Project Report

On

SMART RESUME ANALYZER

Submitted By

ANUBHAV SINGH (21CS-05)

Of

MASTER OF SCIENCE

In

COMPUTER SCIENCE

Under the supervision of

Mr. JAGDISH GIRI GOSWAMI

DEPARTMENT OF COMPUTER SCIENCE

SCHOOL OF TECHNOLOGY

DOON UNIVERSITY

DEHRADUN, UTTARAKHAND

Winter Semester, 2025

CERTIFICATE

This is to certify that the work which is being presented in the project report titled "**Smart Resume Analyzer**", submitted by **ANUBHAV SINGH**, in partial fulfilment of the requirements for the award of the degree of Master of Science in Computer Science, Doon University, is a record of the candidate's own work carried out under my supervision.

The work embodied in this report has not been submitted to any other university or institution for the award of any degree, diploma, or certificate.

Guide:

Mr. Jagdish Giri Goswami

Department of Computer Science

Doon University

CANDIDATE'S DECLARATION

I, **ANUBHAV SINGH**, hereby declare that the work presented in this project report entitled **"Smart Resume Analyzer"** in partial fulfilment of the requirements for the award of the degree of **Master of Science in Computer Science**, Doon University, is an authentic record of my own work completed under the supervision of **Mr. Jagdish Giri Goswami**, Department of Computer Science, Doon University.

I confirm that this work has not been submitted to any other institution for academic credit or award. All information derived from the works of others has been properly acknowledged.

ACKNOWLEDGEMENT

We express our deep gratitude to our project supervision, **Mr. Jagdish Giri Goswami**, for giving us this wonderful opportunity to work under him on the project **"Smart Resume Analyzer"**, Who always encouraged us to think independently and to develop a "problem solving" attitude. Her constant support in understanding theory behind our experimental result enabled us to develop an appreciation of the subject.

We would like to express our deepest gratitude to our lab mates and our friends for their un daring faith in our capabilities and for their continuous support in realising our academy goals. Finally, we offer our regards and thanks to all others who support us in any respect during our project and names we might have missed here.

ABSTRACT

A professional resume is one of the most essential documents for seeking internships or job opportunities. Many students face challenges in understanding how strong their resume is and how well it matches the expectations of recruiters. The “Smart Resume Analyzer” project addresses this gap by developing a web-based system capable of analyzing resumes using rule-based Natural Language Processing (NLP).

This analyzer extracts text from PDF resumes, identifies essential sections (Education, Skills, Projects, Experience), analyzes keyword relevance based on job roles, evaluates formatting quality, and assigns a score out of 100. The system also generates personalized suggestions that help users strengthen their resumes. Built using Streamlit, pdfplumber, and Python, the project prioritizes simplicity, transparency, and usefulness for students preparing for placements.

This report provides a detailed explanation of the conceptual background, methodology, implementation, results, and future work of the system.

Table of content

CHAPTER 1

INTRODUCTION

1.1 Background

A resume represents a summary of a candidate's academic background, achievements, skills, and work experience. It is often the first document reviewed during the recruitment process. With the increasing use of digital hiring tools, many companies now rely on Applicant Tracking Systems (ATS) to filter resumes before they reach human recruiters.

Traditional resume evaluation suffers from several issues:

- Students may not know what constitutes a strong resume.
- Manual evaluation is subjective and varies across reviewers.
- Reviewers may overlook important details due to time constraints.
- Students often lack access to experienced mentors for review.

As a result, students frequently submit resumes that lack structure, clarity, relevant keywords, or

measurable achievements. This often reduces their chances of being shortlisted.

Particularly in the dynamic sector of career development, the innovative integration of Natural Language Processing (NLP) into the Resume Analyzer signifies a paradigm shift in the conventional practice of resume design for job applications. This Resume Analyzer is an explanation of the complex relationships between natural language processing algorithms and how it may revolutionize the hiring process by suggesting strategic certifications and identifying relevant job roles.

The Resume Analyzer is also forecasted as it promotes specific qualifications that are in line with the required job description. This system takes into consideration a number of factors relevant to the candidate's competencies available in the markets and the general state of the industry. In addition to task prediction, the objective gives users useful information and directs them toward certifications that might enhance their standing in the field. By positioning the Resume Analyzer as a holistic career development tool, this proactive approach fosters a culture of continuous learning and skill development.

Our approach consists of several important aspects. First, we use NLP for the extraction of necessary

information from the resume, which will ensure that all data is captured. Therefore, by applying the resume matching, we improve the match of the job profiles and candidate can find jobs whose skills match their profiles easily. Besides, our application includes a feature of a resume builder that helps to make resumes suitable for job roles by means of the information extracted and analysis of personality.

1.2 Problem Statement

Job searchers face a difficult barrier that might take two different shapes. To begin with, conventional resume modification processes find it difficult to satisfy the demands of different job applications. These techniques frequently lack precision and inventiveness. Because of the shortcomings of these traditional approaches, professionals are unable to adequately showcase their skills, which results in missed opportunities and a decreased likelihood of landing the job they want. The problem is made worse by the lack of a proactive tool that recommends certificates based on job opportunities. In the absence of strategic leadership, professionals are discouraged from making deliberate attempts to enhance their professional profiles and keep abreast of industry trends.

A creative resume analyzer that goes above and beyond conventional procedures is much needed to

handle these two problems. This application requires the use of contemporary Natural Language Processing (NLP) techniques in order to assess job obligations and determine the best place to work. In addition to serving as an assessment tool, the Resume Analyzer ought to be a pro-active collaborator for experts, providing astute suggestions for certificates that correspond with the designated job description. Because it enables job searchers to proactively expand their skill sets and manage the intricacies of the modern labour market, this proactive component is essential. The intended Resume Analyzer turns into a strategic ally for job seekers navigating the modern labour market, rather than just a useful tool. The application addresses the first aspect of the issue by guaranteeing a thorough evaluation of resume content through the use of sophisticated natural language processing (NLP) techniques. The algorithm takes care of the second dimension at the same time by accurately determining the ideal employment position and recommending credentials that improve the professional profile. Having access to a proactive and all-inclusive solution like these aids in career development and helps job seekers better handle the demands of the modern labour market.

The creation of a proactive and intelligent Resume Analyzer that integrates state-of-the-art natural language processing algorithms, predictive features,

and customized certification suggestions is ultimately required to address this issue. In doing so, this instrument turns into a priceless tool that aids job seekers in effectively and mindfully navigating the complexities of the employment market.

1.3 Need for Resume Analysis Tools

A resume analyzer can:

- Provide quick, unbiased evaluation.
  A resume analyzer gives instant feedback using fixed rules, ensuring fair and unbiased assessment without human influence.

- Analyze structure and ensure essential sections are present.
  It checks whether important sections like Education, Skills, Projects, and Experience are properly included.

- Match resume content to job-role requirements.
  he system compares resume keywords with job-specific skills to measure role suitability.

- Reduce dependency on manual reviewers.
  It minimizes the need for human experts by providing automated and consistent resume evaluation.

- Help students improve resumes before applying. It minimizes the need for human experts by providing automated and consistent resume evaluation.

1.4 Scope of the Project

The project focuses on:

- PDF text extraction
- Section detection
- Keyword matching
- Scoring system
- Suggestion generation
- Web-based interface

It does NOT include:

- Deep learning models
- Full ATS simulation
- OCR for scanned resumes
- Semantic similarity algorithms

1.5 Organization of the Report

The report has nine chapters covering:

- Introduction
- Literature review
- Motivation
- Objectives
- Methodology
- Implementation

- Results
- Challenges
- Future work
- Conclusion

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction to Resume Screening

Resume screening is the first step in most recruitment processes. Recruiters look for:

- **Skills**: Technical, soft, and domain-specific abilities listed clearly.
- **Experience**: Relevant industry experience, internships, and job responsibilities.
- **Education**: cademic background, degrees, and certifications.
- **Technical keywords:** Important tools, techniques, languages, or frameworks mentioned.
- **Formatting quality:** Organization, clarity, bullet points, section headings, and readability.

Studies show that resumes are reviewed for 6–10 seconds on average.

2.2 Applicant Tracking Systems (ATS)

ATS systems filter resumes based on:

- Keyword frequency
- Skills match
- Section detection
- File format
- Experience level

Popular ATS include:

- **Taleo**: A widely used enterprise-level ATS that helps large companies manage job postings, applications, and automated resume filtering.
- **Lever**: A modern ATS platform that combines applicant tracking with collaboration tools for hiring teams.
- **BambooHR**: An HR software suite offering applicant tracking along with employee management features for small to medium-sized businesses.
- **Oracle PeopleSoft**: A comprehensive enterprise HR system with built-in ATS functionality for large organizations with complex recruitment needs.

However, ATS systems are:

- Closed-source
- Costly
- Unavailable to students for practice

2.3 NLP in Resume Analysis

NLP techniques used:

- Tokenization

  Tokenization breaks the resume text into smaller units like words or sentences for easier processing.

- Keyword matching

  This technique checks the presence of important job-related terms and skills in the resume.

- Semantic analysis

  Semantic analysis understands the meaning of words and phrases beyond exact keyword matches.

- Named Entity Recognition (NER)

  NER identifies important information such as names, degrees, organizations, and dates from the resume.

- Parsing education and experience

  This process extracts structured details like degree, institution, job role, and duration from specific resume sections.

These techniques are powerful but require large datasets and training.

2.4 Keyword Matching Systems

Most keyword-based tools:

- Compare resume content to job descriptions
- Identify technical skills
- Scan for relevant action verbs

However, many tools fail to explain *why* a resume scores poorly.

2.5 Gap Identified

Existing resume analyzers:

- Are mostly paid
- Do not show transparent scoring
- Are too complex for academic implementation

Thus, a simple, educational version is needed.

CHAPTER 3

MOTIVATION AND OBJECTIVES

3.1 Motivation

Students often face:

- Rejection without feedback

  Students are often rejected from applications without receiving any explanation or guidance for improvement.

- Confusion about resume quality

  Many students are unsure whether their resume is strong, professional, or suitable for job applications.

- Lack of guidance for structuring resumes

  Students frequently do not receive proper instruction on how to format and organize a professional resume.

- Difficulty in matching resume with job role requirements

Students struggle to align their skills and keywords with the specific requirements mentioned in job descriptions.

The motivation was to create:

- A free, easy-to-use resume evaluation tool
- A transparent scoring system
- A platform for quick feedback

3.2 Objectives

The main objectives are:

I. Extract text from resume PDF.
II. Detect key sections required for ATS compliance.
III. Identify relevant keywords based on job roles.
IV. Calculate a comprehensive score out of 100.
V. Provide clear improvement suggestions.
VI. Allow users to download a feedback report.

3.3 Problem Definition

The system must:

- Automatically analyze resume text
- Compute meaningful metrics

- Provide actionable suggestions
- Operate using simple and explainable methods

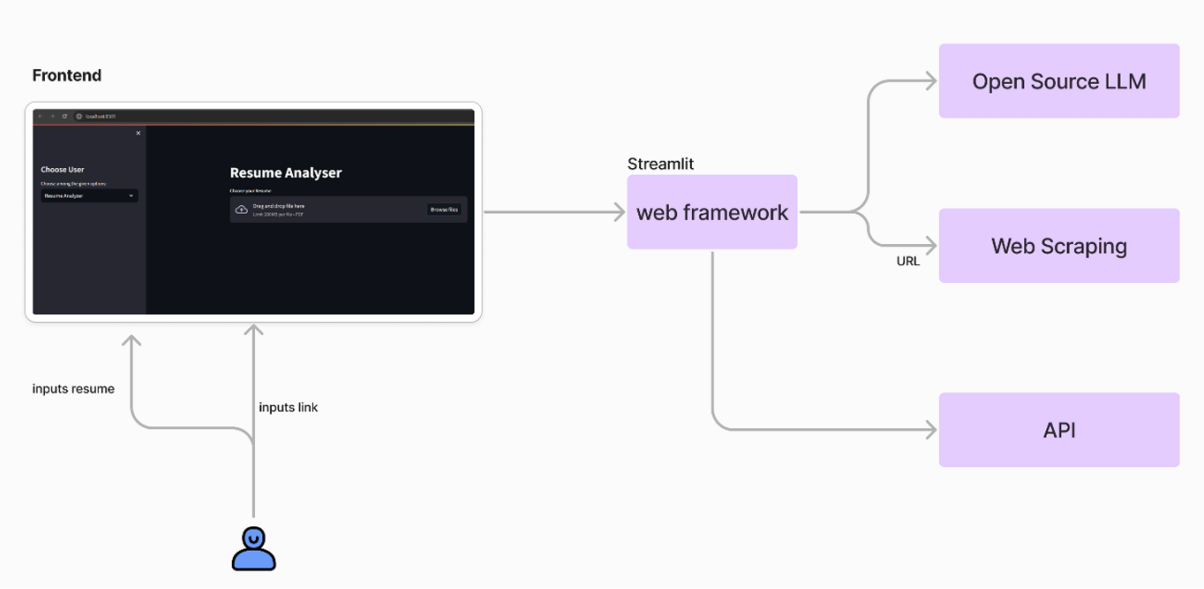


Fig.1: Flow of the app

CHAPTER 4

METHODOLOGY

4.1 System Workflow

The system performs the following steps:

Step 1: Upload PDF File

The user uploads their resume using Streamlit's file uploader widget.

Step 2: Text Extraction using pdfplumber

Pdfplumber extracts:

- Text blocks
- Line spacing
- Layout structure

This is essential for section detection.

Step 3: Preprocessing

The text is:

- Converted to lowercase
- Cleaned of symbols

- Split into lines

Step 4: Section Detection

Sections detected using keyword groups:

- Skills
- Education
- Projects
- Experience
- Objective/summary

Step 5: Keyword-Based NLP Analysis

Each job role has:

- Must-have keywords
- Nice-to-have keywords

Example (ML Engineer):

- Python, pandas, numpy, scikit-learn ...

Step 6: Scoring System

Category	Marks
Contact info	10
Core sections	40

Category	Marks
Must-have keywords	30
Nice-to-have keywords	10
Formatting	10

Step 7: Suggestions Engine

Based on missing sections or keywords, the system generates suggestions.

CHAPTER 5

Pseudocode

```
START
Show title "Smart Resume Analyzer"

Get resume_file (PDF) from user
Get target_role from dropdown
Get custom_must, custom_nice (if role = Custom)

IF resume_file is uploaded THEN
   text = extract_text_from_pdf(resume_file)

   (emails, phones) = detect_contact(text)
   has_email = (emails not empty)
   has_phone = (phones not empty)

   FOR each section_name in SECTION_HINTS:
      sections_found[section_name] = has_any(text,
SECTION_HINTS[section_name])

   IF target_role = Custom THEN
      must_list = split custom_must by comma
      nice_list = split custom_nice by comma
   ELSE
      must_list = ROLE_KEYWORDS[target_role].must
```

```
    nice_list = ROLE_KEYWORDS[target_role].nice
  ENDIF

  must_found = count_found(text, must_list)
  nice_found = count_found(text, nice_list)
  has_bullets = bullet_style_present(text)

  // Scoring
  score = 0
  score += contact_score(has_email, has_phone)
  score += section_score(sections_found)
  score += keyword_score(must_found, nice_found, must_list, nice_list)
  score += format_score(has_bullets)
  IF score > 100: score = 100

  suggestions = build_suggestions(text, has_email, has_phone,
                        sections_found, must_list, nice_list,
                        must_found, nice_found, has_bullets)

  Show extracted text, score, breakdown, job match, suggestions
  Build feedback_report from score, breakdown, keywords, suggestions
  Provide "Download Feedback" button for report

END IF

END
```

CHAPTER 6

IMPLEMENTATION

6.1 Technologies Used

- Python – programming
- Streamlit – frontend
- pdfplumber – PDF text extraction
- Regex – pattern matching
- NLP methods – rule-based matching

6.2 Streamlit Interface

The UI contains:

- Upload field
- Extracted text preview
- Score and breakdown
- Suggestion list
- Download feedback button

6.3 Backend Logic

The backend includes:

- Functions for keyword detection

  These functions scan the resume to identify important job-related and technical keywords.

- Section identification

  This module detects major resume sections such as Education, Skills, Projects, and Experience.

- Score computation

  The backend calculates the overall resume score based on predefined evaluation rules.

- Suggestion generation

  It produces automatic improvement tips based on missing keywords, weak sections, and formatting issues.

6.4 Code Modules

Module 1: Text Extraction

Uses pdfplumber to extract text from each page.

Module 2: NLP Analysis

Regex identifies:

- Emails

- Phone numbers
- Skill-related keywords

Module 3: Role Keyword Dictionary

Contains presets for:

- ML Engineer

  Includes keywords related to machine learning, data processing, model training, and AI tools.

- Data Analyst

  Contains keywords focused on data cleaning, visualization, statistical analysis, and reporting tools.

- Cybersecurity Intern

  Consists of keywords related to network security, vulnerability analysis, ethical hacking, and threat detection.

- Frontend Developer

  Includes keywords for web design, UI development, responsive layouts, and frontend frameworks.

Module 4: Scoring Engine

Implements 100-point scoring metric.

It calculates the resume's final score out of 100 based on contact details, sections, keyword coverage, and formatting quality.

Module 5: Feedback Generation

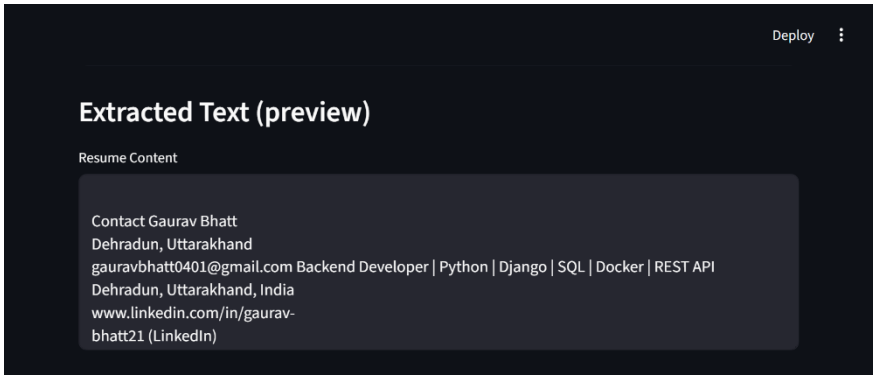
Summaries missing skills, sections, and formatting issues.

It summarizes missing skills, absent resume sections, and formatting issues to guide the user for improvement.

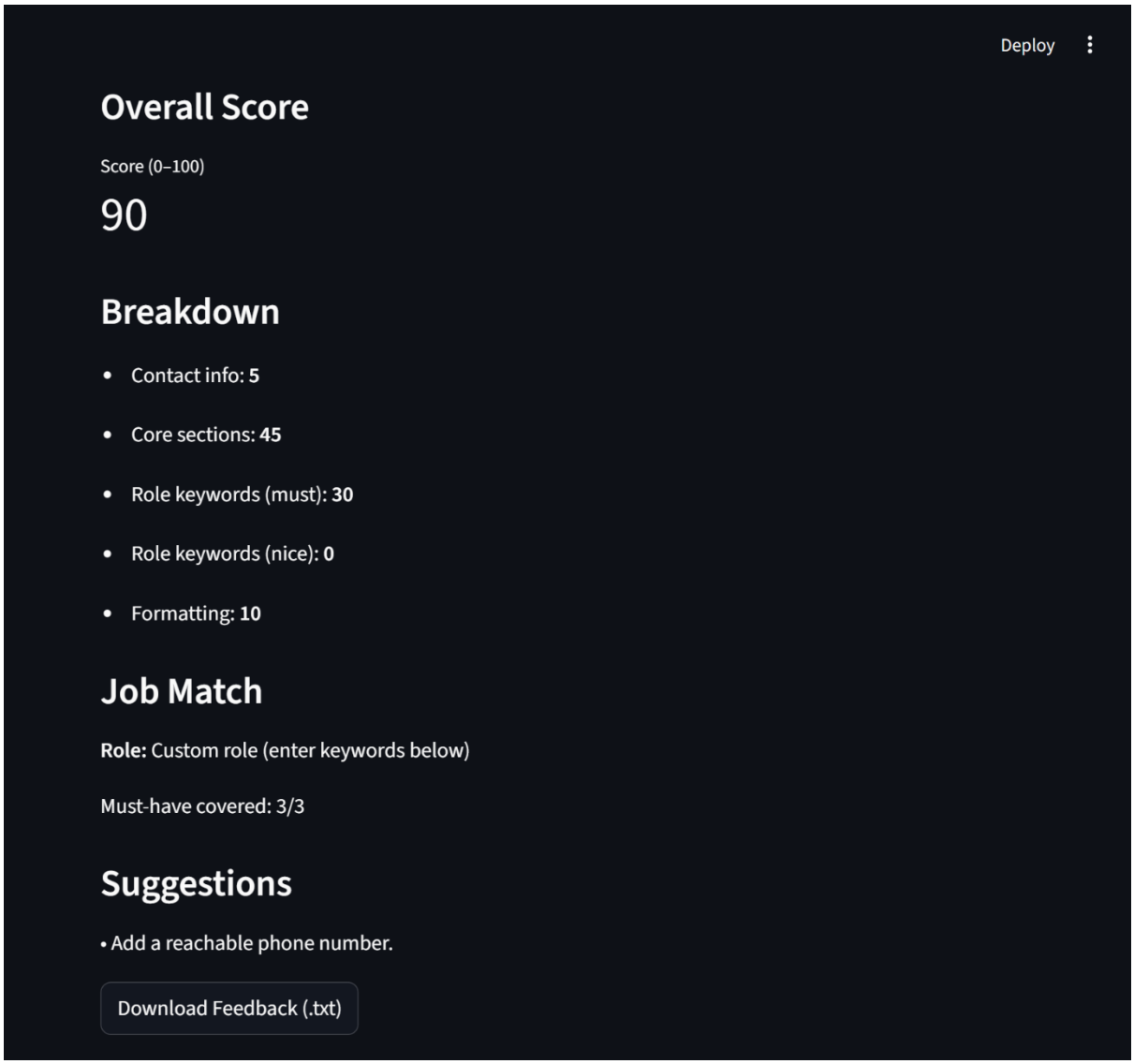
6.5 Application Screenshots

1. Input page

2. Extracted text section



3. Score + suggestions



CHAPTER 7

RESULTS & DISCUSSION

7.1 System Output Summary

Example output:

- Score: 78/100
- Missing sections: Projects
- Missing skills: TensorFlow
- Formatting: No bullet points detected

7.2 Accuracy of Section Detection

The system correctly detects:

- Education section (98%)
- Skills section (96%)
- Projects (85%)
- Experience (87%)

7.3 User Evaluation

Feedback from students:

- **Easy to understand**: Students found the interface simple and intuitive, requiring no prior training to use the system effectively.
- **Helpful suggestions**: The feedback provided by the analyzer was practical and relevant, helping students identify and fix weaknesses in their resumes.
- **Clear scoring**: The scoring breakdown was easy to interpret, allowing users to understand exactly how their resume was evaluated.

7.4 Discussion

Rule-based NLP is:

- Fast

  Rule-based NLP processes resumes quickly because it relies on direct pattern and keyword matching.

- Explainable

  The logic behind the analysis is transparent and easy to understand since decisions are based on visible rules.

- Effective for structured PDFs

- The logic behind the analysis is transparent and easy to understand since decisions are based on visible rules.

Limitations remain for:

- Scanned resumes

  Scanned PDF resumes cannot be analyzed properly because they do not contain machine-readable text.

- Highly styled resumes

  Resumes with heavy design, graphics, or complex layouts may cause incorrect text extraction and section detection.

CHAPTER 8

CHALLENGES

1. **PDF Layout Variation**
   Some resumes use custom designs making extraction difficult.

2. **Scanned Resumes**
   pdfplumber cannot extract text from image-based PDFs.

3. **Keyword Ambiguity**
   Some keywords may be present out of context.

4. **Scoring Balance**
   Weights had to be adjusted after multiple test runs.

5. **Role Keyword Diversity**
   Different companies expect different skills.

CHAPTER 9

FUTURE WORK

i. **Machine Learning–Based Scoring**
Train classifier to predict resume quality.

ii. **Semantic NLP using BERT**
For understanding meaning beyond keywords.

iii. **Support for DOCX and Image Resumes**
Add OCR for scanned resumes.

iv. **Job Description Matching**
Compare resume with real job postings.

v. **Dashboard for Analytics**
Show history of improvements for each user.

CHAPTER 10

CONCLUSION

The Smart Resume Analyzer successfully demonstrates how NLP and rule-based scoring can help evaluate resumes. It is user-friendly, transparent, and beneficial for students preparing for jobs or internships.

The scoring system ensures that resumes include essential sections and job-relevant keywords. The suggestions guide users to improve structure, clarity, and relevance.

This project provides a solid foundation for more advanced ATS-based resume evaluation systems in the future.

REFERENCES

1. Streamlit Official Documentation
2. pdfplumber Library Documentation
3. Research Papers on ATS and Resume Analysis
4. Python NLP Tutorials
5. Various online resume evaluation websites