Project Report: Intelligence Resume Screening Using AI

---

1. Project Title

Intelligence Resume Screening Using AI

---

2. Team Members

- Bhumit Tanwar: Introduction & Problem Statement
- Pratyush Jha: Backend Technologies
- Tarun: Frontend Technologies
- Sohail Khan: System Working & Conclusion

---

3. Introduction

In the current job market, recruiters often face the challenge of screening a large volume of resumes. Manual screening is time-consuming, prone to errors, and often subjective. Our project, the AI-Based Resume Analyzer, aims to automate resume screening by parsing resumes, extracting structured information, and matching candidates to suitable job profiles using artificial intelligence techniques.

---

4. Objective

The main objective of this project is to:

- Automate resume screening.
- Extract and organize key information from resumes.
- Match candidate profiles to the most relevant job roles.
- Provide recruiters with quick, unbiased, and accurate candidate recommendations.

---

5. Technologies Used

Backend

- **Python:** Core programming language.
- **pdfminer3:** Used for extracting text from PDF resumes.
- **pyresparser:** Extracts structured information like name, skills, email, and experience.
- **pandas:** Data processing and management.
- **pymysql:** Connection and communication with a MySQL database to store resumes and job data.

Frontend

- **Streamlit:** For building a fast, interactive web application where users can upload resumes.
- **streamlit-tags:** Adds user-friendly tags and labels in the Streamlit app.
- **Plotly:** Creates visualizations like skill-match graphs and charts.
- **Pillow:** Processes and optimizes images if resumes contain graphics.

Other Tools

- **Uvicorn:** ASGI server for running the FastAPI backend (if required).
- **MySQL Database:** Storage for parsed resume data and job descriptions.

---

6. System Workflow

1. **Resume Upload:** The user uploads a PDF or DOCX file through the Streamlit web interface.
2. **Resume Parsing:**
    - The resume is parsed using pdfminer3 and pyresparser.
    - Key fields like name, email, phone number, skills, and experience are extracted.
3. **Data Storage:**
    - Parsed resume data is saved in a MySQL database.
4. **Job Matching:**
    - The system compares extracted skills and experience with available job descriptions.
    - Using NLP and cosine similarity, a match score is calculated.
5. **Result Display:**
    - The system displays the top matching jobs and match percentages using Plotly charts.
    - Recruiters and candidates can view the best-fit job recommendations.

---

7. Features

- Automated parsing of resumes.
- Skill and experience extraction.
- Job-candidate matching using AI.
- Interactive web interface for easy uploads and results.
- Visual representation of matching scores.
- Database storage for efficient data retrieval.

---

8. Benefits

- Saves recruiter time.
- Reduces human bias.
- Increases recruitment efficiency.
- Improves candidate-job matching accuracy.
- Enhances user experience through a simple web application.

---

9. Conclusion

The AI-Based Resume Analyzer provides an intelligent and automated solution for recruiters and job seekers alike. By leveraging AI and modern web technologies, it transforms traditional recruitment processes, making them more efficient, unbiased, and scalable.

This project demonstrates the potential of AI in solving real-world human resource challenges.

---

10. Future Scope

- Integration with LinkedIn profiles and online job portals.
- Multilingual resume parsing.
- Sentiment analysis of experience descriptions.
- Predictive analytics for career path recommendations.

---

11. References

- pdfminer3 documentation
- pyresparser documentation
- Streamlit documentation
- Plotly official website
- scikit-learn for similarity matching concepts