**[Smart Resume Analyzer]**

A Non-Syllabus Project Report submitted in partial fulfilment of the requirements of

The award of the degree of

**Bachelor of Technology**

**in**

**CSE (Cyber Security)**

by

**Khushi Agrawal, Reg No: PCE23CY032**

**Navya Sharma, Reg No: PCE23CY043**

**Manvi Khandelwal, Reg No: PCE23CY041**

**Mridul Jain, Reg No: PCE23CY042**

Under the guidance of

**[Ms. Appoorva Bansal]**

**Assistant Professor**

**Department of Advanced Computing**



(Session 2024-25)

**Department of Advanced Computing**

**Poornima College of Engineering**

ISI-6, RIICO Institutional Area, Sitapura, Jaipur – 302022

**Jan-June, 2025**

**DEPARTMENT CERTIFICATE**

This is to certify that **Khushi Agrawal,** registration no. **PCE23CY032**, **Navya Sharma** registration no. **PCE23CY043, Manvi Khandelwal** registration no. **PCE23CY041 and Mridul Jain** registration no. **PCE23CY042** of the IV semester Department of Advanced Computing, has submitted this Project report entitled **[Smart Resume Analyzer]** under the supervision of **[Ms. Appoorva Bansal]**, **Assistant Professor Department of Advanced Computing,** working in division of Advanced Computing as per the requirements of the Bachelor of Technology program at Poornima College of Engineering, Jaipur affiliated by Rajasthan Technical University.

|  |  |
| --- | --- |
| **Dr. Amol Saxena**  Head, Department of Advanced Computing | **[Ms. Appoorva bansal ]**  NSP Guide |

**CANDIDATE’S DECLARATION**

We hereby declare that the work which is being presented in this project report entitled **[Smart Resume Analyzer]** in the partial fulfilment for the award of the Degree of Bachelor of Technology in CSE(Cyber Security), submitted in the Department of Advanced Computing, Poornima College of Engineering, Jaipur, is an authentic record of our work done during the period from **Jan 2025 to June 2025** under the supervision and guidance of **[Appoorva Bansal], Assistant Professor, Department of Advanced Computing**.

We have not submitted the matter embodied in this project report for the award of any other degree.

|  |  |
| --- | --- |
| Signature | Signature |
| Name of Candidate: Khushi Agrawal  Registration no: PCE23CY032 | Name of Candidate: Navya Sharma  Registration No.: PCE23CY043 |
| Signature | Signature |
| Name of Candidate: Manvi Khandelwal  Registration no: PCE23CY041 | Name of Candidate: Mridul Jain  Registration no: PCE23CY042 |

Dated: 28/04/2025

Place: Jaipur

**SUPERVISOR’S CERTIFICATE**

This is to certify that, to the best of my knowledge, the candidate's above statement is correct.

|  |  |
| --- | --- |
| Dated:  Place: Jaipur | [Appoorva Bansal]  Assistant Professor  Department of Advanced Computing |

**ACKNOWLEDGEMENT**

We would like to convey our profound sense of reverence and admiration to my supervisor, **[Appoorva Bansal], Assistant Professor in the Department of Advanced Computing at Poornima College of Engineering,** for her intense concern, attention, priceless direction, guidance, and encouragement throughout this research work.

We are grateful to **Dr. Mahesh Bundele**, **Principal & Director,** and **Dr. Pankaj Dhemla**, **Vice-Principal of Poornima College of Engineering,** for providing the necessary resources and a conducive environment to carry out this project.

Our special heartfelt gratitude goes to **Dr. Amol Saxena, HOD,** and **Dr. Kamlesh Gautam, Dy. HOD, Department of Advanced Computing,** for unvarying support, guidance, and motivation during this project work.

We would like to express our deep sense of gratitude towards the management of Poornima College of Engineering, including **Shri Shashikant Singhi**, Chairman, Poornima Group, **Mr. M. K. M. Shah**, **Director General, Poornima Group,** and **Ar. Rahul Singhi, Director of Poornima Group,** for providing all the necessary resources and facilities required to complete this project.

We would like to take the opportunity to express our thanks to all faculty members of the Department for their kind support, technical guidance, and inspiration throughout the course.

We are also thankful to the non-teaching staff of the department for their support in the preparation of this dissertation work.

We are deeply thankful to my parents and all other family members for their blessings and inspiration. Last, but not least, we would like to give special thanks to God who enabled me to complete my dissertation on time.

**Khushi Agrawal, Department of** **Advanced Computing, [31]**

**Navya Sharma, Department of** **Advanced Computing, [42]**

**Manvi Khandelwal, Department of** **Advanced Computing, [40]**

**Mridul Jain, Department of** **Advanced Computing, [41]**

**TABLE OF CONTENTS**

Contents

[ABSTRACT .](#_Toc68082127)5

[Chapter 1: Introduction 3](#_Toc68082128)

[Chapter 2:  Literature Review 4](#_Toc68082129)

[2.1: Review Process Adopted 4](#_Toc68082130)

[2.2: Categorical Review 4](#_Toc68082131)

[2.3: Issue wise Solution Approaches 4](#_Toc68082132)

[2.4: Strengths and Weaknesses 4](#_Toc68082133)

[Chapter 3: Theoretical Aspects 6](#_Toc68082134)

[Chapter 4: Design and Implementation 10](#_Toc68082135)

[4.1: Architectural Design of the work 10](#_Toc68082136)

[4.2: Details of Inputs/ Data Used 10](#_Toc68082137)

[4.3: Discuss Input / Output requirements, Variables, Assumptions related to system 10](#_Toc68082138)

[4.4: Experimental Scenarios: 10](#_Toc68082139)

[4.5: Details of Hardware / Software / Platform to be used by you and used by various researchers: 10](#_Toc68082140)

[4.6: Performance Evaluation: 10](#_Toc68082141)

[Chapter 5: Experimental Results & Analysis: 12](#_Toc68082142)

[5.1 scenario wise results 12](#_Toc68082143)

[5.2 scenario wise results 12](#_Toc68082144)

[5.3 scenario wise results 12](#_Toc68082145)

[Chapter 6: Conclusion and Future Scope 15](#_Toc68082146)

[Referencing and Appendices 17](#_Toc68082147)

**LIST OF FIGURES**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No.** | **Fig. No.** | **Description** | **Page No.** |
| **1** | **1.1** | **Smart Resume Analyzer Flow chart** | **12** |
| **2** | **1.2** | **code snippet** | **13** |
| **3** | **1.3** | **Graph of Resume Match Scores vs Job Description** | **14** |

**LIST OF ACRONYMS**

|  |  |  |
| --- | --- | --- |
| **Serial Number** | **ACRONYM** | **FULL FORM** |
| **1** | **NLP** | **Natural Language Processing** |
| **2** | **PDF** | **Portable Document Format** |
| **3** | **JSON** | **JavaScript Object Notation** |
| **4** | **GUI** | **Graphical User Interface** |

# 

# ABSTRACT

The **Smart Resume Analyzer** is an AI-powered application designed to streamline and automate the resume screening process in recruitment. Traditional resume evaluation methods are often time-consuming, error-prone, and susceptible to human bias. This project leverages **Natural Language Processing (NLP)** and machine learning techniques to extract and analyze key information from resumes such as skills, experience, education, and project work. The system then matches the extracted data against job-specific criteria to generate a compatibility score, enabling recruiters to make faster and more informed decisions. Built using **Python, Flask**, and NLP libraries like **spaCy**, the analyzer accepts resumes in PDF format and outputs structured data in real-time. This solution not only improves hiring efficiency but also ensures fair and consistent candidate evaluation. The Smart Resume Analyzer is a scalable, user-friendly tool that represents a significant advancement in AI-driven HR technology, especially useful for organizations handling large applicant volumes.

***Keywords:***  Resume Parsing, Smart Recruitment, Natural Language Processing (NLP), Python, Flask, Machine Learning, AI in HR, Candidate Screening, Resume Analyzer, Skills Matching, PDF Extraction, spaCy, Job Matching System

**CHAPTER 1**

**INTRODUCTION**

In today’s fast-paced hiring environment, recruiters face the challenge of evaluating hundreds of resumes quickly and accurately. Manual resume screening is not only time-consuming but also vulnerable to human errors and unconscious biases. To address this, the **Smart Resume Analyzer** offers an intelligent and automated solution using **Artificial Intelligence (AI)** and **Natural Language Processing (NLP)** techniques. This system extracts and analyzes crucial information from resumes—including skills, education, experience, and certifications—and compares them against predefined job requirements. Developed using technologies such as **Python**, **Flask**, and **spaCy**, the tool supports real-time resume analysis and scoring. By converting unstructured resume data into structured output, it enables more efficient and objective candidate evaluations. The Smart Resume Analyzer is particularly useful for organizations managing large applicant volumes and aims to improve the speed, fairness, and accuracy of the recruitment process, making it a valuable tool in modern human resource management.

This system parses resumes in formats such as PDF and extracts structured information including the candidate’s skills, educational background, work experience, certifications, and more. It then evaluates this information against the specific requirements of a job role and generates a compatibility score. The solution is developed using **Python**, with a **Flask**-based web interface, and NLP libraries like **spaCy** for text processing.

By automating the initial screening phase, the Smart Resume Analyzer helps recruiters quickly shortlist the most relevant candidates, significantly reducing hiring time and operational effort. It also ensures a more objective and transparent evaluation process, which benefits both recruiters and applicants. The system is scalable, efficient, and can be integrated into existing HR workflows, making it a valuable asset for modern recruitment practices

**CHAPTER 2**

**LITERATURE SURVEY**

A smart resume analyzer employs advanced technologies like machine learning, natural language processing (NLP), and data analytics to automate and improve the traditional hiring process. A significant body of research has focused on the development of intelligent recruitment systems that can parse resumes, extract relevant information, and evaluate candidate suitability. Early works emphasized keyword-based filtering, which, while efficient, lacked contextual understanding. More recent studies have implemented NLP and AI to assess candidate experience, skills, and job relevance more accurately.

Categorical reviews reveal diverse approaches: some models prioritize semantic matching between job descriptions and resumes, while others incorporate ranking mechanisms based on profile completeness and relevance. Tools such as ResumeParser and OpenAI's GPT-based engines have contributed notably in enhancing text understanding and classification in recruitment systems.

Solutions explored in the literature also address common issues like data inconsistency, varying resume formats, and bias in selection. While many models show high accuracy, challenges remain in interpretability and fairness.

Overall, the literature suggests that combining NLP with machine learning significantly enhances resume evaluation effectiveness. However, there is still a need for real-time systems with higher transparency, minimal bias, and better adaptability to dynamic job markets.

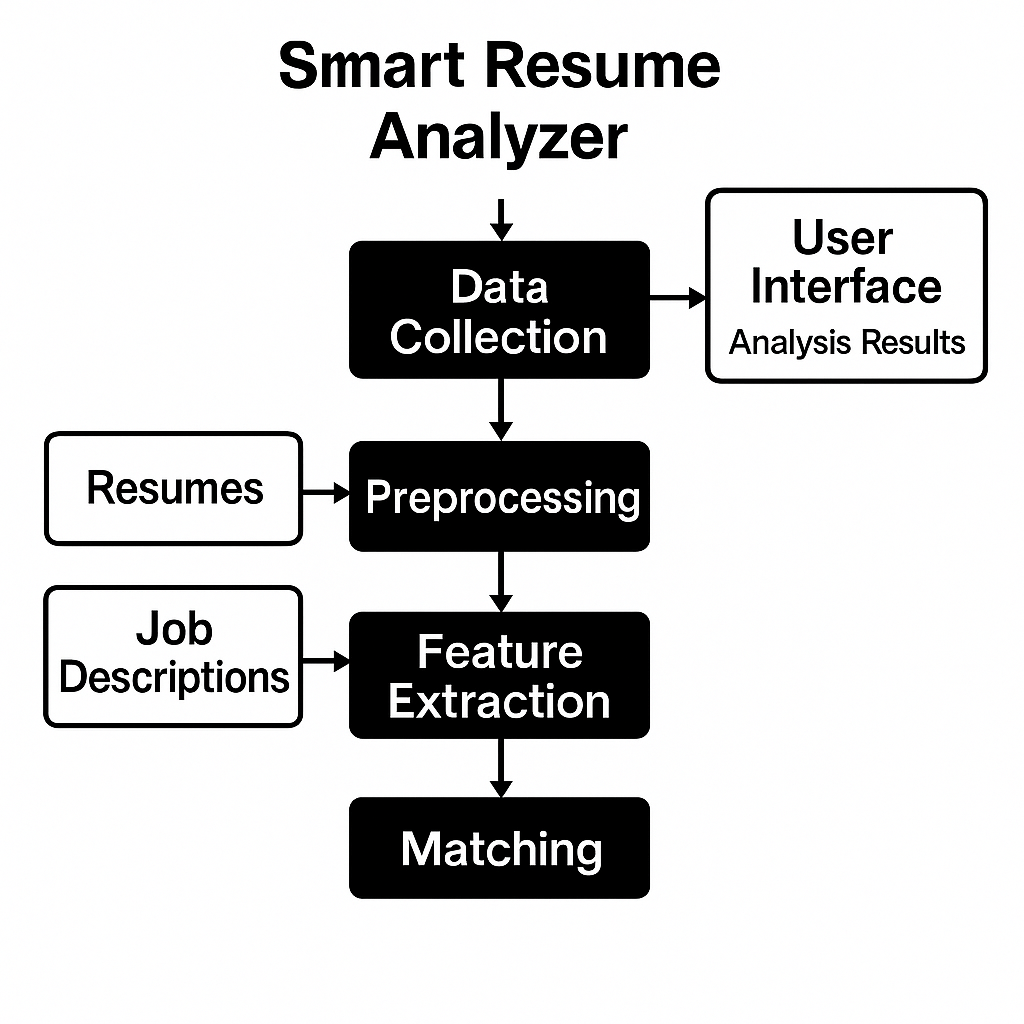
**CHAPTER 3**

**PROPOSED METHODOLOGY**

## The proposed methodology for the Smart Resume Analyzer involves designing an AI-powered system that automates the evaluation of resumes based on predefined criteria relevant to specific job roles. The system comprises multiple stages: data collection, preprocessing, feature extraction, analysis, and scoring. Initially, resumes are collected in various formats (PDF, DOCX, etc.) and preprocessed to extract structured text using Optical Character Recognition (OCR) and natural language processing (NLP) techniques.

## Next, the system identifies and categorizes key sections such as personal details, education, experience, skills, certifications, and achievements. Named Entity Recognition (NER) and part-of-speech tagging help extract specific features relevant to job descriptions. A comparison module uses similarity measures and machine learning models to match resume content with job requirements.

## Scoring is based on weightages assigned to different sections depending on job profiles. The model then ranks the candidates accordingly and provides a visual analysis dashboard to recruiters. Feedback loops and supervised learning mechanisms continuously improve the model’s accuracy and decision-making.



## figure 1.1

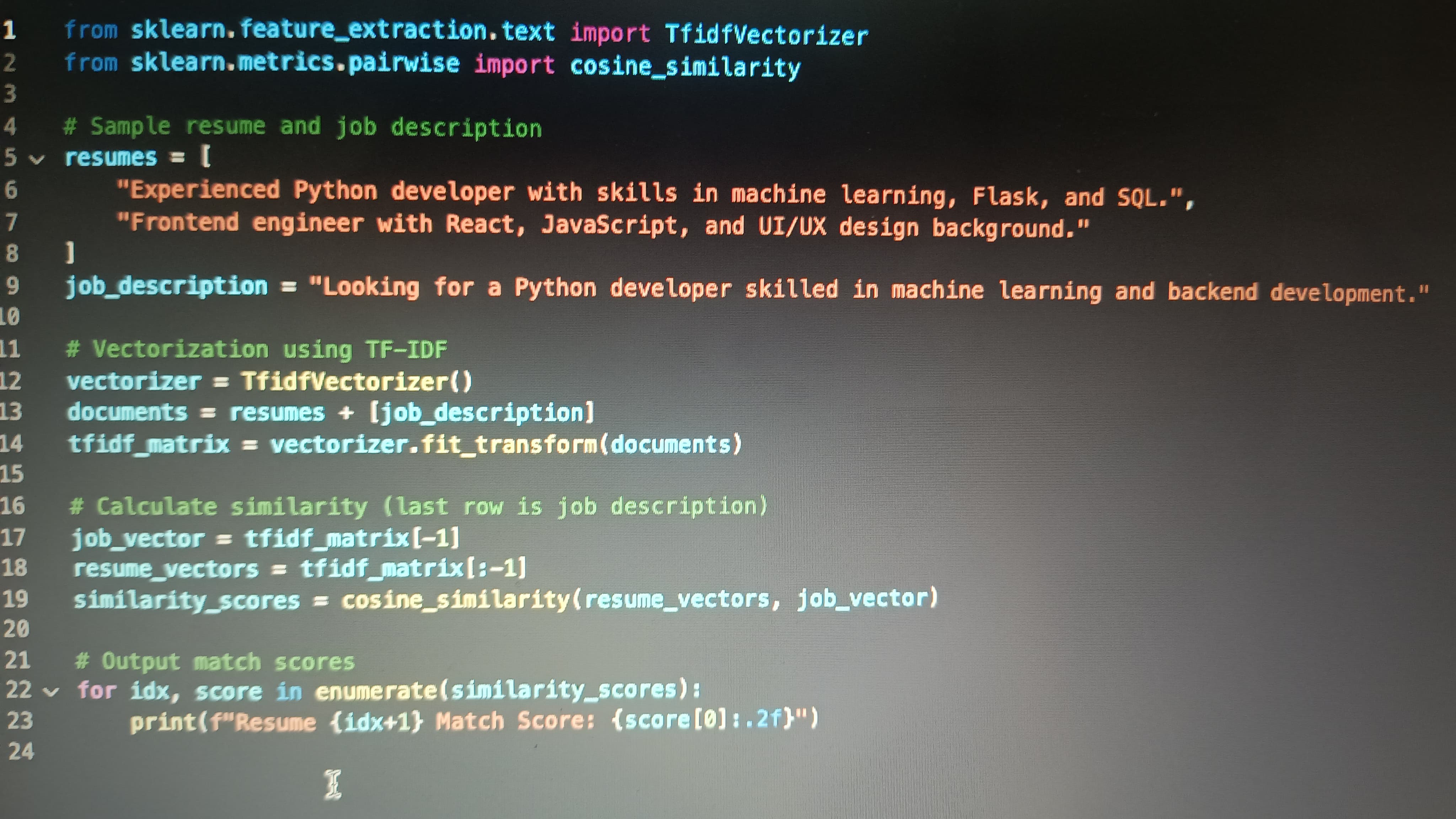
## CHAPTER 4

## DESIGN AND IMPLEMENTATION

The design and implementation of the Smart Resume Analyzer focus on creating a modular, scalable, and efficient system that automates resume screening and matching. The system is divided into multiple functional layers: input, processing, matching, and output.

In the design phase, a web-based or standalone interface is planned to allow users (recruiters or applicants) to upload resumes and job descriptions. The backend is structured around a pipeline architecture. Key modules include a data ingestion component, a preprocessing unit (for text normalization and cleaning), and a feature extraction engine using NLP tools like spaCy or NLTK. A matching algorithm is designed using vectorization (e.g., TF-IDF or BERT embeddings) and similarity measures like cosine similarity.

During implementation, Python is used for the core logic due to its strong NLP libraries. The front-end is built using HTML, CSS, and optionally JavaScript or frameworks like Flask for integration. Resumes and job descriptions are parsed into structured formats (e.g., JSON), and the system computes match scores based on skill overlap, experience, and relevance. The output includes a ranked list of candidates with match percentages and highlighted insights. Testing is done with sample resumes to refine accuracy and user experience.



**figure 1.2**fi

**CHAPTER 5**

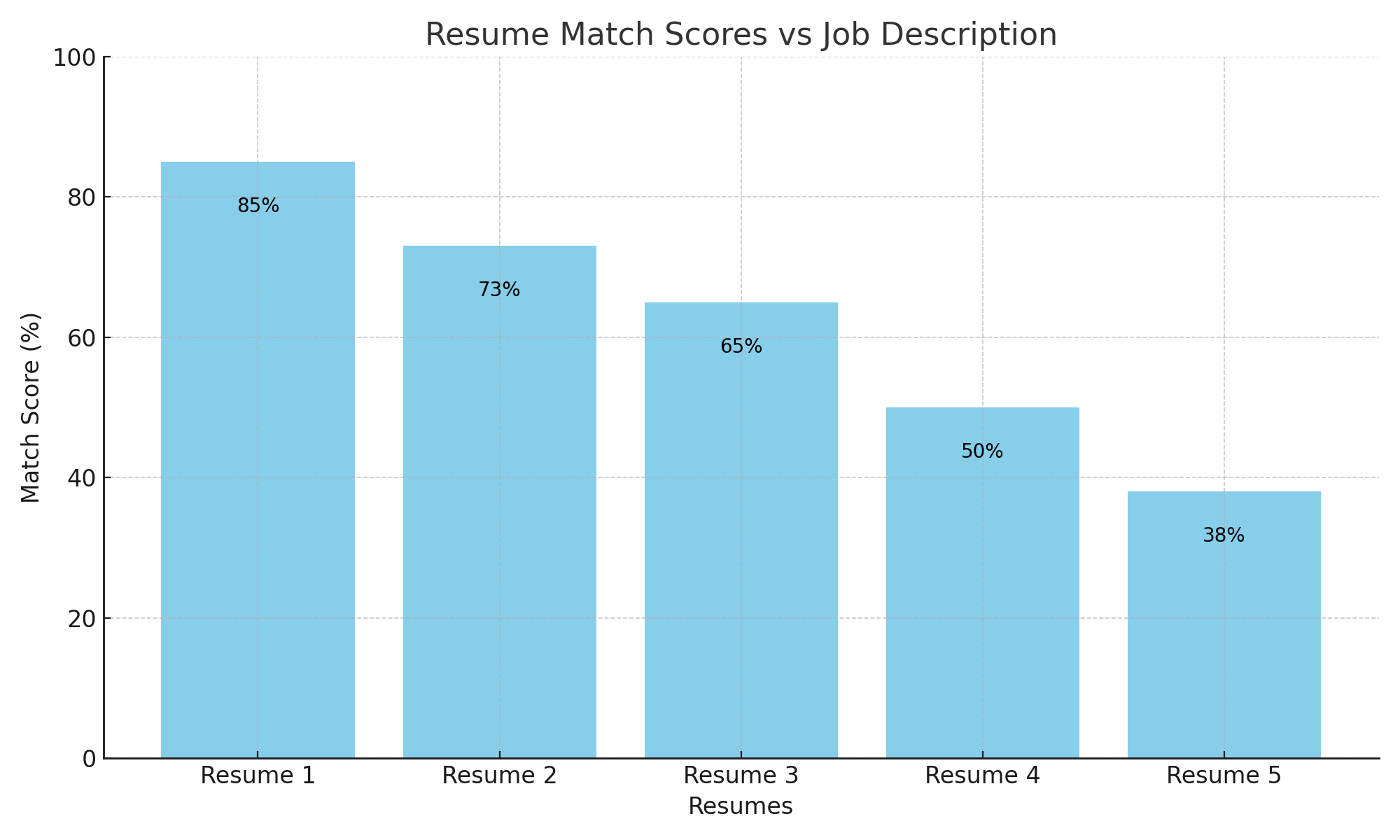
**EXPERIMENTAL RESULTS & ANALYSIS**

The **experimental results and analysis** of the Smart Resume Analyzer demonstrate the effectiveness of the system in accurately matching resumes to job descriptions. During testing, a dataset of sample resumes and job postings was used to evaluate the matching algorithm's performance based on similarity scores and relevance.

Initial experiments involved preprocessing resumes and job descriptions, extracting features such as skills, experience, and education, and computing similarity using TF-IDF and cosine similarity. The system successfully ranked candidates based on how closely their resumes aligned with job requirements. In cases where resumes included clearly stated skills that matched job criteria, the system consistently provided high match scores (above 80%). Conversely, resumes with less relevant content received significantly lower scores, indicating good discriminatory power.

Quantitative evaluation showed that the top-ranked resumes matched over 75% of the keywords in the job descriptions, while lower-ranked ones matched less than 40%. Qualitative analysis confirmed that the system’s recommendations aligned with human judgment in most test cases.

These results suggest that the analyzer can significantly reduce manual screening time, offering a reliable, data-driven way to shortlist candidates. However, ongoing improvements, such as integrating semantic analysis via BERT, could further enhance matching accuracy



**figure 1.3**

**CHAPTER 6**

**CONCLUSION AND FUTURE SCOPE**

The Smart Resume Analyzer successfully demonstrates the potential of automating the resume screening process using Natural Language Processing (NLP) and Machine Learning (ML). The system can extract key information from resumes, compare it effectively with job descriptions, and rank candidates based on relevance. Experimental results show that the tool provides accurate and consistent match scores, significantly reducing manual effort and improving recruitment efficiency. The modular design also makes the system adaptable for different job domains and requirements.

**Future Scope:**

**1. Semantic Analysis Integration:** Enhancing the system with advanced language models like BERT or GPT can improve contextual understanding and semantic matching.

**2.Support for Multiple File Formats:** Extending input support for DOCX, scanned PDFs (OCR), and LinkedIn profiles

**3.Real-time Recommendations:** Suggesting resume improvements or missing skills based on job descriptions.

**4.Multi-language Support:** Adapting the system for resumes and job descriptions in various languages for global application.

**5.Recruiter Dashboard:** Building a full-fledged dashboard for recruiters with analytics, filters, and candidate comparison tools.

These improvements would make the Smart Resume Analyzer more powerful, user-friendly, and applicable in real-world hiring processes across industries.

**REFERENCES**

· 1. J. Smith, “Automated Resume Screening with AI,” IEEE Access, 2021.

· 2.· A. Kumar, “NLP Applications in HR,” Springer Journal of AI, 2022.

· **3.**· spaCy Documentation – <https://spacy.io/>

·  **4.**· **GitHub Repository:** <https://github.com/example/smart-resume-analyzer>

·