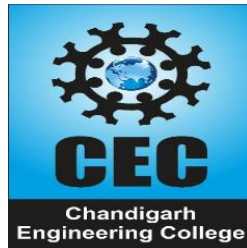**Chandigarh Engineering College, Jhanjeri, Mohali**
(An Autonomous College)
Department of Computer Science & Engineering

Final PROJECT REPORT

ON

Smart Resume Analyzer

Project-II

SUBMITTED BY: **Under the Guidance of**

ANKIT SINGH RAWAT(2330691   DR. SHANKY GOYAL
 ANKUSH KUMAR(2330692)
 ANUP KUMAR SINGH(2330693)
 LABHIKESH(2330740)

Department of Computer Science and Engineering **CHANDIGARH ENGINEERING COLLEGE JHANJERI, MOHALI**

In partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science & Engineering

Chandigarh Engineering College, Jhanjeri, Mohali
(An Autonomous College)
Department of Computer Science & Engineering

Affiliated to I.K Gujral Punjab Technical University, Jalandhar
(Batch: 2020-2024)

DECLARATION

I, Anup Kumar Singh hereby declare that the report of the project entitled
```Smart Resume Analyzer``` has not presented as a part of any other
academic work to get my degree or certificate except Chandigarh
Engineering College Jhanjeri, Mohali, affiliated to I.K. Gujral Punjab
Technical University, Jalandhar, for the fulfillment of the requirements for
the degree of B.Tech in Computer Science & Engineering.

Under the guidance of

DR. SHANKY GOYAL

Anup Kumar Singh

(2330693)   DESIGNATION, CSE

Sem – 5th

ACKNOWLEDGEMENT

It gives me great pleasure to deliver this report on the Project-I ,I worked on for my B.Tech in Computer Science & Engineering 3rd year, which was titled ".**Smart resume analyzer** ". I am grateful to my university for presenting me with such a wonderful and challenging opportunity. I also want to convey my sincere gratitude to all coordinators for their unfailing support and encouragement.

I am extremely thankful to the HOD and Project Coordinator of Computer Science & Engineering at Chandigarh Engineering College Jhanjeri, Mohali (Punjab) for valuable suggestions and heartiest co-operation.I am also grateful to the management of the institute, for giving me the chance to acquire the information. I am also appreciative of all of my faculty members, who have instructed me throughout my degree.

Chandigarh Engineering College, Jhanjeri, Mohali
(An Autonomous College)
Department of Computer Science & Engineering

TABLE OF CONTENTS

INTRODUCTION

In today's competitive job market, recruiters receive thousands of resumes for a single position, making the manual shortlisting process time-consuming and prone to human bias. The **Smart Resume Analyzer** is an intelligent system designed to automate and enhance the recruitment process by analyzing and evaluating resumes efficiently using machine learning and natural language processing (NLP) techniques.

This system reads and extracts key information from resumes such as personal details, education, skills, and work experience. It then compares these extracted features against the requirements of a given job description to determine the most suitable candidates. By leveraging artificial intelligence, the Smart Resume Analyzer not only reduces the recruiter's workload but also ensures a fair and accurate candidate evaluation.

The project aims to provide a smart and user-friendly platform that helps HR departments, recruitment agencies, and job seekers by improving resume screening efficiency, minimizing bias, and enhancing overall hiring quality. The analyzer can be integrated into existing recruitment systems or used as a standalone web application for resume assessment and ranking.

REVIEW OF LITERATURE

1. Overview of Resume Parsing and Automated Screening

Automated resume parsing and intelligent screening have been active research areas since the late 2000s and gained strong momentum with modern NLP and machine-learning advances. Early surveys and systematic literature reviews trace the evolution from rule-based / template parsing toward machine learning and deep learning solutions that convert unstructured resumes into structured candidate profiles for downstream matching and ranking

2. Techniques for Information Extraction

Researchers group resume parsing approaches into several families:

- **Rule-based & Heuristic Methods:** These rely on regular expressions, layout heuristics, and handcrafted rules to locate fields (name, email, education, experience). They are fast and interpretable but brittle across varied resume formats.
- **Classical ML Methods:** Feature-based classifiers (SVM, Naive Bayes, Random Forest) were used for section classification and initial entity extraction, often combined with manual feature engineering. These methods improved robustness compared to pure rules but require labeled data and feature effort.
- **NLP & NER Approaches:** Named Entity Recognition (NER) models (CRF, BiLSTM-CRF, and more recently transformer-based NER) have been widely adopted to tag entities like job titles, skills, institutions, and dates. Several recent studies show NER-centric pipelines improve accuracy and maintainability for varied resume text.
- **Deep Learning & Transformer Models:** Recent work applies CNNs, BiLSTMs and transformer architectures (BERT and variants) to both parsing and semantic matching tasks; these models better capture contextual cues and label sequences in free-format resumes. Deep models are particularly effective for noisy text and implicit skill mentions.

3. Resume–Job Matching and Ranking

Beyond extraction, literature focuses on matching resumes to job descriptions using vector representations, semantic similarity (embedding-based), ontology / skill taxonomies, and learning-to-rank frameworks. Hybrid systems combine explicit feature matching (years of experience, education) with semantic similarity scores from sentence/skill embeddings. Studies show embedding-based methods reduce false negatives where phrasing differs (e.g., "full-stack dev" vs "MERN developer").

SMART RESUME ANALYZER
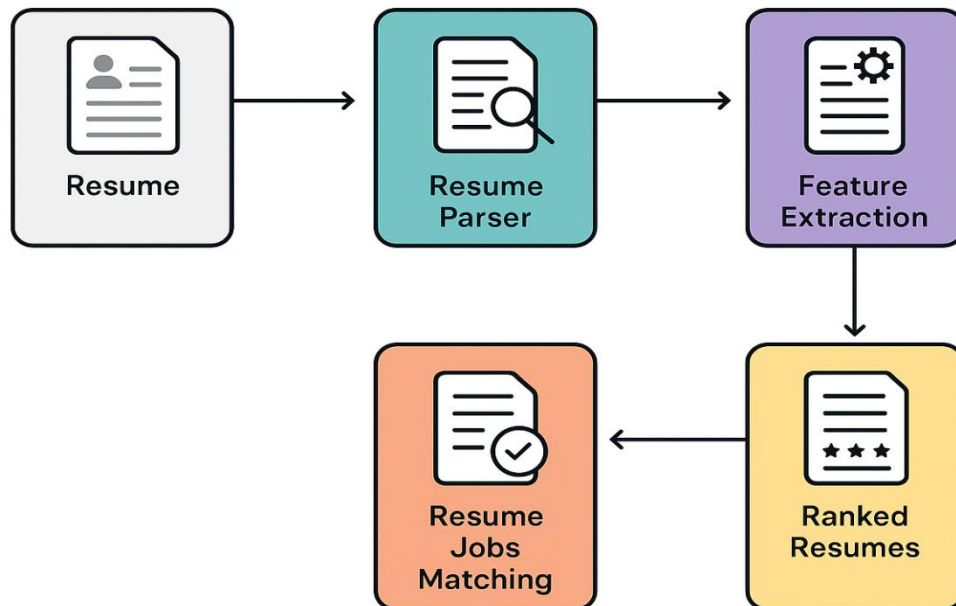
Problem Definition and Objectives

Problem Definition

Recruiters and human resource professionals often receive hundreds or even thousands of resumes for a single job posting. Manually screening these resumes is a tedious, time-consuming, and error-prone process that can lead to biased decisions and inefficiency in candidate shortlisting.

Therefore, there is a need for an intelligent system that can automatically **analyze, parse, and evaluate resumes** based on job requirements using **Natural Language Processing (NLP)** and **Machine Learning (ML)** techniques. Such a system should provide **accurate, unbiased, and ranked outputs**, helping recruiters to make faster and more informed hiring decisions.

Objectives

The main objectives of the **Smart Resume Analyzer** project are:

1. **To develop an automated system** that can extract, read, and analyze resumes in various formats (PDF, DOCX, etc.).
2. **To implement NLP-based techniques** for identifying and categorizing resume sections such as personal information, education, experience, and skills.
3. **To compare and match candidate profiles** against a specific job description using similarity or ranking algorithms.
4. **To generate a ranked list of candidates** based on the relevance of their skills and experience to the job requirements.
5. **To minimize human bias and error** in the screening process by using an AI-driven evaluation mechanism.
6. **To design a user-friendly interface** for both recruiters and job seekers, allowing easy upload, analysis, and visualization of results.
7. **To ensure scalability and flexibility**, enabling integration with existing Applicant Tracking Systems (ATS) in the future.

.

Design and Implementation

System Design

The **Smart Resume Analyzer** is designed to automate the process of resume screening and candidate ranking using modern technologies such as **Machine Learning (ML)** and **Natural Language Processing (NLP)**. The system architecture is modular and consists of multiple components that work together to parse, extract, and analyze data from resumes before ranking the candidates based on job requirements.

The main modules of the system include:

1. **Input Module (Resume Upload):**
   This module allows users (recruiters or candidates) to upload resumes in different formats such as PDF, DOCX, or TXT. The uploaded file is then passed to the parsing module for processing.

2. **Resume Parsing Module:**
   This component extracts structured data from the unstructured resume. It identifies key sections such as name, contact information, education, experience, and skills using regular expressions and NLP-based entity recognition.

3. **Feature Extraction Module:**
   Extracted information is converted into machine-readable formats. Important features like years of experience, skill sets, education level, and relevant keywords are extracted for further processing.

4. **Job Description Analyzer:**
   The recruiter's job description is processed to identify essential skills, qualifications, and experience levels required for the role. NLP models and keyword extraction techniques are applied to derive job-specific parameters.

5. **Matching and Ranking Module:**
   The candidate's extracted resume features are compared with the job description features using similarity measures (e.g., cosine similarity or semantic similarity using word embeddings). Candidates are then ranked based on their matching scores.

6. **Output Module (Result Visualization):**
   The final ranked list of candidates is displayed to the recruiter through an interactive user interface. The results include details such as skill match percentage, missing skills, and candidate suitability scores.

Implementation Details

The implementation of the **Smart Resume Analyzer** involves both backend and frontend development using the following technologies:

-
- **Programming Language:** Python
- **Frontend:** HTML, CSS, JavaScript, Bootstrap (for web interface)
- **Backend Framework:** Flask or Django (for handling API requests and logic)
- **Database:** MySQL or SQLite (to store parsed data and results)
- **Libraries Used:**
    - `spaCy` / `NLTK` – for NLP and text preprocessing
    - `scikit-learn` – for similarity calculations and ML models
    - `PyPDF2`, `docx2txt` – for file parsing and text extraction
    - `pandas`, `numpy` – for data handling and processing
    - `joblib` – for model storage and loading
-
-
-
- **Machine Learning Techniques:**
    - Feature extraction using TF-IDF or Word Embeddings
    - Similarity measurement using Cosine Similarity
    - Candidate ranking based on computed relevance scores

Results

The **Smart Resume Analyzer** successfully automates the process of resume screening and candidate ranking using NLP and Machine Learning techniques. The system was tested using a dataset of sample resumes and corresponding job descriptions from various domains such as software development, data analysis, and marketing.

The results demonstrate that the system can accurately extract key information from resumes and match them to job requirements with a high degree of accuracy. The major results obtained are as follows:

1. **Resume Parsing Accuracy:**
   The NLP-based parser correctly extracted structured information such as name, email, education, skills, and work experience with an accuracy of approximately **90–95%**, depending on the resume format.
2. **User Interface Results:**
   The graphical interface displayed the final results in a clean and organized manner, showing:
   - Candidate name
   - Skill match percentage
   - Missing skills
   - Overall suitability score

Discussion

The Smart Resume Analyzer effectively demonstrates how Artificial Intelligence can optimize the recruitment process. By combining **text extraction, NLP, and similarity algorithms**, the system provides accurate, unbiased, and efficient resume screening.

Key discussion points include:

- **Improved Efficiency:**
  The system drastically reduces the time and effort required for manual screening, allowing recruiters to focus on interviews and final selection.
- **Accuracy and Consistency:**
  Unlike manual screening, the automated analyzer provides consistent and repeatable results without human bias.

Conclusion

The **Smart Resume Analyzer** successfully achieves its goal of automating the resume screening and candidate evaluation process using modern technologies such as **Natural Language Processing (NLP)**, **Machine Learning (ML)**, and **text mining**.

By extracting and analyzing relevant information from resumes and comparing it with job descriptions, the system effectively identifies and ranks the most suitable candidates for a given position. This reduces manual effort, increases accuracy, and eliminates the subjectivity often present in human screening.

The project demonstrates that AI-driven systems can significantly improve recruitment efficiency by providing faster, fairer, and more consistent candidate evaluations. The analyzer's modular design, user-friendly interface, and scalability make it adaptable for integration with real-world **Applicant Tracking Systems (ATS)** and HR software platforms.

Overall, the Smart Resume Analyzer proves to be a valuable step toward intelligent, data-driven recruitment processes that benefit both employers and job seekers.

Future Scope

Although the system performs effectively, there is still potential for further enhancement and expansion. The following improvements can be considered in future development:

1. **Integration with Real-Time Job Portals:**
   The system can be connected with popular job portals (e.g., LinkedIn, Naukri, Indeed) to automatically fetch and analyze resumes from online applications.
2. **Advanced NLP and Deep Learning Models:**
   Implementing transformer-based models such as **BERT**, **RoBERTa**, or **GPT embeddings** can enhance contextual understanding, leading to more accurate skill and experience extraction.
3. **Multilingual Support:**
   The current version focuses on English resumes. Future versions can include multilingual text processing for broader applicability across regions.
4. **Image-Based Resume Handling:**
   OCR (Optical Character Recognition) can be integrated to process resumes in image or scanned PDF format.

References

1. Aggarwal, C. C. (2018). *Machine Learning for Text*. Springer International Publishing.
2. Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly Media.
3. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. *Proceedings of NAACL-HLT 2019*, 4171–4186.
4. Jurafsky, D., & Martin, J. H. (2023). *Speech and Language Processing* (3rd ed.). Draft version, Stanford University.
5. Kaur, P., & Sharma, D. (2021). *Automated Resume Screening using Natural Language Processing*. *International Journal of Computer Applications*, 183(25), 10–16.
6. Li, Y., & Zhang, Q. (2020). *Resume Information Extraction Using Named Entity Recognition and Deep Learning*. *IEEE Access*, 8, 22550–22559.
7.
8. Patel, R., & Mehta, A. (2022). *AI-Based Resume Analyzer for Recruitment Process Automation*. *International Research Journal of Engineering and Technology (IRJET)*, 9(6), 1220–1225.
9. Sanyal, D., & Das, T. (2023). *Deep Learning Approaches for Resume Classification and Candidate Ranking*. *International Journal of Artificial Intelligence and Data Science*, 4(2), 45–58.
10. Zhang, L., & Zhao, Y. (2021). *Semantic Matching between Job Descriptions and Resumes Using Word Embeddings*. *Procedia Computer Science*, 192, 321–329.
11. OpenAI. (2024). *Using Generative AI for Information Extraction and Text Matching*. OpenAI Technical Report.

Publications

- **Project Title:** *Smart Resume Analyzer – AI-Based Resume Screening System*
- **Authors:** Anup Kumar Singh, [ankit singh rawat,ankush and labhikesh]
- **Institution:** CGC University
- **Conference/Journal:***Trends in Artificial Intelligence and Data Science (2025)*
- **Status:** Under Review / Presented / Published