causality. Overall, LightGBM confirms that the reward package, climate, and role design explain a significant part of the variation in the risk of departure.

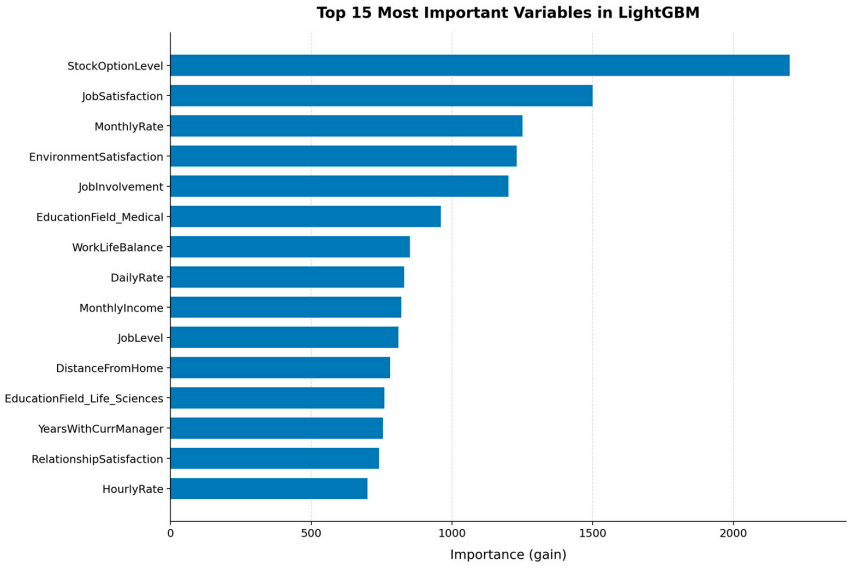

Figure 4. The importance of the characteristics of the LightGBM model.

Overall, the results confirm that LightGBM is capable of capturing complex and relevant relationships between employee characteristics and their organizational behavior, while providing useful information for substantiating strategic decisions in the field of human resources.

XGBoost (Extreme Gradient Boosting) [10] is a high-performance boosting algorithm based on decision trees, introduced in 2014 by Chen et al., which has quickly gained popularity in the scientific community due to its high accuracy and computational efficiency. The model uses an ensemble learning approach, in which a series of weak estimators (simple trees) are built sequentially, each new tree being trained to correct the errors made by the previous trees. By aggregating these trees, XGBoost forms a robust model, capable of capturing complex relationships between variables.

The model (Figure 5) clearly places StockOptionLevel at the forefront, followed by EducationField categories (especially Other/Technical Degree/Marketing/Medical/Life Sciences) and occupational roles (JobRole_Sales Representative, Manufacturing Director, Human Resources). Also appearing are indicators of involvement/satisfaction (JobInvolvement, JobSatisfaction, EnvironmentSatisfaction), level/seniority variables (JobLevel), and organizational context (Department_Research & Development). The profile suggests that the reward package, specialization/role, and involvement/satisfaction contribute substantially to the separation of classes.

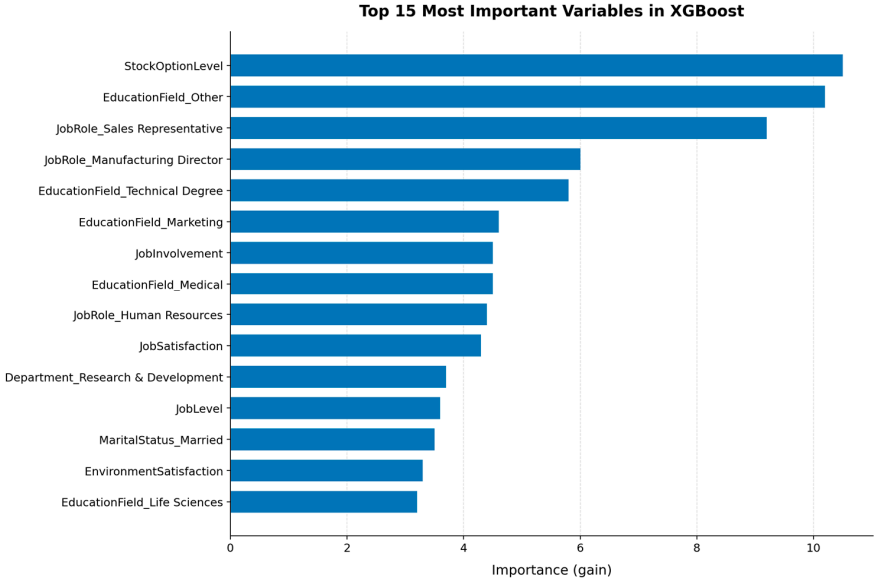

Figure 5. The importance of the characteristics of the XGBoost model.

Tree-based models (Random Forest, LightGBM, and XGBoost) are unanimous in placing StockOptionLevel and JobSatisfaction at the top of the predictors. In contrast, Logistic Regression seems to give more importance to categorical variables with many levels (such as EducationField and Department).

3.4. Experimental Validation and Evaluation Protocol

To robustly assess model performance and avoid reliance on a single train/test split, we used stratified k-fold cross-validation with k = 5. Each fold preserves the unbalanced distribution of the "leave" versus "stay" class.

In each iteration, SMOTE was applied exclusively to the training subset of the fold to rebalance the minority class ("leave") without contaminating the validation data. The validation fold was not oversampled. For reporting precision/recall/F1 on the test set,

we used the default threshold of 0.5 for all models. For each model, we calculated three metrics on the validation fold:

1. F1-score for the "leave" class (the trade-off between precision and recall for the critical class for HR);
2. AUC-ROC, calculated on probability scores (predict_proba[:, 1]) but not on discrete labels to measure the overall separation between classes at all possible thresholds;
3. PR-AUC, reported as Average Precision (AP), which is also calculated on probability scores (predict_proba[:, 1]), is appropriate in unbalanced contexts because it summarizes the precision–recall trade-off for the positive class over the entire range of thresholds. As a reference, the random baseline for AP equals the positive-class prevalence (≈0.161 in this dataset).

As shown in Figure 6, XGBoost maintains higher precision at moderate recall values, consistent with its stronger AP in cross-validation.

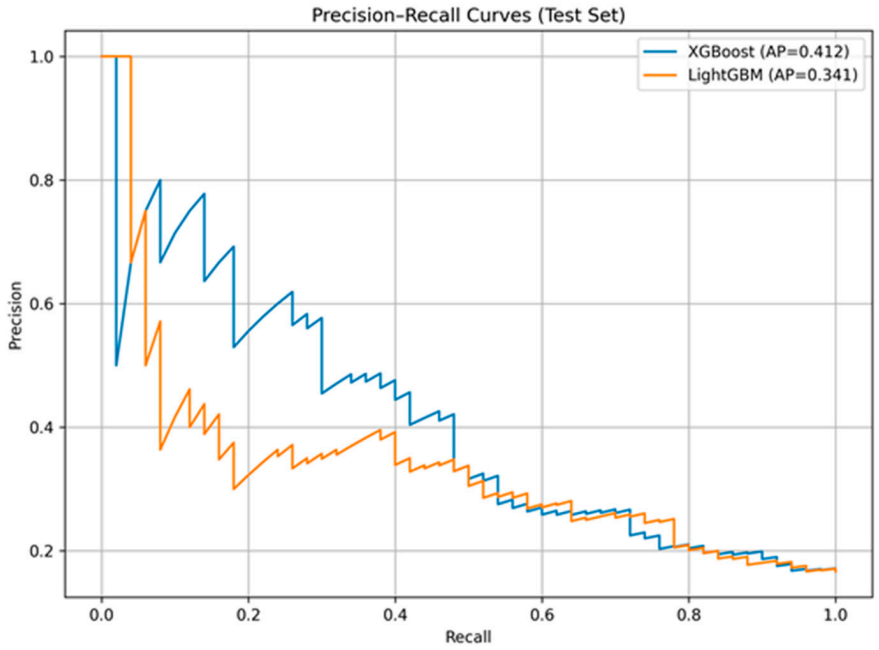

Figure 6. Precision–Recall curves on the test set for XGBoost and LightGBM.

We report the mean and standard deviation (mean ± std) across the five folds as a measure of model stability.

Note that cross-validation results are reported as mean ± std across folds, whereas the test-set values correspond to a single split. Therefore, differences between CV means and test-set point estimates are expected, and the standard deviation reflects fold-to-fold variability rather than uncertainty in the held-out test metric.

Tables 1 and 2 summarize performance under stratified 5-fold cross-validation (mean ± std) and on the held-out test set (threshold = 0.5), together with threshold-independent AP and ROC-AUC computed from predicted probabilities. Logistic Regression achieves the highest mean F1 for the minority ("leave") class (0.439 ± 0.048), indicating strong and consistent minority-class detection across folds. XGBoost attains a similar mean F1 (0.432 ± 0.076) while improving mean ROC-AUC (0.776 ± 0.028 vs. 0.758 ± 0.022 for Logistic Regression), suggesting better overall risk ranking from probability scores. LightGBM provides the highest mean ROC-AUC (0.791 ± 0.026) with competitive F1 (0.419 ± 0.055) and relatively low variability, indicating robust class separation across thresholds.

Table 1. Comparison of model prediction performance and cross-validation.

Model	F1	AUC-ROC	PR-AUC
Logistic Regression	0.439 ± 0.048	0.758 ± 0.022	0.451 ± 0.107
RandomForest	0.339 ± 0.128	0.758 ± 0.037	0.479 ± 0.058
ExtraTreesClassifier	0.309 ± 0.131	0.762 ± 0.038	0.506 ± 0.038
XGBoost	0.432 ± 0.076	0.776 ± 0.028	0.556 ± 0.056
LightGBM	0.419 ± 0.055	0.791 ± 0.026	0.527 ± 0.056

Table 2. Test-set performance for all models (fixed decision threshold = 0.5).

Model	Precision	Recall	Accuracy	F1	AUC-ROC	PR-AUC
Logistic Regression	0.50	0.42	0.83	0.46	0.771	0.438
RandomForest	0.61	0.33	0.85	0.43	0.776	0.370
ExtraTreesClassifier	0.92	0.33	0.88	0.49	0718	0.419
XGBoost	0.65	0.45	0.87	0.53	0.797	0.412
LightGBM	0.61	0.33	0.85	0.43	0.794	0.341

In contrast, non-boosted tree ensembles show lower and substantially more variable minority-class F1 values: Random Forest (0.339 ± 0.128) and ExtraTrees (0.309 ± 0.131). The large standard deviations (>0.12) suggest sensitivity to the training composition, which may translate into less predictable performance when retrained. The higher fold-to-fold variability of Random Forest and ExtraTrees is likely driven by the limited number of minority-class ("leave") samples and the sensitivity of bagging-based trees to local partition choices: Small changes in training composition (and the SMOTE-generated samples) can shift early splits and propagate across the ensemble, affecting threshold-dependent metrics such as F1. In contrast, boosting methods (XGBoost/LightGBM) learn an additive model with shrinkage and regularization, which tends to produce a more stable decision function across folds, leading to lower variance and more consistent ranking-based performance.

Differences between cross-validation averages and a single held-out test split are expected due to sampling variability. While Logistic Regression has the best mean F1 in cross-validation, XGBoost achieves the highest test-set F1 (precision = 0.65; recall = 0.45), reflecting the best precision–recall trade-off on this particular split.

A notable case is ExtraTrees: despite weak and unstable cross-validation F1, it reaches very high test precision (0.92) with low recall (0.33), indicating conservative behavior (few positive predictions, but highly accurate). This reinforces the importance of considering cross-validation stability (mean ± std) alongside point estimates on a single test partition when selecting models for operational use.

Previous work by M. Karimi et al. [4] on staff turnover prediction often evaluates a limited set of classifiers using repeated train/test splits and reports accuracy as the primary metric, with ROC/AUC typically presented only as a supplementary analysis. In contrast, our study emphasizes methodological rigor under class imbalance by adopting a leakage-aware protocol: we perform stratified 5-fold cross-validation and apply SMOTE only within the training portion of each fold, while keeping the held-out test set untouched for final evaluation.

Furthermore, we report imbalance-sensitive metrics—F1 for the minority class (Attrition = 1) and PR-AUC reported as Average Precision (AP)—alongside ROC-AUC, enabling a more informative assessment than accuracy alone in rare-event settings. Finally, we broaden the benchmark to include modern gradient-boosting models (XGBoost and LightGBM) in addition to classical baselines, and we complement performance reporting with an interpretable analysis of predictive risk factors. Recent work [11] has argued that

evaluating imbalanced classifiers benefits from metrics that explicitly account for intrinsic characteristics of the data distribution, rather than relying on a single summary score. In this direction, Chao et al. (2022) [11] propose an efficiency curve framework to assess classifier performance under imbalance by relating performance to data-intrinsic factors and the operating behavior of the classifier across different conditions. This perspective is consistent with our evaluation design: in addition to threshold-dependent metrics (Precision/Recall/F1 at a fixed threshold), we emphasize threshold-independent measures computed from probability scores (PR-AUC reported as Average Precision and ROC-AUC), and we report cross-validation mean ± std to capture robustness across folds (Table 3).

Table 3. Related-work comparison: evaluation rigor, imbalance handling, metrics, and model coverage.

Dimension	Our Paper	Paper "Employee Turnover Analysis Using Machine Learning Algorithms" [4]
Leakage control	Stratified 5-fold CV (mean ± std); SMOTE applied only within training folds.	Repeated train/test splits (3 runs) with accuracy reporting; no central leakage-aware resampling protocol described.
Class imbalance handling	Explicit imbalance treatment via fold-wise SMOTE	No clearly specified resampling strategy.
Metrics	F1 for Attrition = 1, PR-AUC (Average Precision, AP), ROC-AUC (CV and test)	Primarily accuracy (ROC/AUC discussed as an auxiliary analysis).
Models compared	Logistic Regression, Random Forest, ExtraTrees, XGBoost, LightGBM	AdaBoost, SVM, Random Forest.

Prior work has also evaluated employee turnover prediction on the IBM HR Attrition dataset. In particular, the MDPI study "A Hybrid Predictive Model for Employee Turnover: Integrating Ensemble Learning and Feature-Driven Insights from IBM HR Analytics" explores ensemble-based approaches and reports feature-driven insights for interpretability using the same dataset. Building on this line of research, our contribution focuses on strengthening the evaluation protocol under class imbalance by controlling for leakage (fold-wise SMOTE within stratified cross-validation) and by reporting imbalance-sensitive metrics (F1 for Attrition = 1 and PR-AUC/AP) together with ROC-AUC and cross-validation variability (mean ± std) [12].

4. Conclusions and Future Directions

This study shows that machine learning models can support employee retention processes not only by identifying employees at risk of leaving but also by prioritizing cases where early intervention is most important.

Logistic Regression achieves the highest average F1 (0.439 ± 0.048), which means that it recognizes potential "leaving" employees well and does so in a stable way. This recommends it for scenarios where the organization's goal is maximum coverage—that is, "we don't want to miss people who are about to leave," even if that means some false positives.

Boosting-based models, in particular XGBoost and LightGBM, offer an attractive balance between detection and prioritization under class imbalance. Importantly, PR-AUC reported as Average Precision (AP)—which is more informative than ROC-AUC alone in highly imbalanced settings—highlights XGBoost as the strongest ranking model in cross-validation (best mean AP = 0.556 ± 0.056). LightGBM achieves the highest mean ROC-AUC (0.791 ± 0.026) with competitive AP, indicating robust discrimination and ranking performance. The relatively low variability across folds further supports their

suitability for recurring use (e.g., periodic monitoring and risk triage). These conclusions are consistent with the AP results reported in Tables 1 and 2.

The ensemble models on trees without boosting (RandomForest and ExtraTreesClassifier) achieve comparable AUC-ROC values, but they exhibit high variability in the F1 score (± 0.128, ± 0.131). This instability indicates that the number of "cases marked as risk" can vary significantly from one run to another, which makes them harder to use directly as an operational alerting engine without an additional step of adjusting the decision threshold, operational monitoring, and human supervision.

The practical conclusion is that the choice of the model depends on the business objective:

- If the organization wants maximum coverage of the departure risk (recall/F1 on "leave"), Logistic Regression remains a surprisingly competitive and transparent solution;
- If the organization wants a risk prioritization and triage tool (who is most likely to leave, not just who "might leave"), boosting models (especially XGBoost/LightGBM) are the most suitable option due to their strong PR-AUC/AP (and ROC-AUC) and stability across folds.

In addition, the analysis of the importance of variables for tree models (LightGBM, XGBoost) highlights factors such as workload (e.g., overtime), career stagnation (years since last promotion), and compensation level. This information is operationally useful because it highlights potential priority areas for HR interventions (e.g., workload/overtime, career progression, compensation), which should be validated through organizational analysis and/or controlled evaluations.

This study is evaluated on a single publicly available dataset (IBM HR Attrition), which was collected several years ago and may not fully reflect current workforce dynamics, organizational practices, or labor-market conditions. As a result, the absolute performance levels and the relative ranking of models may vary when applied to more recent or organization-specific data with different feature distributions, policies, or attrition drivers. While our primary contribution is methodological, providing a leakage-aware, reproducible evaluation protocol and imbalance-sensitive benchmarking, external validation on additional, more recent datasets is needed to assess generalizability and temporal robustness.

To strengthen external validity, our future research will validate the proposed leakage-aware evaluation protocol on additional, more recent employee attrition datasets and assess cross-dataset generalization. Concretely, we plan to replicate the full pipeline under the same experimental controls (fold-wise resampling, consistent preprocessing, and imbalance-sensitive metrics) across datasets from different time periods and organizations and to evaluate how model performance and feature effects transfer between domains. Such cross-dataset validation will help quantify temporal drift and domain shift and will clarify the extent to which conclusions drawn from the IBM HR Attrition benchmark generalize to current workforce conditions.

Limitations: (i) SMOTE applied to the One-Hot representation may generate less semantically plausible synthetic examples; the results are interpreted as a predictive benchmark. (ii) Importances and coefficients report predictive associations, not causal relationships; therefore, recommendations for HR interventions need to be validated through organizational analysis and/or controlled experiments.

Our future work will include a focused ablation study on imbalance handling, comparing fold-wise SMOTE against cost-sensitive learning (e.g., class weights) and, where applicable, categorical-aware oversampling SMOTENC (SMOTE for nominal and continuous features), assessing their impact on PR-AUC (AP) and F1 for the minority class.

Author Contributions: Conceptualization, A.M.C. and A.N.P.; methodology, A.M.C.; software, A.M.C.; validation, A.M.C. and A.N.P.; formal analysis, A.M.C.; investigation, A.M.C.; resources, A.M.C.; data curation, A.M.C.; writing—original draft preparation, A.M.C.; writing—review and editing, A.M.C. and A.N.P.; visualization, A.M.C.; supervision, A.N.P.; project administration, A.M.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: We used an opened dataset—the IBM HR Attrition dataset—because it is publicly available, widely used as a benchmark, and enables full reproducibility of the leakage-aware protocol and model comparison. We do not need ant ethical approval. There are no new subjects involved.

Informed Consent Statement: We used an opened dataset—the IBM HR Attrition dataset—because it is publicly available, widely used as a benchmark, and enables full reproducibility of the leakage-aware protocol and model comparison. We do not need ant ethical approval. There are no new subjects involved.

Data Availability Statement: The data used in this study are publicly available. The IBM Employee Attrition dataset can be accessed at the following: https://github.com/IBM/employee-attrition-aif360/tree/master/data (accessed on 20 August 2025).

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

SMOTE	Synthetic Minority Over-Sampling Technique;
PR-AUC	Precision–Recall Area Under the Curve;
AP	Average Precision;
ROC-AUC	Area Under the Receiver Operating Characteristic Curve;
CV	Cross-Validation;
HR	Human Resource;
ML	Machine Learning;
DL	Deep Learning;
LLM	Large Language Model;
GPT	Generative Pre-Trained Transformer;
SVM	Support Vector Machine;
GOSS	Gradient-based One-Side Sampling;
EFB	Exclusive Feature Bundling;
LightGBM	Light Gradient Boosting Machine;
XGBoost	Extreme Gradient Boosting;
SMOTENC	SMOTE for Nominal and Continuous Features.

References

1. Dhavan, A.; Kalse, A.A. Utilizing machine learning for predictive analysis of employee turnover and retention strategies. *Anvesak* **2024**, *54*, 76–87.
2. Gahar, R.M.; Hidri, A.; Hidri, M.S. Let's predict who will move to a new job. In *IEEE International Conference on Advanced Systems and Emergent Technologies (IC_ASET), Hammamet, Tunisia, 29 April–1 May 2023*; IEEE: New York, NY, USA, 2023; pp. 2–3.
3. Liu, H.; Ge, Y. Employee turnover prediction: A cross-component attention transformer with consideration of competitor influence and contagious effect. *arXiv* **2025**, arXiv:2502.01660. [CrossRef]
4. Karimi, M.; Viliyani, K.S. Employee turnover analysis using machine learning algorithms. *arXiv* **2024**, arXiv:2402.03905. [CrossRef]
5. IBM Employee Attrition (AIF360) Dataset. Available online: https://github.com/IBM/employee-attrition-aif360/tree/master/data (accessed on 20 August 2025).
6. Sasse, L.; Nicolaisen-Sobesky, E.; Dukart, J.; Eickhoff, S.B.; Götz, M.; Hamdan, S.; Komeyer, V.; Kulkarni, A.; Lahnakoski, J.; Love, B.C. On leakage in machine learning pipelines. *arXiv* **2023**, arXiv:2311.04179.

7. Varkiani, S.M.; Pattarin, F.; Fabbri, T.; Fantoni, G. Predicting employee attrition and explaining its determinants. *Expert Syst. Appl.* **2025**, *272*, 126575. [CrossRef]
8. Runchi, Z.; Liguo, X.; Qin, W. An ensemble credit scoring model based on logistic regression with heterogeneous balancing and weighting effects. *Expert Syst. Appl.* **2023**, *212*, 118732. [CrossRef]
9. Nachouki, M.; Naaj, M.A. Predicting student performance to improve academic advising using the random forest algorithm. *Int. J. Distance Educ. Technol.* **2023**, *20*, 1–17. [CrossRef]
10. Maulana, A.; Faisa, F.R.; Noviandy, T.R.; Rizkia, T.; Idroes, G.M.; Tallei, T.E.; El-Shazly, M.; Idroes, R. Machine Learning Approach for Diabetes Detection Using Fine-Tuned XGBoost Algorithm. *Infolitika J. Data Sci.* **2023**, *1*, 1–7. [CrossRef]
11. Chao, X.; Kou, G.; Peng, Y.; Fernández, A. An efficiency curve for evaluating imbalanced classifiers considering intrinsic data characteristics: Experimental analysis. *Inf. Sci.* **2022**, *608*, 1131–1156. [CrossRef]
12. Muna, I.A.; Hamza, W.K.; Mian, U.S. A Hybrid Predictive Model for Employee Turnover: Integrating Ensemble Learning and Feature-Driven Insights from IBM HR Analytics. *Information* **2026**, *17*, 208. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.