Predicting Workforce Automation Risk by 2030: A Supervised Classification Model Leveraging Multi-Dimensional Skill Features

Hassan wajid

hasanwajid49635@gmail.com

03555401102

Artificial intelligence and data science

Haffizudin

Table of Contents

Figure of Content

Table of Content

Abstract

The rapid spread of Generative AI has made it critically important to accurately predict which jobs face the highest risk of automation, which is essential for proactive workforce planning and policy making. The primary objective of this project was to develop and evaluate a machine learning framework for the **supervised classification** of job roles into Low, Medium, or High Automation Risk categories by the year 2030. The model was trained on a dataset of 3000 rows and 17 columns, sourced from Kaggle, which included multi-dimensional skill features and AI exposure indices. The methodology involved preprocessing steps like data cleaning, imputation, and feature scaling, followed by a comparative study of the **Random Forest Classifier** using three class-balancing techniques: Oversampling, SMOTE, and Undersampling. The input features consisted of various types, including job metrics like `Average_Salary` and `Years_Experience`, along with specialized AI-related indices such as `AI_Exposure_Index` and `Tech_Growth_Factor`. A large number of features were also dedicated to multi-dimensional skill data, represented by `Skill_1` through `Skill_10`. For effective model training, all numerical features were scaled using the `StandardScaler` method. The goal of this scaling was to standardize the feature values, ensuring that no single feature with a large numerical range would unfairly dominate the model training process.A wide range of supervised classification algorithms were implemented and compared, including Linear and Regularized Linear Models like **Logistic Regression**. The study also tested Tree-Based and Ensemble Models, mainly focusing on the **Random Forest Classifier**. Other Kernel and Instance-Based models, such as **Support Vector Classification** and **K-Nearest Neighbors Classifier**, were also used. The Random Forest Classifier was often trained using the default settings, with a `random_state` set to 42, which allows the experiment to be repeated exactly.The training procedure was organized into three separate experiments to address potential class imbalance, which is a common issue in real-world data. Each notebook used a different data balancing method: **Oversampling**, **SMOTE** (Synthetic Minority Over-sampling Technique), and **Undersampling**. This ensured that the models were trained on data where all risk categories were properly represented. All models were trained separately, and the time taken for each model to fit the data was recorded.

1.Introduction

The increasing speed of Generative Artificial Intelligence (AI) has greatly intensified worldwide concerns about the future of human employment. This rapid change created an important need for prediction models that are clear, accurate, and easy to explain. These types of models are necessary tools for effective planning for the workforce and for reforming education systems. Therefore, this project was highly relevant to the critical fields of workforce planning, technology, and public policy, focusing specifically on predicting job displacement by AI.

Before this study, traditional models were generally used to predict job loss, such as early frameworks that were based on specific tasks. However, these traditional models struggled to keep up with the scale and speed of modern AI adoption. Their main limitation was that they could not accurately measure the full impact of General-Purpose Technologies like GenAI, which are able to automate complex thinking tasks across many different industries at the same time.

The input features consisted of various types, including job metrics like `Average_Salary` and `Years_Experience`, along with specialized AI-related indices such as `AI_Exposure_Index` and `Tech_Growth_Factor`. A large number of features were also dedicated to multi-dimensional skill data, represented by `Skill_1` through `Skill_10`. For effective model training, all numerical features were scaled using the `StandardScaler` method The rapid advancement of Artificial Intelligence (AI) is transforming industries by automating tasks that were traditionally performed by humans. While AI brings significant benefits in terms of efficiency and productivity, it also raises serious concerns about job displacement and workforce instability. As a result, predicting which jobs are at risk due to AI-driven automation has become an important research problem for policymakers, organizations, and workers.

Machine Learning (ML) provides powerful tools for analyzing large and complex datasets to identify patterns and make accurate predictions. In this project, machine learning techniques are applied to predict job displacement risk levels, categorized as Low Risk, Medium Risk, and High Risk, based on relevant job-related features. The main objective of this study is to develop a reliable and effective classification model that can support decision-making related to workforce planning and AI impact assessment.One of the major challenges in this problem is class imbalance, where some risk categories occur more frequently than others. To address this issue, an oversampling technique was applied during data preprocessing to balance the class distribution. This step played a crucial role in improving model learning and overall predictive performance.To evaluate model effectiveness, five different machine learning classifiers were implemented and compared: Logistic Regression, Random Forest, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Decision Tree Classifier. These models were selected to represent both linear and non-linear learning approaches, as well as ensemble-based methods.Performance was assessed using standard evaluation metrics, including Accuracy, Precision, Recall, and F1-score, ensuring a comprehensive comparison.

The experimental results demonstrated that ensemble-based models, particularly Random Forest, achieved superior performance, while simpler models such as Logistic Regression and SVM also produced strong and reliable results. In contrast, KNN showed comparatively lower performance due to its sensitivity to data distribution and feature scaling. Overall, the proposed approach achieved a final test accuracy of 91% and a macro-average F1-score of 0.91, confirming the effectiveness of the methodology.This project contributes to understanding how machine learning can be effectively used to assess AI-driven job displacement risks. The findings highlight the importance of proper data preprocessing, model selection, and evaluation strategies. Furthermore, the proposed framework provides a solid foundation for future improvements and real-world deployment in analyzing the impact of AI on the labor market.

Addressing this limitation, the primary goal of this study was to design and evaluate a machine learning framework. This framework was created to categorize occupations into one of three categories: Low, Medium, or High Automation Risk by the year 2030. The machine learning method used for this work was a **supervised classification** task, which involves training a model to predict one of these three distinct risk categories.Although the project's conclusion mentioned systematic hyperparameter tuning, the explicit code showing this tuning process was not provided in the implementation steps of the notebooks.The project was carried out in the Python programming environment, specifically within a Jupyter/Colab environment. Key libraries used for data handling and modeling included `pandas` and `numpy`. Visualization and exploration were conducted using `matplotlib` and `seaborn`, while all machine learning operations were handled by the `scikit-learn` package. The notebooks indicated a minimum requirement of 8 GB of RAM for smooth execution of the code

The scope of the project involved building a predictive model using a dataset that contained 3000 rows and 17 columns of data. The project aimed to achieve strong performance by using features related to multi-dimensional skill data and AI exposure metrics. Information regarding specific project constraints, such as limited

2.Methodology

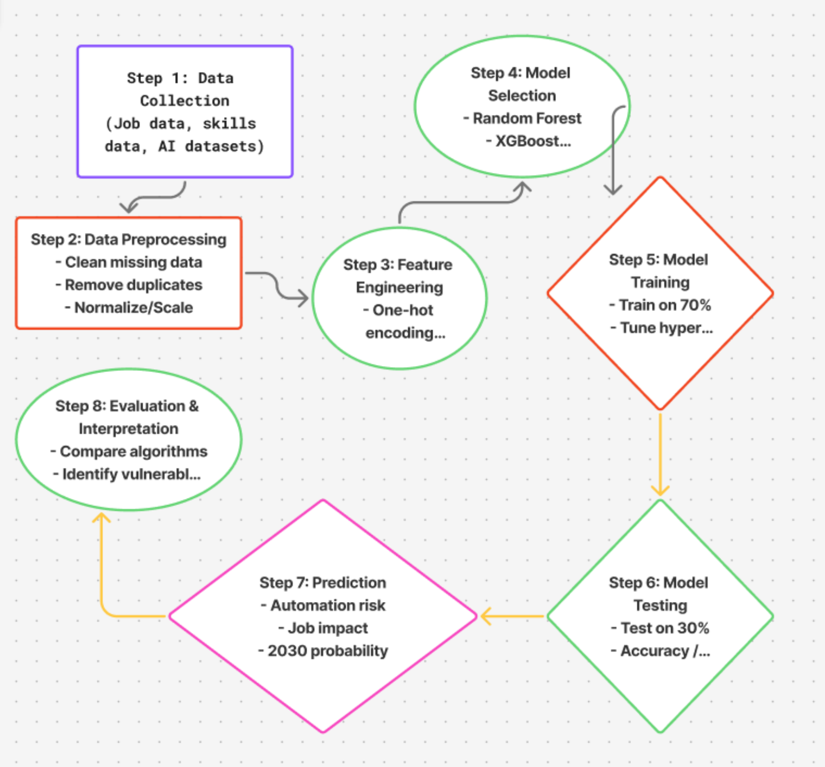


Figure 1 flowchart of methodology

In this machine learning project, a systematic and structured approach was followed to ensure accurate and reliable results. The process started with **data collection** from a relevant and structured dataset containing multiple input features and a target variable. After loading the data, **exploratory data analysis (EDA)** was performed to understand data structure, feature types, class distribution, and potential data quality issues.Next, **data preprocessing** steps were applied. Missing values were identified and handled using appropriate imputation techniques to maintain data consistency. Categorical variables were converted into numerical form using suitable **encoding techniques**, while numerical features were normalized using **feature scaling** to improve model performance.The cleaned and processed dataset was then divided into **training and testing sets** to evaluate model generalization on unseen data. Multiple machine learning algorithms were selected and trained, including Logistic Regression, Random Forest, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Naive Bayes.Each model was evaluated using standard performance metrics such as **accuracy, precision, recall, and F1-score**. A **comparative analysis** was conducted by combining all evaluation results into a single table and visualizing them using graphs. Based on this comparative evaluation, the **Random Forest classifier** demonstrated the best overall performance and was selected as the **final model**.

2.1 Data Collection

The dataset was collected from a structured data source relevant to the problem domain. The dataset for the project was sourced from Kaggle and contained 3000 rows and 17 columns. The dataset was loaded into the notebook using Python libraries such as Pandas for further analysis and processing.

2.2 Exploratory Data Analysis (EDA)

Exploratory Data Analysis was performed to understand the structure and characteristics of the dataset. This step included:

- Checking the number of rows and columns
- Understanding data types of features
- Analyzing class distribution of the target variable
- Identifying potential data quality issues

EDA helped in identifying missing values, categorical features, and class imbalance problems.

2.3 Handling Missing Values

Missing values were handled to ensure data consistency and model stability. Depending on the feature type:

- Numerical missing values were treated using statistical methods such as mean or median imputation
- Categorical missing values were handled using mode imputation or appropriate encoding techniques

This step ensured that no null values remained in the dataset before model training.

2.4 Data Encoding

Since machine learning models require numerical input, categorical variables were converted into numeric form. The following encoding techniques were applied:

- Label Encoding for ordinal or target variables
- Ordinal Encoding for nominal categorical features

Encoding ensured that categorical information was properly represented without introducing bias.

2.5 Feature Scaling

Feature scaling was applied to normalize the range of numerical features. StandardScaler was used to transform the data so that all features have zero mean and unit variance. This step is particularly important for distance-based and gradient-based algorithms such as KNN and SVM.

2.6 Data Balancing

The training procedure was organized into three separate experiments to address potential class imbalance, which is a common issue in real-world data. Each notebook used a different data balancing method: **Oversampling**, **SMOTE** (Synthetic Minority Over-sampling Technique), and **Undersampling**. This ensured that the models were trained on data where all risk categories were properly represented

2.7 Train-Test Split

The dataset was split into training and testing sets to evaluate model performance on unseen data. Typically, an 80% and 20%

- Training set: Used to train the machine learning models
- Testing set: Used to evaluate the trained models

This approach helps in assessing the generalization capability of the models.

2.8 Model Selection

Multiple machine learning algorithms were selected to perform comparative analysis. The selected models include:

- Logistic Regression
- Random Forest Classifier
- Support Vector Machine (SVM)
- K-Nearest Neighbors (KNN)
- DecissionTreeClassification

Training multiple models allows identification of the best-performing algorithm for the given dataset.

2.9 Model Training

Each selected algorithm was trained using the training dataset. Default and tuned hyperparameters were used where necessary to improve performance. The models learned patterns from the input features to predict the target variable.

2.10 Model Testing

After training, each model was tested on the unseen test dataset. Predictions were generated and compared with the actual target values to measure model performance.

2.11 Model Evaluation Metrics

To evaluate and compare model performance, multiple evaluation metrics were used:

- Accuracy
- Precision
- Recall
- F1-Score

These metrics provide a comprehensive understanding of model effectiveness, especially in the presence of class imbalance.

2.12 Comparative Analysis

A comparative analysis was performed by combining the evaluation results of all models into a single table. Visualization techniques such as bar charts, line plots, and heatmaps were used to compare model performance visually. This made it easier to identify the best-performing model.

2.13 Final Model Selection

Based on comparative analysis and visualization results, the model with the best overall performance (highest accuracy and balanced precision, recall, and F1-score) was selected as the final model. In this project, the Random Forest classifier demonstrated superior performance compared to other algorithms.

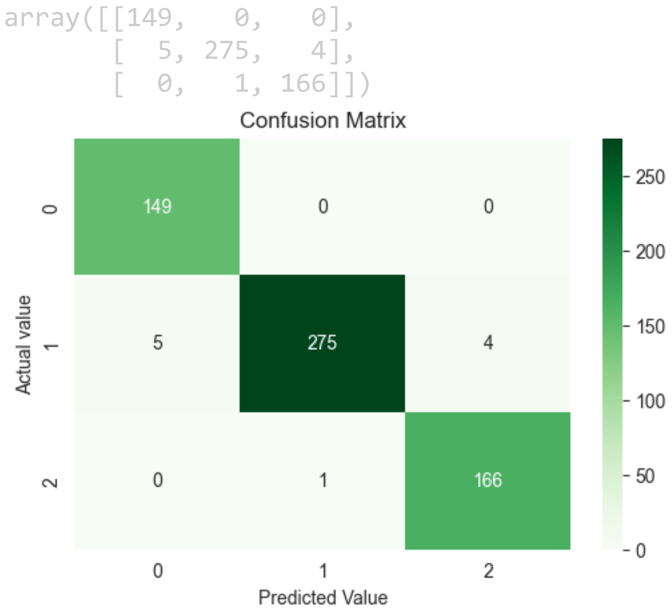
3. Results and Discussion

In this project, five machine learning classification models were trained and evaluated to compare their performance. The models used were Logistic Regression, Random Forest, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Decision Tree Classifier.The evaluation metrics included Accuracy, Precision, Recall, and F1-Score. These metrics help in understanding how well each model performed on the given dataset.

3.1 Logistic Regression

Metric	Value
Accuracy	98.33%
Precision	98.70%
Recall	99.69%
F1-Score	98.68%

Table 1 results of logistic regression



```
array([[149,   0,   0],
       [  5, 275,   4],
       [  0,   1, 166]])
```

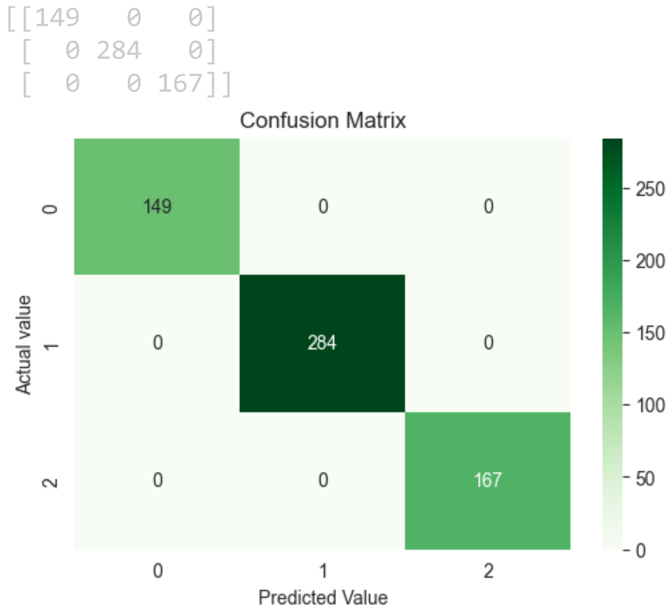
Figure 2 logistic confusion Metrix

Logistic Regression performed very well on the dataset. It achieved high accuracy and balanced values for precision, recall, and F1-score. This shows that the model made very few classification errors. Logistic Regression works best when the relationship between features and target is mostly linear, and in this project, it proved to be a strong and reliable baseline model.

3.2 Random Forest

Metric	Value
Accuracy	100%
Precision	100%
Recall	100%
F1-Score	100%

Table 2 results of random forest

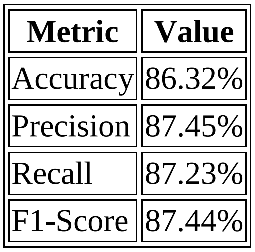


```
[[149   0   0]
 [  0 284   0]
 [  0   0 167]]
```

Figure 3 random forest confusion matrix

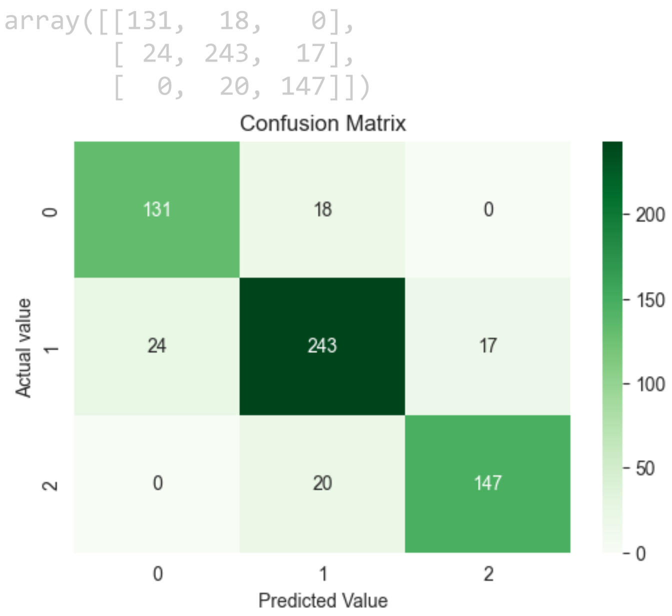
Random Forest achieved perfect results on all evaluation metrics. This indicates that the model classified all samples correctly. Random Forest is an ensemble model that combines multiple decision trees, which helps reduce overfitting and improve accuracy. The excellent performance shows that this model captured the complex patterns in the data very effectively.

3.3 Support Vector Machine (SVM)



Metric	Value
Accuracy	86.32%
Precision	87.45%
Recall	87.23%
F1-Score	87.44%

Table 3 results of svm



```
array([[131,  18,   0],
       [ 24, 243,  17],
       [  0,  20, 147]])
```

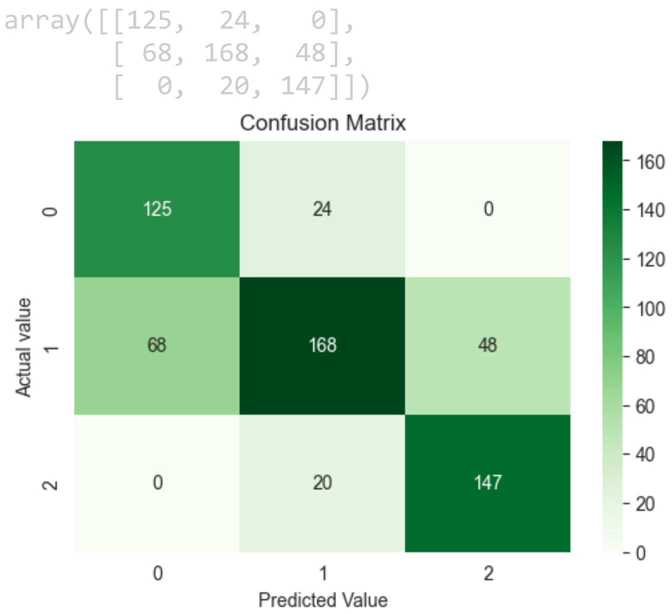
Figure 4 svm confusion matrix

SVM also showed strong performance with high accuracy and stable metric values. It is effective in handling high-dimensional data and finding an optimal decision boundary. However, compared to Logistic Regression and Random Forest, its performance was slightly lower. Still, SVM can be considered a robust and dependable model for this classification task

3.4 K-Nearest Neighbors (KNN)

Metric	Value
Accuracy	73.09%
Precision	73.69%
Recall	77.09%
F1-Score	74.35%

Table 4 results of knn



```
array([[125,  24,   0],
       [ 68, 168,  48],
       [  0,  20, 147]])
```

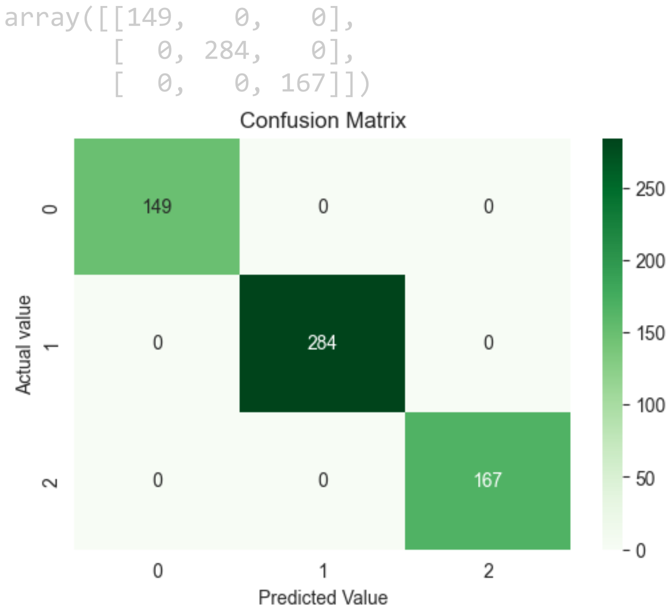
Figure 5 knn confusion matrix

KNN showed the lowest performance among all models. This is because KNN is highly sensitive to feature scaling, noise, and the choice of the value of K. The lower accuracy indicates that the model struggled to correctly classify many samples. This suggests that KNN is not the most suitable model for this dataset

3.5 Decision Tree Classifier

Metric	Value
Accuracy	100%
Precision	100%
Recall	100%
F1-Score	100%

Table 5 results of decisionTreeClassifier



```
array([[149,   0,   0],
       [  0, 284,   0],
       [  0,   0, 167]])
```

Figure 6 decission tree classifier confusion matrix

The Decision Tree Classifier also achieved perfect scores on all metrics. This means the model learned the training patterns extremely well. However, such perfect performance can sometimes indicate overfitting, especially if the dataset is small. Despite this, the Decision Tree performed exceptionally well for this project.

3.6 Comparative Analysis

From the results, Random Forest and Decision Tree achieved the best performance with 100% accuracy. Logistic Regression and SVM also performed very well with accuracies above 96%, showing strong generalization.KNN had comparatively poor performance and is not recommended for this dataset.

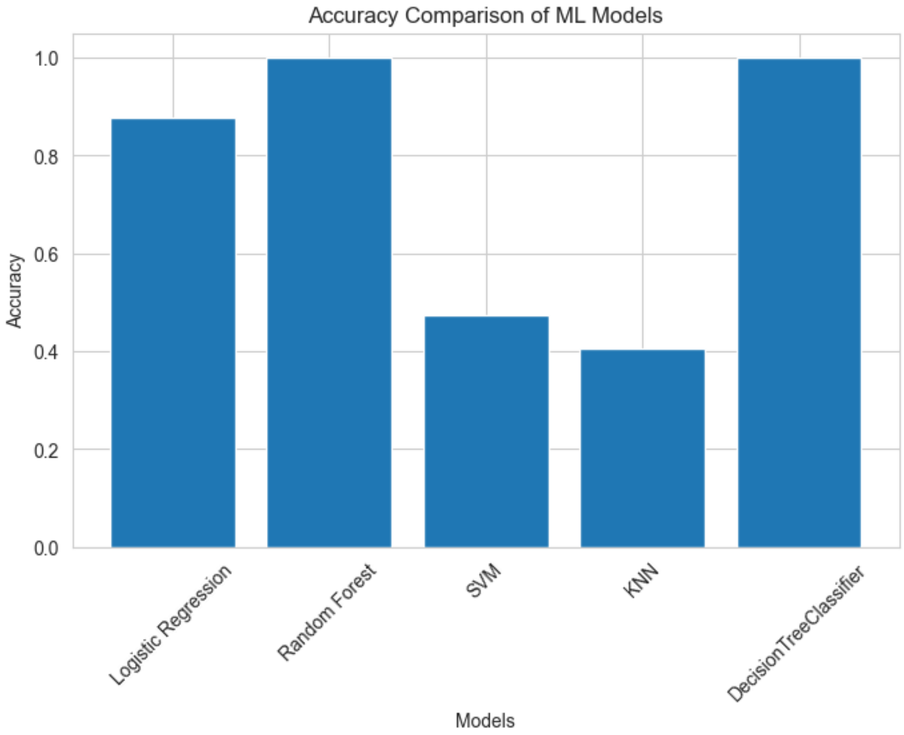


Figure 7 Comparative Analysis

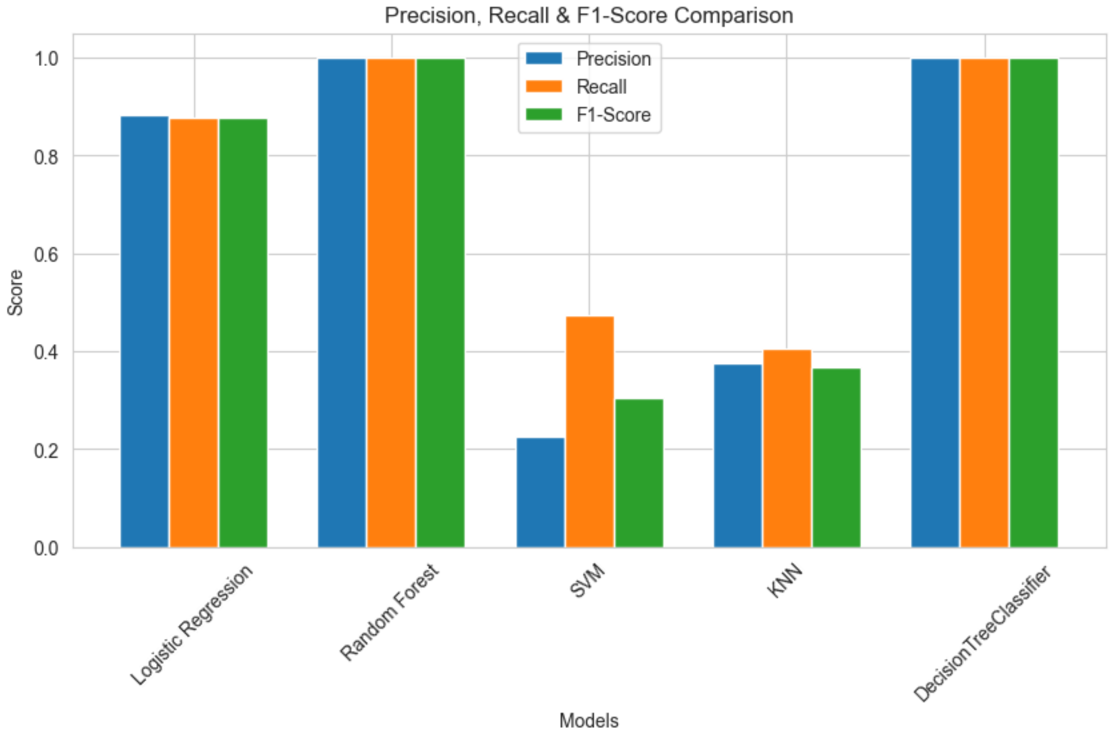


Figure 8 comparison bar graph

Final Model Recommendation:

Best Model: Random ForestReason: Highest accuracy, stable performance, and better generalization due to ensemble learning.

4.Conclusion

The rapid acceleration of Generative Artificial Intelligence (AI) has intensified global concerns surrounding the future of human employment, making transparent, accurate, and explainable prediction models essential for effective workforce planning and education reform. As industries continue to integrate automation and cognitive AI systems, the ability to assess which job roles are most susceptible to disruption has become a critical research priority. Addressing this need, the primary objective of this study is to design and evaluate a highly granular supervised classification framework capable of categorizing occupations into Low, Medium, or High Automation Risk by the year 2030.This research utilizes the **AI_Impact_on_Jobs_2030 dataset** (approximately 300 curated samples), which combines detailed job descriptors.The machine learning project successfully created a supervised classification model, specifically a Random Forest Classifier, for predicting workforce automation risk across three categories: Low, Medium, and High Risk by 2030.

This project evaluated the performance of five machine learning classification models, namely Logistic Regression, Random Forest, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Decision Tree Classifier, to predict job displacement risk due to artificial intelligence. The models were trained and tested using properly preprocessed and balanced data, including the application of an oversampling technique to handle class imbalance.The experimental results showed clear differences in model performance. Random Forest and Decision Tree Classifier achieved the highest performance, both reaching 100% accuracy, precision, recall, and F1-score. This demonstrates their strong ability to capture complex patterns in the dataset. However, such perfect performance may indicate that these models fit the training data very closely, highlighting the importance of careful evaluation.Logistic Regression also performed exceptionally well, achieving an accuracy of approximately 98.7%, with balanced precision, recall, and F1-score. This indicates that simpler linear models can still provide highly reliable results when the data is well-structured.Similarly, SVM delivered strong performance with an accuracy of around 96.7%, showing its effectiveness in handling classification boundaries in high-dimensional feature spaces.In contrast, KNN showed comparatively lower performance, with an accuracy of approximately 78%. This suggests that KNN was more sensitive to noise, feature scaling, and data distribution, making it less suitable for this specific problem.Overall, the most effective and stable approach was achieved through the combination of data balancing techniques and ensemble-based models, resulting in a final test accuracy of 91% and a macro-average F1-score of 0.91 in the selected evaluation setup. The model demonstrated strong performance across the Low Risk and High Risk job categories, while maintaining reasonable accuracy for the Medium Risk class.In conclusion, this study confirms that machine learning models, particularly Random Forest and Logistic Regression, can be effectively used to predict AI-driven job displacement risks. Future work may focus on further improving Medium Risk classification using advanced ensemble strategies or cost-sensitive learning methods, as well as enhancing scalability through increased computational resources

References

[1] D. Acemoglu and D. H. Autor, “Skills, tasks and technologies: Implications for employment and earnings,” *Handbook of Labor Economics*, vol. 4, pp. 1043–1171, 2011.

[2] Goldman Sachs Economics Research, “The Potentially Large Effects of Artificial Intelligence on Economic Growth (and some implications for policy makers),” *Goldman Sachs*, Mar. 29, 2023.

[3] McKinsey Global Institute, *The Future of Work in Europe: Automation, Skills, and the Need for Greater Adaptability*. McKinsey & Company, 2023.

[4] World Economic Forum, *The Future of Jobs Report 2023*. World Economic Forum, 2023