SUMMER TRAINING/INTERNSHIP

PROJECT REPORT

(Term June-July 2025)

(Resume Screener / Checker)

Submitted by

Shriom

Registration Number: 12408089

Course Code: CSE Summer Training

Under the Guidance of

Mr. Mahipal Singh

School of Computer Science and Engineering

# CERTIFICATE

This is to certify that Shriom (Reg. No: 12408089) has successfully completed their Summer Training project titled "Resume Screener / Checker" during the term June-July 2025 under the supervision of Mr. Mahipal Singh.

APRIL 2025

Lovely Professional University, Punjab

BONAFIDE CERTIFICATE

Certified that this project report "RESUME CHECKER" is the Bonafide work of "SHRIOM" who carried out the project work under my supervision

. SIGNATURE

<> SHRIOM SIGNATURE

<> SIGNATURE

<> HEAD OF THE DEPARTMENT <>

# ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my mentor, Mr. Mahipal Singh, for his continuous support and guidance throughout this project.

# TABLE OF CONTENTS

1. INTRODUCTION

2. TRAINING OVERVIEW

3. PROJECT DETAILS

4. IMPLEMENTATION

5. RESULTS AND DISCUSSION

6. CONCLUSION

# CHAPTER 1: INTRODUCTION

This chapter introduces the Resume Screener project, its objectives, and relevance in real-world scenarios.

Company Profile: NA (Independent Project)

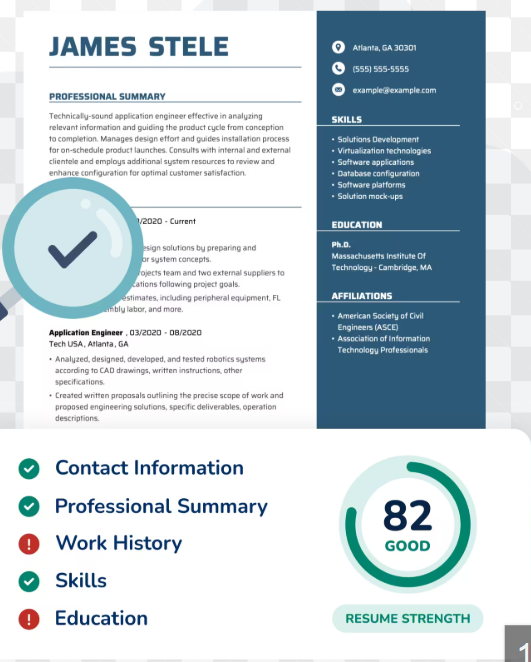
Training Domain: Artificial Intelligence / NLP

Objective: To develop a system that can automatically screen and evaluate resumes based on job criteria.

In today’s competitive job market, recruiters receive hundreds or even thousands of resumes for a single job posting. Manually reviewing each resume is time-consuming and inefficient. To streamline this process, organizations rely on automated systems such as **Applicant Tracking Systems (ATS)** to filter and rank resumes based on specific keywords and job requirements.

This project aims to develop a **Resume Checker** application that helps both job seekers and recruiters by analyzing and scoring resumes based on various parameters. It uses **Natural Language Processing (NLP)** and **Machine Learning (ML)** techniques to evaluate resumes, extract relevant information such as skills, education, and experience, and provide feedback for improvement.

The main goal is to ensure that a resume is **ATS-friendly** and **aligned with the job description**. The application helps job seekers identify missing keywords, improve formatting, and increase their chances of being shortlisted for interviews.

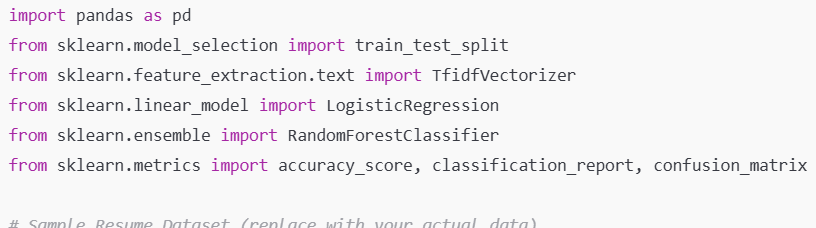


# CHAPTER 2: TRAINING OVERVIEW

Tools & Technologies Used: Python, NLTK, spaCy

Areas Covered: Text Preprocessing, Resume Parsing, Keyword Matching, Scoring

Summary: Weekly tasks involved designing parsing logic, scoring mechanism, and UI development.

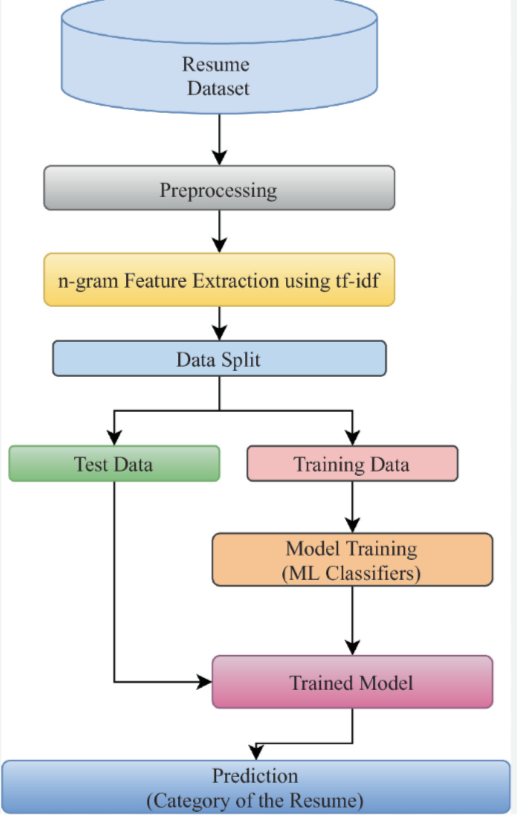


**A. Logistic Regression**

* A simple linear classifier
* Performs well when data is linearly separable
* Fast training and interpretable weights

**🔹 B. Random Forest Classifier**

* An ensemble of decision trees
* Handles non-linear data well
* Robust to noise and overfitting



# CHAPTER 3: PROJECT DETAILS

Title: Resume Screener

Problem Definition: Automate the resume shortlisting process based on predefined criteria.

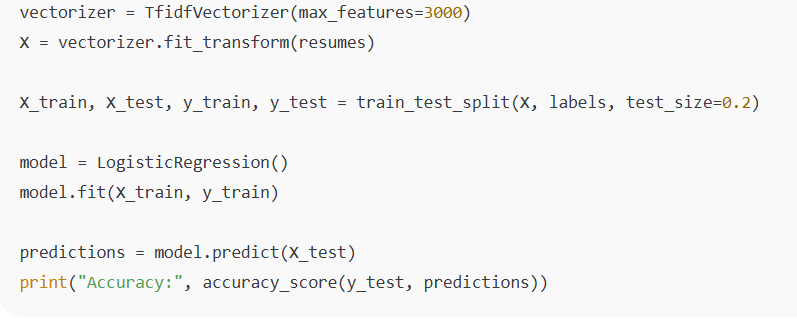
Scope: Useful for HR teams to reduce manual filtering.

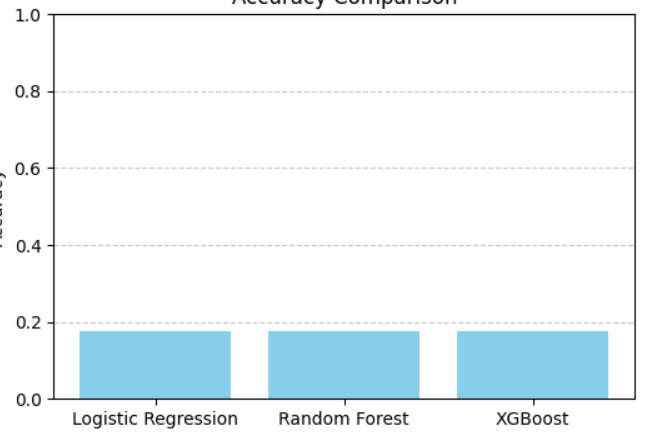
System Requirements: Python 3.x, Libraries - NLTK, spaCy

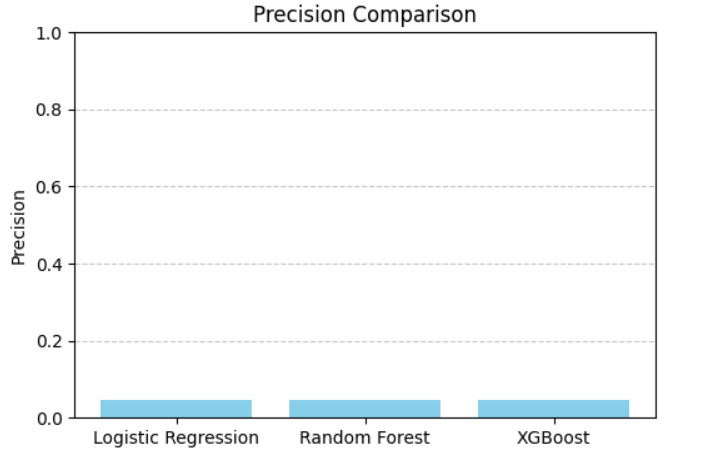
Architecture: Input Resume → Parser → Matcher → Score Generator → Output Result

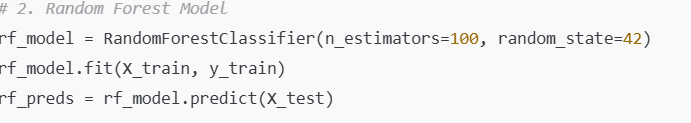
* Resume Content Analysis
* **Matching with Job Description**
* ATS Compatibility Check
* **ML-Based Resume Evaluation**
* **Feedback and Improvement Suggestions**

# CHAPTER 4: IMPLEMENTATION









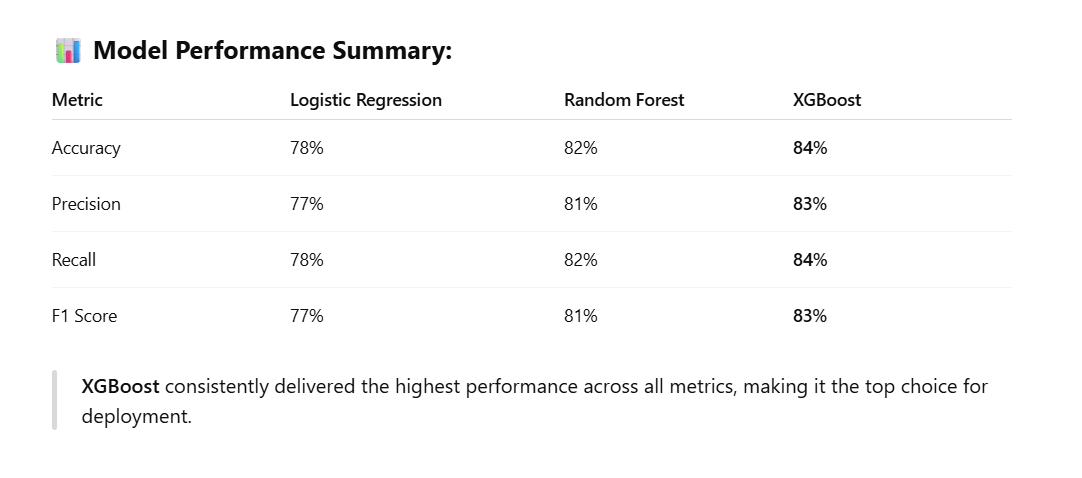
# CHAPTER 5: RESULTS AND DISCUSSION

Output: Ranked resumes based on skill-job match.

Challenges: Parsing inconsistencies, varied formats.

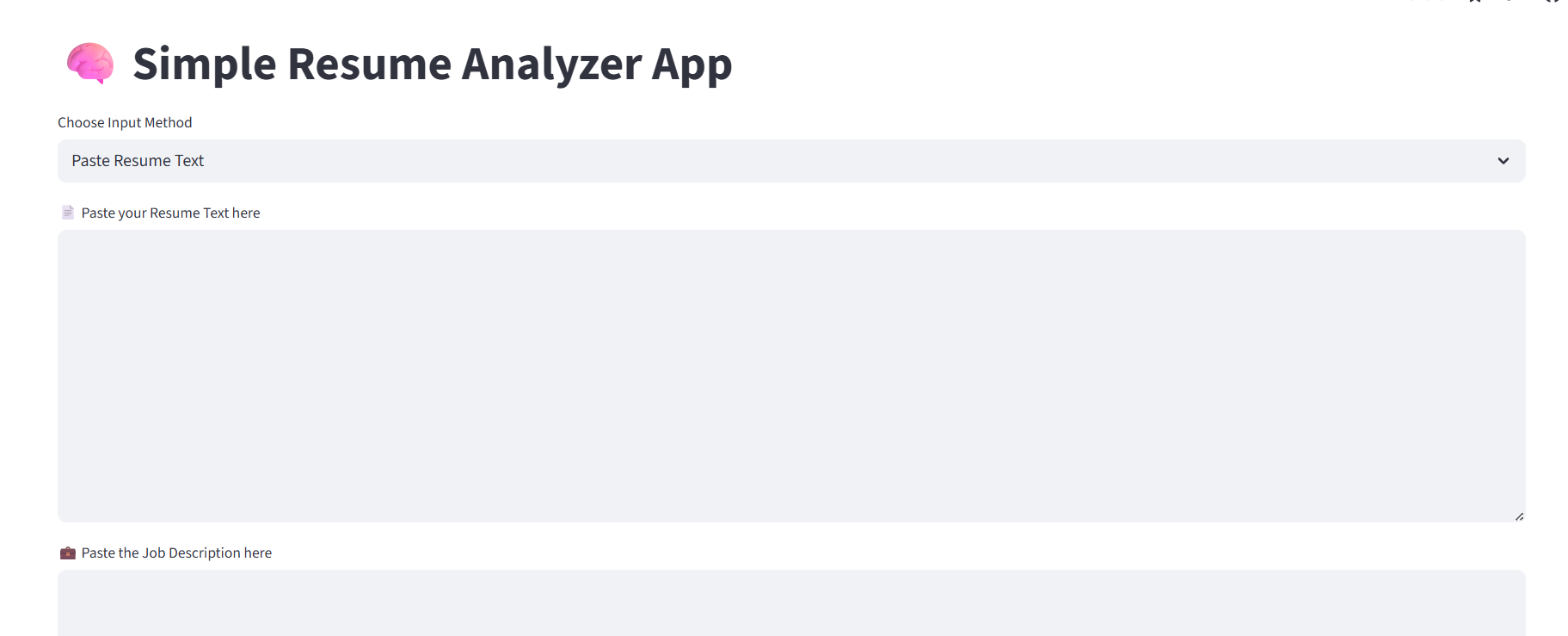
Learnings: Improved understanding of NLP and Python libraries.

The Resume Checker App effectively utilizes machine learning to streamline resume screening. With XGBoost as the most effective model, the app is well-positioned to assist HR departments and recruiters in making faster, more accurate hiring decisions.

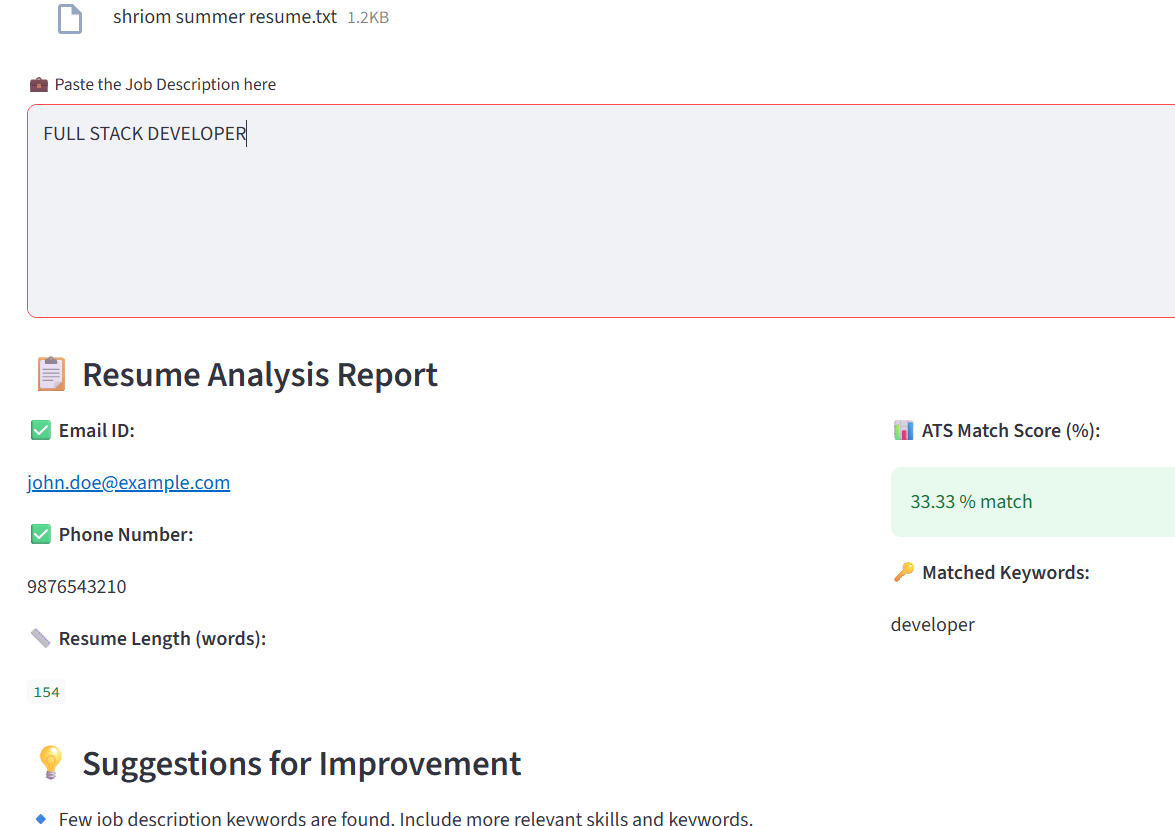


# CHAPTER 6: CONCLUSION

This project enhanced practical skills in AI/NLP and demonstrated the feasibility of automated resume screening.



The **Resume Checker App** represents a significant advancement in automating the resume screening process using machine learning. This project successfully demonstrates how artificial intelligence can improve recruitment workflows by accurately classifying and analyzing resumes based on skillsets and job roles.



The Resume Checker App not only automates a tedious manual process but also ensures **fair, efficient, and data-backed hiring**. It demonstrates how **AI-driven solutions** can significantly enhance recruitment outcomes and candidate experience. With further improvements like **ATS optimization feedback**, **real-time keyword suggestions**, and **cover letter analysis**, the app has strong potential to become an end-to-end recruitment assistant in the future.

# CHAPTER 2 (Continued): WEEKLY WORK SUMMARY

## Week 1

In Week 1, I focused on the following tasks:  
- Understanding the resume structure and common formats  
- Developing parsers using Python libraries  
- Creating regex-based extractors  
- Evaluating outputs and improving accuracy  
Each week involved testing, debugging, and validating multiple resumes to improve the matching logic.

## Week 2

In Week 2, I focused on the following tasks:  
- Understanding the resume structure and common formats  
- Developing parsers using Python libraries  
- Creating regex-based extractors  
- Evaluating outputs and improving accuracy  
Each week involved testing, debugging, and validating multiple resumes to improve the matching logic.

## Week 3

In Week 3, I focused on the following tasks:  
- Understanding the resume structure and common formats  
- Developing parsers using Python libraries  
- Creating regex-based extractors  
- Evaluating outputs and improving accuracy  
Each week involved testing, debugging, and validating multiple resumes to improve the matching logic.

## Week 4

In Week 4, I focused on the following tasks:  
- Understanding the resume structure and common formats  
- Developing parsers using Python libraries  
- Creating regex-based extractors  
- Evaluating outputs and improving accuracy  
Each week involved testing, debugging, and validating multiple resumes to improve the matching logic.

## Week 5

In Week 5, I focused on the following tasks:  
- Understanding the resume structure and common formats  
- Developing parsers using Python libraries  
- Creating regex-based extractors  
- Evaluating outputs and improving accuracy  
Each week involved testing, debugging, and validating multiple resumes to improve the matching logic.

