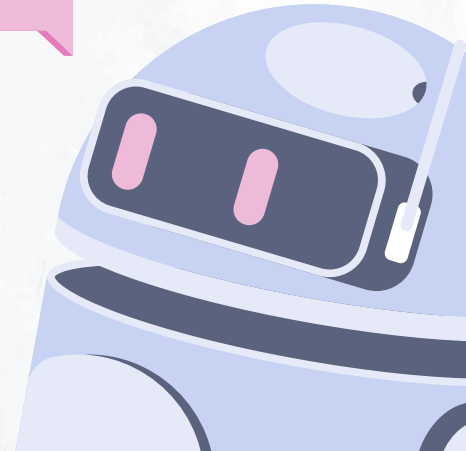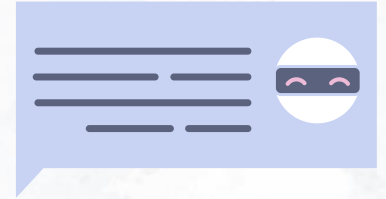# RESUME SCREENING & PLACEMENT PREDICTION MODEL

Presented by:
2105957 - Pushkar Banjare
2105986 - Raj Aryan
2105944 - Akshat Srivastava
2105966 - Ishika Dalai

Content

01 →

Introduction and Problem Statement

Introduction

The main objective of this project is to develop an AI-based automated resume screening system that streamlines the recruitment process, ensuring faster and unbiased candidate evaluation.

Traditional recruitment methods rely on manual resume screening, which is time-consuming, prone to human bias, and often inconsistent due to varying resume formats. These inefficiencies make the process challenging, especially when handling large volumes of applications.

AI and machine learning bring transformative changes to recruitment by automating repetitive tasks like resume parsing and candidate shortlisting. Ensuring consistency and objectivity in decision-making. Providing scalability for processing thousands of resumes efficiently. This project leverages AI to enhance recruitment workflows by using algorithms and data-driven insights to identify the best-fit candidates with speed and accuracy.

Problem Statement

Manual resume screening faces several challenges like it is time-consuming and inefficient, especially with high application volumes. Inconsistencies arise due to varied resume formats and unstructured data. Prone to unconscious bias, leading to unfair and subjective hiring decisions.

Objective:
1. Automate the resume screening process.
2. Utilize OCR (Optical Character Recognition) technology extract text from resumes.
3. Leverage machine learning techniques to improve:

   • Accuracy in identifying qualified candidates.

   • Fairness by eliminating human bias.

   • Speed in processing recruitment applications.

# Key Components and System Architecture

Key Components

OCR Module:
- Extracts text from PDF resumes efficiently.
- Handles varied formats to ensure accurate text recognition.

Machine learning model:
- ➤ K-Nearest Neighbors (KNN)
- ➤ Logistic Regression
- ➤ Support Vector Classifier
- ➤ Random Forest Classifier

- We have used K-Nearest Neighbors (KNN) model as the accuracy for this model was highest.

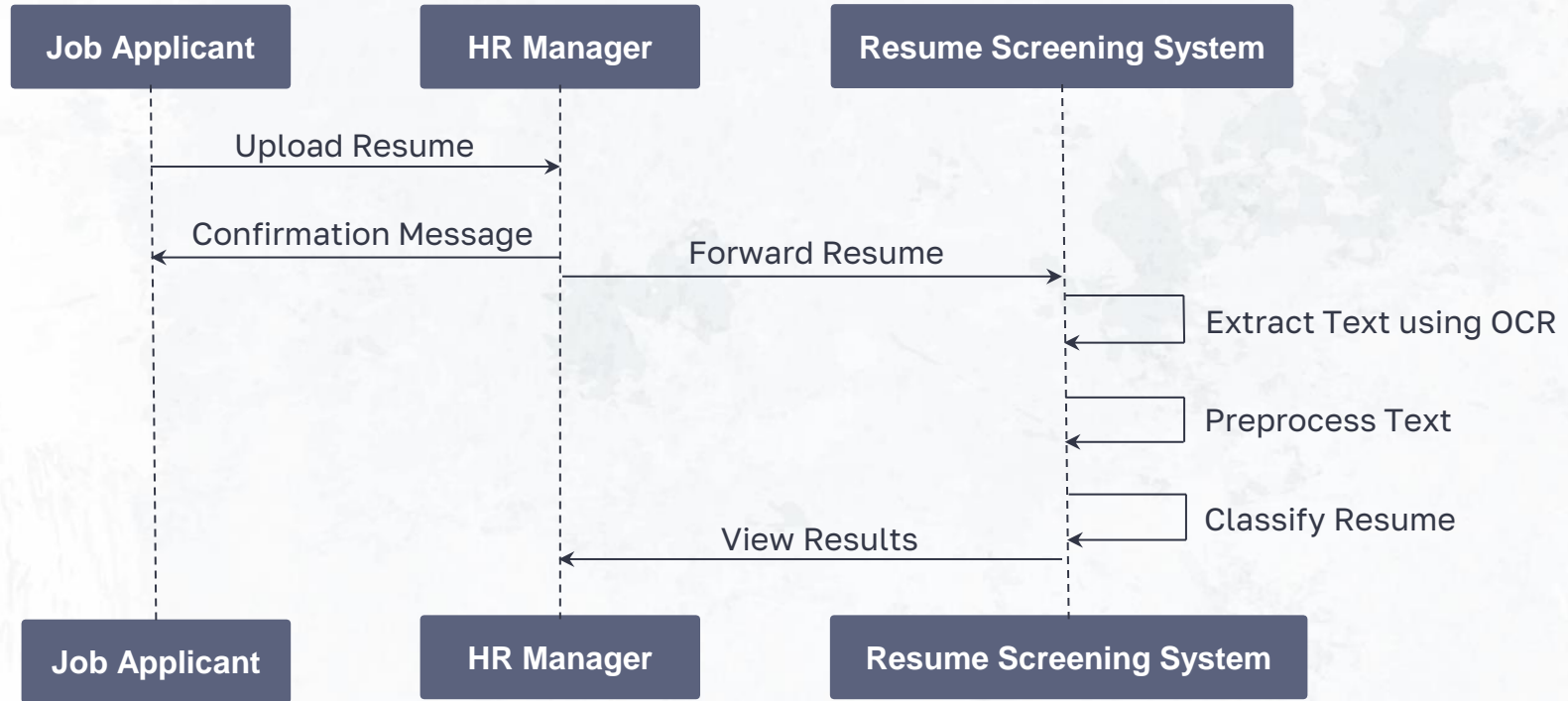# System Architecture
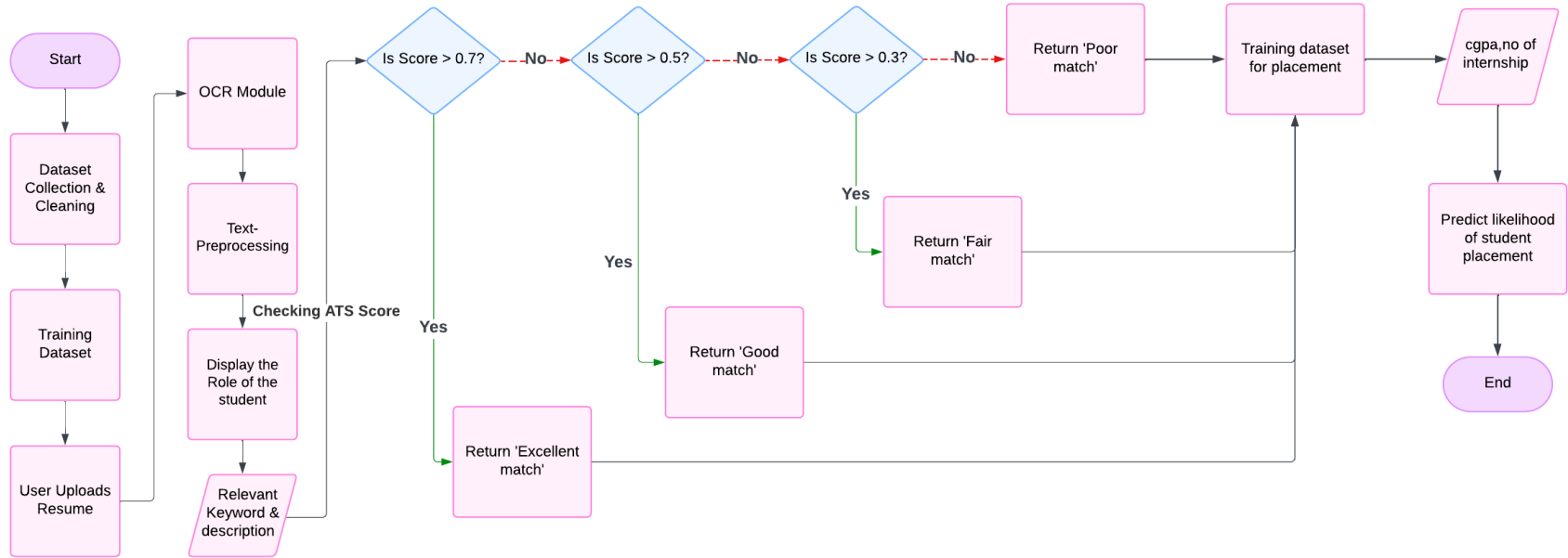
03 —→

Work Flow and Methodology

Work Flow

Methodology

1. Dataset collection & cleaning – Gathering and cleaning resume dataset.
2. Model training – Training the cleaned data.
3. Resume upload & scanning – Extract text from uploaded resume using OCR technology.
4. Text preprocessing – Clean the extracted text to get role.
5. Data Input – Inputting job description and related keywords.
6. ATS score calculation – Matching role and keywords to get ATS score.
7. Dataset collection & cleaning – Gathering additional data for placement prediction.
8. Model training – Training the cleaned data based on CGPA and Internships.
9. Data Input – Inputting the CGPA and Number of Internships done.
10. Final Placement prediction – Analyze whether likely to get placed or not.

04 →

Technology used and Implementation

Technologies Used

Programming Language: Python for core functionality.

Libraries and Tools:

- Scikit-learn for machine learning.

- Tesseract OCR for text extraction.

- Pandas and NumPy for data manipulation.

Cloud Platform: Google Collab for training models and running experiments

Implementation

Data collection – Tesseract module of python is used in OCR to extract text.

Data cleaning – Cleaned the data to make it accurate enough to teach the machine.

Vectorization – Data was vectorized to put into the model.

Model Training – Multiple machine learning models were used like KNN, Logistic Regression, Random Forest Classifier, Support Vector Classifier. Among these KNN was identified as the best model.

Placement prediction – Predicting likelihood of getting placed using ATS from previous dataset and CGPA and internships from second dataset.

05 →

Performance Evaluation and Results

Performance Evaluation

Model Evaluation:

Accuracy: Overall correctness.

Precision: Correctly identified positive cases.

Recall: Coverage of actual positive cases.

F1 Score: Harmonic mean of precision and recall.

System Testing:

Alpha Testing: Perform alpha testing to check system reliability and fix bugs under real-world conditions.

Results

```
Please upload your resume in PDF format:
Choose files  No file chosen                Upload widget is only available when the cell has been executed
Saving Sample Resumes5.pdf to Sample Resumes5.pdf
Given Resume is for: Sales
```

```
Enter relevant keywords for filtering (comma-separated): Sales, Business Development, Lead Generation, Customer Relationship Management, CRM, Negotiation, Account Management,
Enter the job description of the company: Key Responsibilities:  Sales Leadership: Lead and motivate a sales team to achieve sales targets. Business Development: Identify and
ATS Score (Cosine Similarity): 0.58
Interpretation: Good match
Relevant Missing Keywords: None

Your resume already covers the key skills and qualifications from the job description.
```

```
Enter the student's CGPA: 7
Enter the number of internships: 2
The student is likely to get placed.
```

06 →

Conclusion

Conclusion

This project successfully developed an AI-powered automated resume screening system using machine learning techniques. By automating the screening process, the system significantly improves efficiency, reduces human bias, and ensures fairness in recruitment. It leverages advanced technologies like OCR and machine learning algorithms to accurately classify resumes and match candidates with job requirements. The solution is scalable, making it suitable for high-volume recruitment environments, and demonstrates how AI can streamline the recruitment process, saving time and enhancing decision-making in the hiring process.

07 →

References

References

(-) PDF and OCR Processing in Python [online] - https://pypi.org/project/PyPDF2/

(-) Resume Screening using ML [online: Kaggle] - https://shorturl.at/2hGWY

(-) Resume Dataset [online: Kaggle] - https://shorturl.at/wT3yH

(-) Placements Prediction [online: Kaggle] - https://shorturl.at/pXv1m

(-) PDF to OCR Tutorials [online: GFG] - https://tinyurl.com/pd8zc57b

(-) Extracting Text from PDF Files [online: Medium] - https://tinyurl.com/mrb5mc5d

(-) Machine Learning Tutorial [online: GFG] - https://tinyurl.com/2p8xjvrv

(-) Machine Learning Models [online: JavaPoint] - https://tinyurl.com/4dprw5vf

(-) Training Datasets for ML [online: encord] - https://tinyurl.com/3reh7kaj

Thankyou!