International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 11 | November - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

# Employee Attrition Prediction Using Machine Learning

**Nishighandha Manoj Kumavat[1],**

*Computer Engineering S.S.V.P.S. College of Engineering Dhule*

---***---

**Abstract -** Employee attrition is a gradual but intentional reduction in the number of employees in a company or business organization. Employees will at some point in time look to change their job places for a number of reasons. It might be for professional or personal reasons but it does happen. But it is a diminishing in the number of the workforce due to several factors that are very avoidable. Some forms of attrition are unavoidable, like if an employee is retiring or is moving to another city. But when attrition crosses a particular threshold, it becomes a cause for concern. For example, attrition among minority employee groups could be hurting diversity at your organization. Or, attrition among senior leaders can lead to a significant gap in organizational leadership. Do you know where your company stands on the employee attrition curve? The objective is to analyse the factors that affect employee attrition using previous year's employee's historical data and predict chance of the current employee, whether he could leave the organization. A model is proposed along with an algorithm to predict the chance of employee attrition.

***Key Words***: KNN / Random Forest / SVM

## 1.INTRODUCTION

In the modern corporate landscape, where talent retention is a top priority, organizations face a constant challenge in managing employee attrition effectively. Employee attrition not only affects the organization's bottom line but also disrupts workflow, reduces team morale, and can lead to a loss of valuable institutional knowledge. To combat these challenges, organizations are turning to innovative technologies and data-driven approaches. Machine learning, a subset of artificial intelligence, has emerged as a powerful tool for predictive analytics in various domains. In the context of human resources, machine learning techniques offer the potential to forecast and address employee attrition before it

### Key Features of the Dataset

The dataset contains a wide array of features that provide a comprehensive view of the employees and their attributes. Some of the key features include:

1. **Age:** This feature represents the age of the employee, providing insight into the distribution of age groups within the organization.
2. **Business Travel:** Indicates the frequency of business travel undertaken by employees, categorized as "Travel Frequently," "Travel Rarely," and "Non-Travel."
3. **Department:** Specifies the department in which the employee is working, such as Research & Development, Sales, and Human Resources.
4. **Distance from Home:** Represents the distance in kilometers between the employee's home and workplace, offering insights into commuting patterns.
5. **Education:** Categorizes the employee's educational background, ranging from 'Below College' to 'Doctorate.'
6. **Employee Number:** A unique identifier for each employee in the dataset.
7. **Environment Satisfaction:** Measures the employee's satisfaction with their work environment, graded from 1 (Low) to 4 (High).
8. **Gender:** Indicates the gender of the employee.
9. **Job Involvement:** Rates the level of the employee's job involvement, ranging from 1 (Low) to 4 (High).
10. **Job Role:** Specifies the specific job role or position held by the employee.

occurs. This project sets out to leverage machine learning algorithms to create a predictive model that can assist organizations in identifying employees at risk of attrition. The project's main objective is to develop an accurate and reliable predictive model for employee attrition. This involves the collection and analysis of relevant data, the selection and implementation of appropriate machine learning algorithms, and the creation of a user-friendly interface for easy interpretation and utilization of the model's predictions. The implementation of the project involves using Java for data analysis, algorithm comparison, and model building. JavaFX, known for its rich and interactive graphical user interfaces, will be employed to create an application that visually presents different stages of the project, offering users a comprehensive view of the process and outcomes.

This project report will provide a detailed account of each stage of development, including the methodologies adopted, challenges faced, and the final outcomes achieved. Through rigorous research, algorithmic implementation, and thoughtful

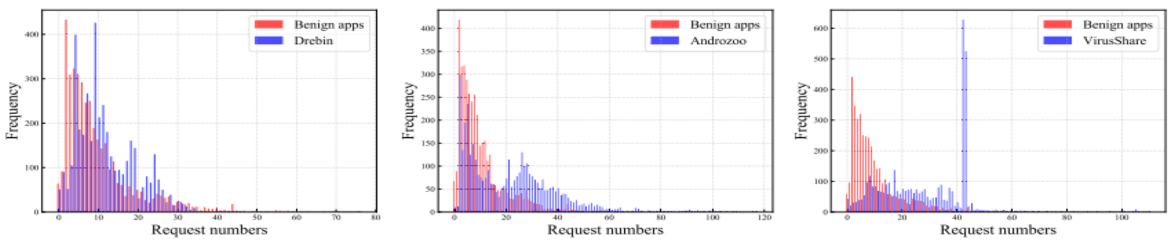
International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 11 | November - 2023 SJIF Rating: 8.176 ISSN: 2582-3930

design, the "Employee Attrition Prediction Using Machine Learning" project aims to contribute to the field of human resource management and predictive analytics.

## 2. MODULES OF THE SYSTEM

The main objectives of this project are as follows:

1. **Develop Predictive Models:** Build and train machine learning models using J48, SVM, and KNN algorithms to predict employee attrition based on historical employee data.
2. **Feature Analysis:** Identify and analyze the significant features that contribute to employee attrition, such as job satisfaction, work environment, compensation, etc.
3. **Algorithm Comparison:** Compare the performance of J48, SVM, and KNN algorithms in terms of accuracy and efficiency for attrition prediction.




(a) Drebin (2010–2012) (b) Androzoo (2017–2020) (c) VirusShare (2018–2020)

**FIGURE 1.** Distribution of the number of permissions required by benign apps and malware samples.

4. **Insights Generation:** Provide actionable insights to organizations about potential attrition risks and key factors driving employee turnover.
5. **Application Development:** Create a user-friendly JavaFx application to showcase the stages of the project, from data analysis to final predictions, providing a rich and interactive user experience.

**TABLE 2.** Top ten permission pairs and their frequencies in Drebin, Androzoo, VirusShare, and the benign dataset. The pairs in gray cells are common ones.

| Drebin (2010–2012) | | Androzoo (2017–2020) | | VirusShare (2018–2020) | | Benign apps | |
|---|---|---|---|---|---|---|---|
| Permission pairs | Frequency | Permission pairs | Frequency | Permission pairs | Frequency | Permission pairs | Frequency |
| INT:RPS | 0.908 | **ANS:INT** | 0.957 | **ANS:INT** | 0.981 | **ANS:INT** | 0.952 |
| **ANS:INT** | 0.697 | INT:RPS | 0.850 | **INT:WES** | 0.969 | **INT:WES** | 0.668 |
| **INT:WES** | 0.681 | ANS:RPS | 0.830 | **ANS:WES** | 0.961 | **ANS:WES** | 0.644 |
| ANS:RPS | 0.670 | **INT:WES** | 0.828 | **AWS:INT** | 0.941 | INT:WL | 0.530 |
| RPS:WES | 0.666 | **ANS:WES** | 0.810 | INT:RPS | 0.941 | ANS:WL | 0.521 |
| **ANS:WES** | 0.525 | RPS:WES | 0.774 | **ANS:AWS** | 0.940 | **AWS:INT** | 0.495 |
| INT:SSMS | 0.510 | **AWS:INT** | 0.666 | ANS:RPS | 0.936 | **ANS:AWS** | 0.493 |
| INT:RBC | 0.502 | **ANS:AWS** | 0.663 | RPS:WES | 0.935 | WL:WES | 0.421 |
| RPS:RBC | 0.491 | AWS:RPS | 0.632 | **AWS:WES** | 0.934 | **AWS:WES** | 0.403 |
| RPS:SSMS | 0.479 | **AWS:WES** | 0.623 | AWS:RPS | 0.917 | INT:VR | 0.371 |

**Machine Learning Algorithms in Employee Attrition Prediction**

1. J48 Algorithm

The J48 algorithm, also known as C4.5, is a decision tree-based classification algorithm. It's widely used for its simplicity and interpretability. The algorithm works by recursively partitioning the dataset based on the attributes that best separate the classes (in this case, employee attrition or not). It selects the most informative attributes to create decision nodes, resulting in a tree-like structure.

In our project, the J48 algorithm plays a crucial role in predicting employee attrition. By analyzing various attributes such as job role, performance, and satisfaction, the algorithm constructs a decision tree that can classify employees as likely to leave or stay. The algorithm's ability to handle categorical and numerical data makes it suitable for our diverse HR dataset. We utilize J48's results to understand the key factors contributing to attrition and make informed decisions to mitigate it.

2. SVM (Support Vector Machine) Algorithm

Support Vector Machine (SVM) is a powerful and versatile algorithm used for classification and regression tasks. SVM aims to find the optimal hyperplane that best separates data points of different classes. It works by transforming data into a higher-dimensional space to maximize the margin between classes. SVM is particularly effective in high-dimensional spaces and is known for handling non-linear relationships.

For our project, SVM is employed to predict employee attrition. By training on features such as job satisfaction, work-life balance, and salary, SVM learns to distinguish between employees likely to leave and those likely to stay. Its versatility allows us to explore complex interactions within the data, enabling accurate predictions and insights.

3. KNN (K-Nearest Neighbors) Algorithm

K-Nearest Neighbors (KNN) is a simple yet effective classification algorithm. It operates based on the principle that similar instances tend to have similar outcomes. KNN assigns a class to a data point based on the class of its nearest neighbors. The "k" parameter determines the number of neighbors considered.

In our project, KNN assists in predicting employee attrition by identifying employees with similar characteristics and comparing their outcomes. By analyzing factors like age, job involvement, and training, KNN can classify employees based on their proximity to others who have left or stayed. KNN's reliance on local patterns in data makes it suitable for uncovering subtle trends in the employee dataset.

## 3. CONCLUSIONS

In conclusion, the "Employee Attrition Prediction Using Machine Learning" project leverages the power of data analysis, machine learning algorithms, and predictive modeling to address a critical challenge faced by organizations. Through the exploration of the IBM HR Analytics Attrition Dataset, we gain insights into the factors influencing employee attrition and build accurate models that assist in identifying potential attrition cases.By selecting appropriate algorithms, preprocessing data, evaluating models, and extracting insights, we equip organizations with the tools needed to make informed decisions and implement strategies that enhance employee retention. This project showcases the potential of data-driven approaches in HR management and paves the way for more sophisticated and comprehensive solutions in the future.

## REFERENCES

[1] (2020). Smartphone Market Share. Accessed: May 2021. [Online]. Available: https://www.idc.com/promo/smartphone-market-share/os

[2] A. Saracino, D. Sgandurra, G. Dini, and F. Martinelli, ‘‘MADAM: Effective and efficient behavior-based Android malware detection and prevention,’’ IEEE Trans. Depend. Sec. Comput., vol. 15, no. 1, pp. 83–97, Jan./Feb. 2018.

[3] Z. Wang, Q. Liu, and Y. Chi, ‘‘Review of Android malware detection based on deep learning,’’ IEEE Access, vol. 8, pp. 181102–181126, 2020.

[4] S. Wang, Q. Yan, Z. Chen, B. Yang, C. Zhao, and M. Conti, ‘‘Detecting Android malware leveraging text semantics of network flows,’’ IEEE Trans. Inf. Forensics Security, vol. 13, no. 5, pp. 1096–1109, May 2018.

[5] S. Garg, S. Peddoju, and A. K. Sarje, ‘‘Network-based detection of Android malicious apps,’’ Int. J. Inf. Secur., vol. 16, no. 4, pp. 385–400, 2017.

[6] Y. Aafer, W. Du, and H. Yin, ‘‘DroidAPIMiner: Mining API-level features for robust malware detection in Android,’’ in Proc. Int. Conf. Secur. Privacy Commun. Syst. Berlin, Germany: Springer, 2013, pp. 86–103.

[7] L. Deshotels, V. Notani, and A. Lakhotia, ‘‘DroidLegacy: Automated familial classification of Android malware,’’ in Proc. ACM SIGPLAN Program Protection Reverse Eng. Workshop (PPREW), 2014, pp. 1–12.

[8] K. Xu, Y. Li, and R. H. Deng, ‘‘ICCDetector: ICC-based malware detection on Android,’’ IEEE Trans. Inf. Forensics Security, vol. 11, no. 6, pp. 1252–1264, Jun. 2016.

[9] H. Cai, N. Meng, B. G. Ryder, and D. Yao, ‘‘DroidCat: Effective Android malware detection and categorization via app-level profiling,’’ IEEE Trans. Inf. Forensics Security, vol. 14, no. 6, pp. 1455–1470, Jun. 2019

[10] B. Sanz, I. Santos, C. Laorden, X. Ugarte-Pedrero, P. G. Bringas, and G. Álvarez, ‘‘Puma: Permission usage to detect malware in Android,’’ in Proc. Int. Joint Conf. CISIS'B ICEUTE SOCO Special Sessions. Berlin, Germany: Springer, 2013, pp. 289–298.

[11] J. Li, L. Sun, Q. Yan, Z. Li, W. Srisa-An, and H. Ye, ‘‘Significant permission identification for machine-learning-based Android malware detection,’’ IEEE Trans. Ind. Informat., vol. 14, no. 7, pp. 3216–3225, Jul. 2018.

[12] S. Liang and X. Du, ‘‘Permission-combination-based scheme for Android mobile malware detection,’’ in Proc. IEEE Int. Conf. Commun. (ICC), Jun. 2014, pp. 2301–2306.
[13] X. Liu and J. Liu, ‘‘A two-layered permission-based Android malware detection scheme,’’ in Proc. 2nd IEEE Int. Conf. Mobile Cloud Comput., Services, Eng., Apr. 2014, pp. 142–148.

[14] A. Arora, S. K. Peddoju, and M. Conti, ''PermPair: Android malware detection using permission pairs,'' IEEE Trans. Inf. Forensics Security, vol. 15, pp. 1968–1982, 2020.

[15] W. Wang, X. Wang, D. Feng, J. Liu, Z. Han, and X. Zhang, ''Exploring permission-induced risk in Android applications for malicious application detection,'' IEEE Trans. Inf. Forensics Security, vol. 9, no. 11, pp. 1869–1882, Nov. 2014.

[16] D. Arp, M. Spreitzenbarth, M. Hubner, H. Gascon, K. Rieck, and C. Siemens, ''DREBIN: and explainable detection of Android malware in your pocket,'' in Proc. NDSS, vol. Effective 14, 2014, pp. 23–26.

[17] K. Allix, T. F. Bissyandé, J. Klein, and Y. L. Traon, ''AndroZoo: Collecting millions of Android apps for the research community,'' in Proc. 13th Int. Conf. Mining Softw. Repositories, May 2016, pp. 468–471.

[18] Virusshare.Com. Accessed: Apr. 2020. [Online]. Available: https:// virusshare.com

[19] Manifest.Permission. Accessed: Jul. 2021. [Online]. Available: https://developer.android.com/reference/android/Manifest.permission?hl=ja

[20] L. Breiman, ''Random forests,'' Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.

[21] C. Cortes and V. Vapnik, ''Support-vector networks,'' Mach. Learn., vol. 20, no. 3, pp. 273–297, 1995.