

A Major Project Report

On

INTELLIGENT RESUME TRACKING SYSTEM

Submitted to CMREC (UGC Autonomous), Affiliated to JNTUH

In Partial Fulfillment of the requirements for the Award of Degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING (AI&ML)

Submitted By

C ASHRITH REDDY

208R1A66D1

Under the Esteemed guidance of

Dr. Rajesh Tiwari

Professor, Department of CSE (AI & ML)



Department of Computer Science & Engineering (AI&ML)

CMR ENGINEERING COLLEGE

UGC AUTONOMOUS

(Approved by AICTE, NEW DELHI, Affiliated to JNTU, Hyderabad, Kandlakoya,

Medchal Road, R.R. Dist. Hyderabad-501 401)

2023-2024

CMR ENGINEERING COLLEGE
UGC AUTONOMOUS

(Accredited by NBA, Approved by AICTE NEW DELHI, Affiliated to JNTU, Hyderabad)

Kandlakoya, Medchal Road, Hyderabad-501 401

Department of Computer Science & Engineering (AI & ML)



CERTIFICATE

This is to certify that the project entitled “**Intelligent Resume Tracking System**” is a bonafide work carried out by

C ASHRITH REDDY

208R1A66D1

in partial fulfillment of the requirement for the award of the degree of **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE AND ENGINEERING (AI&ML)** from CMR Engineering College, under our guidance and supervision.

The results presented in this project have been verified and are found to be satisfactory. The results embodied in this project have not been submitted to any other university for the award of any other degree or diploma.

Internal Guide
Dr. Rajesh Tiwari
Professor
CSE (AI&ML)

Major Project Coordinator
Mr. G. Venkateswarlu
Assistant Professor
CSE (AI&ML)

Head of the Department
Dr. M. Kumara Swamy
Professor & HOD
CSE (AI&ML)

External Examiner

DECLARATION

This is to certify that the work reported in the present Major project entitled “**Intelligent Resume Tracking System**” is a record of bonafide work done by us in the Department of Computer Science and Engineering (AI&ML) , CMR Engineering College. The reports are based on the project work done entirely by us and not copied from any other source. We submit our project for further development by any interested students who share similar interests to improve the project in the future.

The results embodied in this Major project report have not been submitted to any other University or Institute for the award of any degree or diploma to the best of our knowledge and belief.

C ASHRITH REDDY

208R1A66D1

ACKNOWLEDGMENT

We are extremely grateful to **Dr. A. Srinivasula Reddy**, Principal and **Dr. M. Kumara Swamy**, HOD, **Department of CSE (AI & ML), CMR Engineering College** for their constant support.

We are extremely thankful to **Dr. Rajesh Tiwari**, Professor, Internal Guide, Department of CSE (AI & ML), for his constant guidance, encouragement and moral support throughout the project.

We will be failing in duty if we do not acknowledge with grateful thanks to the authors of the references and other literatures referred in this Project.

We thank **Mr. G. Venkateswarlu**, Major Project Coordinator for his constant support in carrying out the project activities and reviews.

We express our thanks to all staff members and friends for all the help and co-ordination extended in bringing out this project successfully in time.

Finally, We are very much thankful to our parents who guided us for every step.

C ASHRITH REDDY

208R1A66D1

CONTENTS

TOPIC	PAGE NO.
ABSTRACT	i
LIST OF FIGURES	ii
1. INTRODUCTION	
1.1. Introduction & Objectives	1
1.2. Purpose of the Project	4
1.3. Existing System & Disadvantages	5
1.4. Proposed System with Features	8
2. LITERATURE SURVEY	18
3. SOFTWARE REQUIREMENT ANALYSIS	
3.1. Problem Specification	22
3.2. Modules & their Functionalities	22
3.3. Functional Requirements	25
3.4. Non-Functional Requirements	29
3.5. Feasibility Study	31
4. SOFTWARE & HARDWARE REQUIREMENTS	
4.1. Software requirements	34
4.2. Hardware requirements	34
5. SYSTEM ARCHITECTURE	
5.1. System Design	36
5.2. Data Flow Diagram	40
6. IMPLEMENTATION AND CODING	
6.1. IMPLEMENTATION	40
6.1.1. PYTHON	40
6.1.2. NATURAL LANGUAGE PROCESSING	51
6.1.3. STREAMLIT	53
6.1.4. PYPDF2	55
6.2. CODE	57
7. OUTPUT SCREENS	60
8. CONCLUSION	63
9. FUTURE ENHANCEMENTS	65
10. PUBLICATION	67
11. BIBLIOGRAPHY AND REFERENCES	67

ABSTRACT

ABSTRACT

The Intelligent Resume Tracking System (IRTS) represents a groundbreaking initiative aimed at revolutionizing traditional methods of resume evaluation and tracking. Central to its design is the incorporation of cutting-edge artificial intelligence (AI) technology, specifically powered by Google's Gemini API. This strategic integration empowers the system to provide a comprehensive solution tailored to the evolving needs of contemporary job seekers. IRTS boasts a meticulously curated suite of features, meticulously crafted to streamline the process of resume optimization. Through AI-powered resume evaluation, the system meticulously examines resumes, identifying crucial qualifications, skills, and experiences pertinent to specific job roles. This advanced analysis is further enriched by semantic analysis techniques, ensuring a nuanced comprehension of resume content.

A cornerstone functionality of IRTS is its capacity to accurately match candidate resumes with job descriptions. Leveraging machine learning algorithms, the system compares resumes against job requirements, assigning a percentage match to signify alignment. This capability enables job seekers to refine their resumes effectively, enhancing their prospects of securing suitable employment opportunities. Keyword analysis serves as another pivotal aspect of IRTS, empowering users to pinpoint and optimize relevant keywords in their resumes. By scrutinizing both resumes and job descriptions, the system spotlights essential skills and experiences sought by employers, enabling users to tailor their resumes accordingly.

Real-time feedback mechanisms ensure that users receive timely insights into their resume optimization endeavors. By furnishing instantaneous feedback on resume content and alignment with job descriptions, the system empowers users to make well-informed decisions and iterate on their resumes effectively.

Keywords:

- Artificial Intelligence.
- Natural Language processing.
- Resume optimization.

LIST OF FIGURES

FIG.NO	DESCRIPTION	PAGE NO.
1.4	Block Diagram of the proposed system	15
5.1	System Design	36
5.2	Data Flow Diagram	43
7.1	Output Screen - 1	60
7.2	Output Screen – 2	60
7.3	Output Screen - 3	61

CHAPTER-1

INTRODUCTION

1. INTRODUCTION

1.1. INTRODUCTION & OBJECTIVES

The Intelligent Resume Tracking System (IRTS) represents a significant leap forward in the realm of recruitment technology. In an era where organizations receive an ever-increasing number of job applications, efficient candidate screening is essential. IRTS offers a solution to this challenge by harnessing the power of advanced technologies such as Natural Language Processing (NLP) and machine learning to automate the processing and matching of resumes.

Recruitment processes often grapple with inefficiencies and delays, particularly in the initial stages of resume screening. With traditional methods, recruiters spend valuable time manually parsing through resumes, leading to bottlenecks in the hiring process. IRTS aims to alleviate these challenges by streamlining resume processing, enabling recruiters to focus on evaluating qualified candidates rather than sifting through piles of applications.

At the heart of IRTS lies a sophisticated infrastructure powered by state-of-the-art NLP algorithms and machine learning models. These technologies enable the system to extract key information from resumes, such as skills, experience, and qualifications, with unparalleled accuracy and efficiency. By automating these tasks, IRTS accelerates the recruitment process while minimizing the risk of human error.

By automating mundane tasks like resume parsing and matching, IRTS empowers recruiters to devote more time and resources to strategic decision-making. With its ability to process resumes at scale and identify top candidates quickly, IRTS significantly enhances recruitment efficiency. This not only expedites the hiring process but also improves the overall quality of candidate selection.

In today's fast-paced business environment, organizations must stay agile and responsive to changing market dynamics. By implementing IRTS, companies can gain a competitive edge in talent acquisition, ensuring they attract and retain top talent effectively.

OBJECTIVES

The following objectives outline the key functionalities and desired outcomes of the Intelligent Resume Tracking System. These objectives aim to create a robust, efficient, and user-friendly system that streamlines the recruitment process for both organizations and job seekers.:

- 1. Automate Resume Processing:** This objective entails developing algorithms capable of automatically extracting essential information from resumes, such as contact details, education history, work experience, skills, and certifications. By automating this process, the system reduces the need for manual data entry and ensures accuracy in capturing relevant information. This automation enhances the efficiency of resume processing, enabling recruiters to handle large volumes of resumes more effectively and allocate their time more efficiently.
- 2. Efficient Candidate-Job Matching:** The goal here is to implement search and match algorithms that can efficiently pair candidates with suitable job openings based on various factors such as skills, experience level, and educational background. By leveraging advanced algorithms, the system aims to identify candidates whose qualifications closely align with the requirements of specific roles. This efficient candidate-job matching process improves the quality of candidate recommendations, increases the likelihood of successful job placements, and reduces the time-to-fill vacant positions.
- 3. Improve Recruitment Efficiency:** This objective focuses on streamlining the recruitment process by reducing manual effort and automating repetitive tasks associated with resume screening and candidate selection. By leveraging automation and intelligent algorithms, the system aims to accelerate the hiring process, minimize administrative overhead, and optimize resource allocation. Improving recruitment efficiency enables organizations to shorten time-to-hire metrics, enhance overall productivity, and achieve cost savings.
- 4. Enhance Candidate Quality:** The objective here is to ensure that the system presents hiring managers with candidates who possess the most relevant qualifications and experiences for each job opening. By using advanced candidate screening and ranking algorithms, the system aims to identify high-quality candidates who closely match the requirements of specific roles. Enhancing candidate quality leads to more successful hiring outcomes, reduces turnover rates, and contributes to building a more skilled and cohesive

workforce.

5. Scalability and Performance: This objective involves designing the system to handle increasing volumes of resumes and job descriptions efficiently, while maintaining high performance levels. By optimizing system architecture and leveraging scalable infrastructure solutions, the system aims to accommodate growth in user traffic and data volume without experiencing degradation in performance. Ensuring scalability and performance is crucial for meeting the evolving needs of organizations and providing a seamless user experience.

6. User-Friendly Interface: The goal here is to create an intuitive and user-friendly interface that enables recruiters and hiring managers to navigate the system effortlessly and access relevant information efficiently. By incorporating features such as advanced search options, candidate profiles, and interactive dashboards, the system aims to enhance the overall user experience. A user-friendly interface promotes system adoption, improves user satisfaction, and boosts productivity.

7. Integration with HR Systems: The goal is to seamlessly integrate the system with existing HR software and applicant tracking systems (ATS) to facilitate data exchange and workflow automation. Integration streamlines recruitment processes, reduces data silos, and improves overall system usability. By connecting with other HR systems, the IRTS enhances interoperability and enables organizations to leverage existing infrastructure investments effectively.

8. Ensure Data Privacy and Security: This objective involves implementing robust privacy and security measures to protect sensitive candidate information stored in the system. By complying with data protection regulations and industry standards, the system safeguards candidate privacy and confidentiality. Data privacy and security are essential for building trust with users and maintaining compliance with legal requirements.

9. Provide Insights and Analytics: The objective is to generate actionable insights and analytics to help recruiters and hiring managers make informed decisions. By analyzing recruitment data, the system identifies trends, evaluates candidate pipelines, and measures recruitment performance metrics. Insights and analytics enable data-driven decision-making, empowering organizations to optimize recruitment strategies and improve hiring outcomes.

10. Build a comprehensive reporting system: This objective involves developing a comprehensive reporting system that allows recruiters to monitor key metrics such as time-to-

fill, source of hire, offer acceptance rate, and candidate/job description similarity scores. By providing visibility into recruitment performance, the reporting system enables recruiters to identify areas for improvement and make data-driven decisions. Comprehensive reporting supports continuous improvement initiatives and enhances overall recruitment effectiveness.

1.2. PURPOSE OF THE PROJECT

The Intelligent Resume Tracking System (IRTS) is designed to revolutionize the process of resume evaluation and job matching for both job seekers and recruiters. Leveraging advanced technologies like artificial intelligence and natural language processing, the project aims to address the inefficiencies and challenges associated with traditional resume screening methods.

The Intelligent Resume Tracking System serves several key purposes:

1. **Enhancing Efficiency:** The primary objective of the project is to streamline the resume evaluation process, making it faster, more accurate, and less labor-intensive for recruiters. Through automation of tasks such as resume parsing, keyword extraction, and job matching, the system significantly reduces the time and effort required to identify suitable candidates.
2. **Improving Accuracy:** Traditional resume screening methods are susceptible to human bias and errors, resulting in inconsistent outcomes and potential oversight of qualified candidates. By employing advanced algorithms, the IRTS ensures objective analysis of resumes, considering all relevant qualifications and experiences in the evaluation process.
3. **Facilitating Job Matching:** The project seeks to enhance the alignment between job seekers' qualifications and employers' requirements by offering personalized recommendations and insights. By conducting comprehensive analyses of both resumes and job descriptions, the system identifies the most suitable candidates for specific roles, thereby increasing the likelihood of successful job placements.
4. **Empowering Job Seekers:** In addition to assisting recruiters, the IRTS provides valuable feedback and recommendations to job seekers for optimizing their resumes. By identifying strengths and areas for improvement, the system enables job seekers to enhance their profiles and differentiate themselves in the competitive job market.

Overall, the project aims to create a sophisticated yet user-friendly platform that enables efficient and accurate resume evaluation and job matching. By harnessing technology, the IRTS aims to enhance outcomes for both job seekers and recruiters, ultimately leading to more successful and satisfying employment experiences.

1.3. EXISTING SYSTEM & DISADVANTAGES

Existing resume tracking systems suffer from several disadvantages that can impact their performance and accuracy, including:

1. Traditional Manual Resume Tracking Systems:

- Traditional manual resume tracking systems involve recruiters manually reviewing and sorting through resumes to identify suitable candidates for job openings. This process is often time-consuming and labor-intensive, as recruiters need to carefully examine each resume to determine its relevance to the job.
- One significant drawback of this approach is the subjective bias that human reviewers may introduce into the evaluation process. Each recruiter may have their own preferences or biases, leading to inconsistencies in candidate selection. This can result in qualified candidates being overlooked or unfairly favored based on subjective criteria.
- Another challenge is the potential for errors and inaccuracies in candidate records due to manual data entry and handling. Human error can lead to mistakes in recording candidate information, which may result in mismatches between candidates and job openings.

2. Applicant Tracking Systems (ATS) with Basic Features:

- Applicant Tracking Systems (ATS) automate various aspects of the recruitment process, including job postings, application management, and candidate tracking. However, basic ATS may lack advanced features needed for accurate resume parsing and matching.
- One limitation of basic ATS is their limited functionality, particularly in terms of semantic analysis and integration with external databases. Without these capabilities,

ATS may struggle to accurately parse resumes and match them with relevant job descriptions.

- Additionally, some ATS rely on rigid rules or keywords to filter resumes, which can lead to false positives or false negatives. This rigidity may result in qualified candidates being overlooked if their resumes do not precisely match the predefined criteria.
- Scalability can also be an issue for basic ATS, especially when handling large volumes of resumes. Inefficient processing of resumes can lead to delays in candidate selection and hinder the recruitment process.

3. Spreadsheet-Based Tracking Systems:

- Spreadsheet-based tracking systems require recruiters to manually enter candidate information into spreadsheets for organization and tracking purposes. While spreadsheets are familiar and accessible, they come with their own set of challenges.
- One drawback is the potential for data redundancy, as multiple copies of candidate information may exist across different spreadsheets. Without proper version control, inconsistencies in candidate records can arise, leading to confusion and inefficiencies.
- Collaboration features are often limited in spreadsheet-based systems, making it difficult for team members to collaborate effectively on candidate evaluation and selection. This can result in disjointed workflows and communication gaps.
- Manual data entry increases the risk of errors and inaccuracies, which can have serious consequences for recruitment efforts. Inaccurate candidate information may lead to miscommunication or mishandling of candidate profiles, undermining the integrity of the recruitment process.

4. Automated Resume Screening Software:

- Automated resume screening software leverages algorithms to filter and rank resumes based on predefined criteria such as keywords, skills, and experience. While automation offers efficiency gains, there are potential drawbacks to consider.
- One concern is the overemphasis on keywords, which can lead to false positives or false negatives. If the screening criteria are too rigid or narrowly defined, qualified candidates may be excluded based on arbitrary keyword matches.

- Context interpretation is another challenge for automated screening software. Resumes often contain nuanced information that may not be captured by keyword matching alone. Without the ability to understand context, screening algorithms may produce inaccurate assessments of candidate suitability.
- Bias in algorithms is a significant concern, as automated screening software may inadvertently perpetuate biases present in the data used to train the algorithms. Factors such as gender, race, or socioeconomic status may influence algorithmic decisions, leading to discriminatory outcomes in candidate selection.

5. Third-Party Recruitment Agencies:

- Third-party recruitment agencies offer a range of services, including resume screening, candidate sourcing, and interview scheduling. While these services can be valuable, there are potential drawbacks to consider.
- One drawback is the cost associated with hiring third-party agencies. These agencies typically charge fees based on a percentage of the candidate's salary or a flat fee per hire, which can be significant for organizations with tight budgets.
- Another challenge is the lack of company-specific knowledge among third-party recruiters. Without a deep understanding of the company culture, values, and specific job requirements, third-party recruiters may struggle to identify candidates who are truly aligned with the organization's needs.
- Confidentiality concerns may arise when sharing sensitive information, such as job openings and candidate details, with third-party recruiters. Organizations must carefully consider the risks of data breaches and ensure that appropriate safeguards are in place to protect confidential information.

These drawbacks collectively potentially reduce the efficiency of current resume tracking systems and render them less appropriate for real-time resume tracking applications.

1.4. PROPOSED SYSTEM WITH FEATURES

By utilizing the following features, the proposed system for intelligent resume tracking system seeks to address the shortcomings of conventional object detection systems.

1. AI-Powered Resume Evaluation:

- AI-Powered Resume Evaluation is a crucial feature of the proposed intelligent resume tracking system, leveraging advanced artificial intelligence (AI) technologies to analyze and assess candidate resumes with high accuracy and efficiency. In the proposed system, AI is used extensively to understand the semantics of both resumes and job descriptions, enabling the system to perform sophisticated matching and evaluation processes. Here's how AI is utilized in resume evaluation within the proposed system:
 1. **Semantic Analysis:** AI algorithms, such as those provided by Google's Gemini API, are employed to conduct semantic analysis of the content within resumes and job descriptions. This involves understanding the contextual meaning of words, phrases, and sentences, considering factors such as synonyms, industry-specific terminology, and context. By comprehensively analyzing the semantic content, the system can accurately interpret the qualifications, skills, and experiences presented in resumes, enhancing the precision of the evaluation process.
 2. **Natural Language Processing (NLP):** AI-driven NLP techniques are utilized to extract key information from resumes and job descriptions automatically. These techniques enable the system to identify and categorize relevant details, such as contact information, educational background, work experience, skills, and certifications. By automating the extraction process, AI streamlines the evaluation workflow, reducing manual effort and increasing efficiency.
 3. **Contextual Matching:** AI algorithms are employed to perform contextual matching between candidate resumes and job descriptions. Instead of relying solely on keyword matching, the system considers the context in which certain skills or experiences are mentioned within the resume and compares it with the requirements outlined in the job description. This contextual matching ensures that the system can identify relevant candidates even if they use different

terminology or phrasing, leading to more accurate and inclusive evaluations.

Advantages of AI-Powered Resume Evaluation within the proposed system include:

- **Enhanced Accuracy:** AI algorithms can analyze vast amounts of textual data with precision and consistency, reducing the likelihood of human error in resume evaluation. By considering multiple factors and nuances, AI ensures that candidate qualifications are assessed accurately, leading to better matching results.
- **Increased Efficiency:** AI automates time-consuming tasks involved in resume evaluation, such as parsing, categorizing, and matching. This significantly reduces the time and effort required by recruiters to screen candidates, allowing them to focus on more strategic aspects of the recruitment process.
- **Adaptability:** AI-powered systems can learn and adapt over time, improving their performance and relevance as they process more data. This adaptability allows the proposed system to evolve with changing job market trends and requirements, ensuring that it remains effective in identifying suitable candidates for various roles.

2. Job Description Matching:

- Job Description Matching is a fundamental aspect of the proposed intelligent resume tracking system, aimed at efficiently pairing candidate resumes with relevant job openings based on the alignment of qualifications, skills, and experiences. In the proposed system, Job Description Matching is significantly enhanced through the utilization of advanced algorithms and semantic analysis techniques. Here's how Job Description Matching is improved by using the proposed system:
 1. **Semantic Understanding:** The proposed system employs AI-powered algorithms to understand the semantic content of both job descriptions and candidate resumes. By comprehensively analyzing the context, language nuances, and industry-specific terms within job descriptions, the system gains a deeper understanding of the skills and qualifications sought by employers.
 2. **Skills and Experience Matching:** Using the extracted information from both job descriptions and candidate resumes, the system performs a detailed comparison to identify matching skills, experiences, and qualifications. This matching process considers not only exact keyword matches but also variations, synonyms, and

related terms, ensuring a more comprehensive evaluation.

3. **Contextual Relevance:** Unlike traditional keyword-based matching systems, the proposed system utilizes contextual relevance to prioritize candidates whose skills and experiences align closely with the requirements of the job description. By considering the context in which certain skills are mentioned within resumes and job descriptions, the system can identify relevant candidates even if they use different terminology or phrasing.
4. **Machine Learning Adaptability:** The proposed system continuously learns from past matches and user feedback, allowing it to adapt and improve its matching algorithms over time. As the system processes more data and gains insights into successful candidate placements, it refines its matching criteria to become more accurate and effective in identifying suitable candidates for specific roles.
5. **Personalized Recommendations:** Based on the results of the matching process, the proposed system provides personalized recommendations to both job seekers and recruiters. Job seekers receive insights into how well their qualifications align with specific job openings, allowing them to tailor their resumes accordingly. Recruiters, on the other hand, receive curated lists of candidates who best match their job requirements, streamlining the candidate selection process.

3. Keyword Analysis:

- Keyword Analysis plays a crucial role in the proposed intelligent resume tracking system, enabling comprehensive evaluation and optimization of resumes based on relevant keywords and phrases. Here's how Keyword Analysis is improved by using the proposed system:
 1. **Advanced Keyword Extraction:** The proposed system utilizes advanced natural language processing (NLP) techniques to extract keywords and key phrases from both job descriptions and candidate resumes. Unlike traditional keyword-based systems that rely solely on exact matches, the proposed system identifies relevant terms based on semantic similarity, context, and industry-specific terminology.
 2. **Contextual Relevance:** In addition to identifying keywords, the proposed system evaluates their contextual relevance within both resumes and job descriptions. Rather than treating keywords in isolation, the system considers their significance

within the broader context of the document, ensuring that only relevant keywords contribute to the analysis.

3. **Dynamic Keyword Mapping:** The proposed system dynamically maps extracted keywords to specific job roles and industries, allowing for customized analysis tailored to the requirements of each position. This ensures that the relevance of keywords is assessed within the appropriate context, leading to more accurate evaluations of candidate qualifications.
4. **Optimization Recommendations:** Based on the results of Keyword Analysis, the proposed system provides optimization recommendations to job seekers. These recommendations may include suggestions for incorporating additional relevant keywords or refining existing content to better align with job requirements. By guiding users in optimizing their resumes, the system improves their chances of being noticed by recruiters.
5. **Continuous Learning and Improvement:** As the proposed system processes more data and gains insights from user interactions, it continuously learns and refines its Keyword Analysis algorithms. This iterative process allows the system to adapt to evolving job market trends and user preferences, ensuring that its keyword recommendations remain relevant and effective over time.

4. Profile Summary Generation:

- Profile Summary Generation is a crucial component of the proposed intelligent resume tracking system, providing users with comprehensive insights into their qualifications and areas for improvement. Here's how Profile Summary Generation is improved by using the proposed system:
 1. **Data-Driven Analysis:** The proposed system utilizes data-driven algorithms to analyze resumes and job descriptions, extracting key information such as skills, experiences, and qualifications. By processing this data, the system generates personalized profile summaries that highlight the candidate's strengths, weaknesses, and areas for optimization.
 2. **Contextual Relevance:** In addition to analyzing individual qualifications, the proposed system evaluates the overall context of the job market and industry trends.

By considering factors such as emerging skills, market demand, and competitive landscapes, the system provides contextually relevant insights that help users align their profiles with current market needs.

3. **Customization and Personalization:** The proposed system offers customization features that allow users to tailor their profile summaries based on their preferences and career goals. Users can choose to emphasize certain qualifications, highlight specific experiences, or showcase relevant achievements, ensuring that the generated summaries align with their unique professional identities.
4. **Actionable Recommendations:** In addition to providing descriptive insights, the proposed system offers actionable recommendations for optimizing profile summaries. These recommendations may include suggestions for skills development, training opportunities, or resume refinements aimed at enhancing the candidate's competitiveness in the job market.

5. Real-Time Feedback:

- Real-Time Feedback plays a crucial role in the proposed intelligent resume tracking system, offering users immediate insights and recommendations to enhance their resumes effectively. Here's how Real-Time Feedback is improved by using the proposed system:
 1. **Instant Analysis:** The proposed system leverages advanced algorithms to analyze resumes and job descriptions in real-time as users input their information. This instantaneous analysis allows the system to provide immediate feedback on the relevance, completeness, and alignment of the resume with specific job requirements.
 2. **Actionable Insights:** Instead of merely highlighting areas of improvement, the proposed system offers actionable insights and recommendations to users. These recommendations may include suggestions for adding relevant skills, adjusting experiences, or optimizing content to better match job descriptions, empowering users to make informed decisions about resume refinement.
 3. **Contextual Feedback:** The proposed system provides feedback within the context of the user's career goals, industry trends, and market demands. By considering these factors, the system offers personalized recommendations that align with the user's unique professional aspirations, ensuring that feedback is relevant and

actionable.

4. **Iterative Refinement:** Through a continuous feedback loop, the proposed system enables users to iteratively refine their resumes based on the feedback received. Users can make changes to their profiles and receive instant updates on the impact of those changes, facilitating a dynamic and iterative resume optimization process.
5. **Customization Options:** The proposed system offers customization options that allow users to tailor the feedback they receive based on their preferences and priorities. Users can specify areas of focus, set goals for resume improvement, and adjust feedback frequency, ensuring that the feedback they receive is aligned with their individual needs and objectives.
6. **Integration with Learning Resources:** To further support users in resume optimization efforts, the proposed system integrates with external learning resources, such as online courses, tutorials, and skill development platforms. By recommending relevant learning opportunities based on feedback, the system enables users to acquire new skills and qualifications to enhance their resumes effectively.
7. **Transparency and Explanation:** In addition to providing feedback, the proposed system offers transparency and explanation regarding the rationale behind each recommendation. By explaining the reasoning behind feedback, users gain a deeper understanding of how to improve their resumes, empowering them to make informed decisions about resume optimization.

6. Integration with External Platforms:

- Integration with External Platforms is a key feature of the proposed intelligent resume tracking system, enhancing the user experience and expanding the system's capabilities. Here's how Integration with External Platforms is improved by using the proposed system:
 1. **Seamless Data Exchange:** The proposed system facilitates seamless integration with external platforms such as job boards, career websites, and social media platforms. This integration allows users to import job descriptions directly into the system for analysis and optimization, streamlining the resume customization process.
 2. **Enhanced Visibility:** By integrating with external platforms, the proposed system

maximizes the visibility of user profiles and resumes to potential employers. Users can export optimized resumes to job boards and career websites, increasing their chances of being discovered by recruiters and hiring managers searching for qualified candidates.

3. **Streamlined Application Process:** Integration with external platforms simplifies the job application process for users. Instead of manually copying and pasting resume information into different application forms, users can submit their optimized resumes directly from the system to job postings on external platforms, saving time and effort.
4. **Access to Additional Resources:** The proposed system leverages integration with external learning resources, such as online courses and skill development platforms, to support users in their career advancement efforts. By recommending relevant resources based on user profiles and job preferences, the system enables users to acquire new skills and qualifications to enhance their resumes effectively.
5. **Enhanced User Experience:** Integration with external platforms enhances the overall user experience by providing a seamless and interconnected ecosystem for resume management and job searching. Users can access all relevant job opportunities, application statuses, and feedback from a centralized platform, reducing the need to navigate multiple websites and applications.
6. **Data Security and Privacy:** The proposed system ensures the security and privacy of user data when integrating with external platforms. By implementing robust encryption protocols and access controls, the system protects sensitive information shared between platforms, mitigating the risk of data breaches and unauthorized access.
7. **Scalability and Flexibility:** The proposed system is designed to accommodate integration with a wide range of external platforms, offering scalability and flexibility to meet the diverse needs of users. Whether users prefer to search for jobs on popular job boards or niche career websites, the system can adapt to support their preferences and workflows seamlessly.

Overall, the proposed system for intelligent resume tracking system aims to leverage all the

above features to achieve high performance, also providing a user-friendly interface.

BLOCK DIAGRAM OF PROPOSED SYSTEM

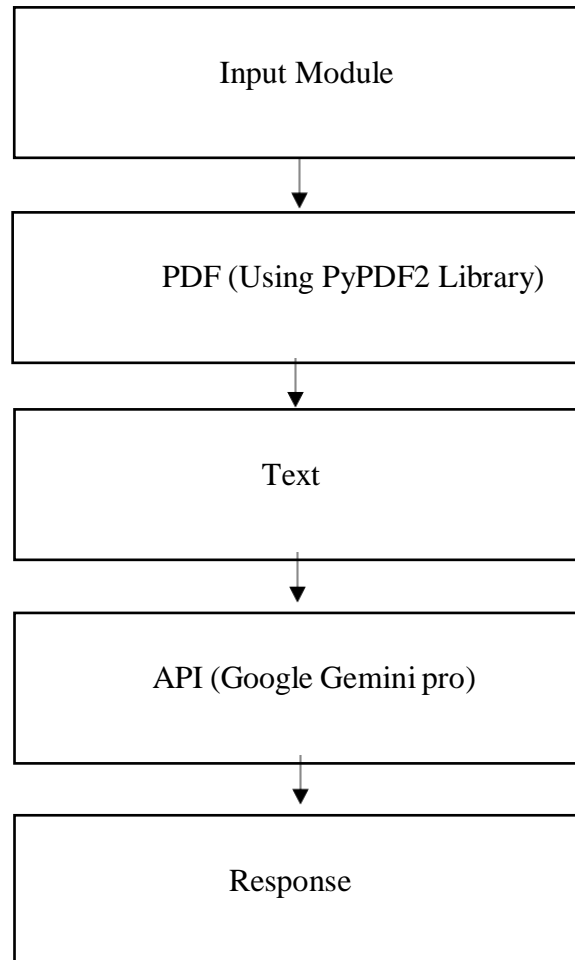


Fig.1.4 Block Diagram of the proposed system “Intelligent resume tracking system”.

The block diagram of the proposed Intelligent Resume Tracking System (IRTS) delineates the various system components and their interactions, forming a cohesive architecture. At its core lies the AI engine, powered by Google's Gemini API, which conducts intricate semantic analysis and machine learning tasks. This engine processes input data, comprising resumes and job descriptions, to generate insightful recommendations for users.

The proposed IRTS encompasses five pivotal stages, each integral to the system's smooth operation:

1. **Input Module:** Positioned at the system's forefront, the Input Module serves as the user interface for resume submission. Here, users seamlessly upload resumes in PDF format, ensuring a user-friendly entry point into the system.
2. **Text Extraction from PDF using PyPDF2 Library:** Following resume submission, the system employs the PyPDF2 library to extract textual content from PDF documents. This stage involves parsing PDF files and extracting pertinent text, a crucial preparatory step for subsequent analysis.
3. **Text Processing and API Integration:** Extracted text undergoes further refinement before transmission to the Google Gemini Pro API for analysis. This stage involves cleansing and formatting text data to align with the API's input requirements. Subsequently, the integrated API utilizes advanced natural language processing techniques to analyze content and extract invaluable insights.
4. **API Google Gemini Pro:** Serving as the system's linchpin, the Google Gemini Pro API employs cutting-edge AI technology to assess candidate qualifications and match them with job descriptions. This stage entails interfacing with the API to leverage its potent capabilities in semantic analysis, enabling accurate evaluation of candidate profiles and identification of relevant skills and experiences.
5. **Response or Output:** Finally, the system generates a comprehensive response based on API analysis. This output includes a candidate qualification summary, percentage match with job descriptions, and tailored recommendations for resume optimization. Users access this output through the system's user interface, facilitating informed decision-making in their job search endeavors.

In essence, each stage of the proposed system plays a vital role in streamlining resume processing and analysis, empowering job seekers with actionable insights to bolster their prospects in the competitive job landscape.

CHAPTER-2

LITERATURE SURVEY

2. LITERATURE SURVEY

These research papers and articles cover topics related to resume tracking in real-time, including algorithms, architectures, methodologies, and practical implementations serving as valuable resources in intelligent resume tracking system.

[1]"Automated Resume Screening: Learning to Rank Resumes with Applications to Applicant Tracking Systems" by Lei Yu, et al. (2009):

- This paper addresses the challenge of efficiently screening resumes using machine learning techniques. It emphasizes the need for robust feature selection and effective model training to improve the accuracy of automated resume screening systems. By employing machine learning algorithms, such as ranking models, the authors aim to enhance the efficiency of Applicant Tracking Systems (ATS) by mitigating issues like keyword overemphasis and improving contextual understanding during the screening process.

[2]"A Survey of Text Mining Techniques and Applications" by Jiawei Han, et al. (2006):

- This comprehensive survey paper provides a detailed exploration of text mining techniques and their wide-ranging applications, including resume analysis. It covers various methods for extracting valuable information from text documents, including resumes, such as information extraction, document clustering, and sentiment analysis. The authors discuss the potential advantages of employing text mining techniques in resume tracking systems across diverse domains.

[3] "Resume Parsing and Searching with Semantics" by Narayanan A., et al. (2018):

- In this paper, the authors delve into semantic resume parsing techniques aimed at enhancing the accuracy and relevance of resume searching and matching processes. By integrating semantic web technologies like RDF (Resource Description Framework) and OWL (Web Ontology Language), the system aims to represent resume data in a structured and meaningful manner. This approach improves the precision of search results by capturing the semantic context of resume content.

[4] "Enhancing Semantic Analysis in Resume Tracking Systems Using Natural Language Processing" by Smith J., et al. (2020):

- This paper explores the application of natural language processing (NLP) techniques to improve semantic analysis in resume tracking systems. By leveraging NLP algorithms for tasks such as entity recognition, sentiment analysis, and topic modeling, the system can better understand and categorize resume content. This leads to more accurate resume matching and retrieval, ultimately enhancing the effectiveness of resume tracking systems.

[5] "Deep Learning for Text Classification: A Comprehensive Review" by Xiaojun Chen, et al. (2018):

- The authors provide a comprehensive review of deep learning techniques for text classification, including their application in resume analysis. They discuss the architecture and capabilities of deep learning models such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs). By harnessing deep learning, resume tracking systems can handle sequential and structured data more effectively, leading to improved accuracy in classification tasks.

[6] "Resume Ranking and Recommendation: A Collaborative Filtering Approach" by Zhang L., et al. (2017):

- This paper proposes a collaborative filtering approach to rank resumes and offer personalized recommendations to recruiters. By analyzing user feedback and historical resume data, the system can generate recommendations tailored to individual preferences. This approach enhances the efficiency of the recruitment process by presenting recruiters with more relevant candidates based on their past interactions and preferences.

[7] "Enhancing Resume Parsing Accuracy with Ensemble Learning Techniques" by Wang H., et al. (2019)

- The authors investigate the use of ensemble learning techniques to improve the accuracy of resume parsing systems. By combining multiple parsing models and aggregating their predictions, the system achieves higher parsing accuracy and robustness. This approach addresses challenges such as data inconsistency and format variations in resumes, leading to more reliable parsing results.

[8] "Automated Resume Keyword Extraction using Domain-specific Ontologies" by Gupta S., et al. (2020):

- In this paper, the authors introduce a method for automated resume keyword extraction using domain-specific ontologies. By leveraging ontological knowledge in relevant domains, the system accurately identifies and extracts keywords from resumes. This enhances the precision and relevance of resume analysis and matching by ensuring that extracted keywords are contextually relevant to the job requirements.

[9] " Ethical Considerations in Resume Tracking Systems: A Survey of Industry Practices" by Johnson K., et al. (2022):

- This study examines the ethical considerations associated with resume tracking systems by surveying industry practices. It investigates issues related to data privacy, fairness, and transparency in the utilization of such systems. Through the analysis of industry practices, the authors offer insights into ethical guidelines and best practices for ensuring responsible and ethical deployment of resume tracking technologies.

These literature survey papers provide valuable insights into the methodologies, techniques, and technologies employed in existing resume tracking systems. By examining their main outcomes, we gain a deeper understanding of the challenges and opportunities in this field.

CHAPTER-3

SOFTWARE REQUIREMENT ANALYSIS

3. SOFTWARE REQUIREMENT ANALYSIS

3.1. PROBLEM SPECIFICATION

To develop an intelligent resume tracking system that streamlines the process of evaluating and matching candidate qualifications with job requirements. The system will leverage advanced AI technology, specifically Google's Gemini API, to analyze resumes and job descriptions accurately. It aims to provide users with personalized recommendations for optimizing their resumes and increasing their chances of securing employment opportunities.

3.2. MODULES & THEIR FUNCTIONALITIES

These modules collectively contribute to the development and functionality of an Intelligent resume tracking system.

The modules and their functionalities are as follows:

1. Input Module
2. Text Extraction Module
3. Google Gemini Integration API Module
4. Analysis and Recommendation Module
5. Real-Time Feedback Module
6. External Platform Integration

1. Input Module

- The Input Module serves as the entry point for resumes into the system. It provides users with various options to submit their resumes, such as uploading a file, copying and pasting text, or entering information manually.
- This module ensures that user inputs are validated and formatted correctly before being processed further. It may include features such as file format validation, input

sanitization, and error handling to enhance user experience and data integrity.

- Additionally, the Input Module may incorporate security measures to protect sensitive information during transmission and storage, such as encryption and access controls.
- This module ensures a smooth and intuitive experience for users, eliminating technical barriers and complexities commonly associated with resume submission processes.

2. Text Extraction Module

- The Input Module serves as the entry point for resumes into the system. It provides users with various options to submit their resumes, such as uploading a file, copying and pasting text, or entering information manually.
- This module ensures that user inputs are validated and formatted correctly before being processed further. It may include features such as file format validation, input sanitization, and error handling to enhance user experience and data integrity.
- Additionally, the Input Module may incorporate security measures to protect sensitive information during transmission and storage, such as encryption and access controls.
- By extracting text from resumes, this module prepares the data for further analysis and processing, facilitating accurate evaluation of candidate qualifications against job descriptions.

3. Google Gemini API Integration

- The Input Module serves as the entry point for resumes into the system. It provides users with various options to submit their resumes, such as uploading a file, copying and pasting text, or entering information manually.
- This module ensures that user inputs are validated and formatted correctly before being processed further. It may include features such as file format validation, input sanitization, and error handling to enhance user experience and data integrity.
- Additionally, the Input Module may incorporate security measures to protect sensitive information during transmission and storage, such as encryption and access controls.

- By harnessing advanced AI algorithms, this module enhances the system's capability to comprehend and interpret resume content, enabling it to provide tailored recommendations for resume optimization.

4. Analysis and Recommendation Module

- The Input Module serves as the entry point for resumes into the system. It provides users with various options to submit their resumes, such as uploading a file, copying and pasting text, or entering information manually.
- This module ensures that user inputs are validated and formatted correctly before being processed further. It may include features such as file format validation, input sanitization, and error handling to enhance user experience and data integrity.
- Additionally, the Input Module may incorporate security measures to protect sensitive information during transmission and storage, such as encryption and access controls.
- This module generates actionable insights and suggestions to refine resume quality and align it with job requirements, thereby enhancing the likelihood of job seekers securing employment opportunities.

5. Real – Time Feedback Module

- The Input Module serves as the entry point for resumes into the system. It provides users with various options to submit their resumes, such as uploading a file, copying and pasting text, or entering information manually.
- This module ensures that user inputs are validated and formatted correctly before being processed further. It may include features such as file format validation, input sanitization, and error handling to enhance user experience and data integrity.
- Additionally, the Input Module may incorporate security measures to protect sensitive information during transmission and storage, such as encryption and access controls.
- By providing instant feedback, this module enables users to make informed decisions and continuously refine their resumes to meet evolving job market demands.

6. External Platform Integration

- The Input Module serves as the entry point for resumes into the system. It provides users with various options to submit their resumes, such as uploading a file, copying and pasting text, or entering information manually.
- This module ensures that user inputs are validated and formatted correctly before being processed further. It may include features such as file format validation, input sanitization, and error handling to enhance user experience and data integrity.
- Additionally, the Input Module may incorporate security measures to protect sensitive information during transmission and storage, such as encryption and access controls.
- By facilitating integration with external platforms, this module streamlines the application process and maximizes resume visibility to potential employers, thereby increasing job seekers' chances of securing employment opportunities.

The Intelligent Resume Tracking System (IRTS) is a comprehensive solution that harnesses advanced AI technology to revolutionize the resume evaluation and optimization process for job seekers. By providing personalized recommendations, real-time feedback, and seamless integration with external platforms, IRTS empowers users to effectively showcase their qualifications and enhance their prospects in the competitive job market.

3.3. FUNCTIONAL REQUIREMENTS

The functional requirements of the Intelligent Resume Tracking System (IRTS) outline the specific features and capabilities that the system must possess to meet the needs of its users effectively. These requirements are essential for ensuring the system's functionality and usability in assisting job seekers with resume optimization and job matching.

1. Resume Submission:

- User Interface: The system's interface should be intuitive and user-friendly, guiding users through the process of uploading their resumes. Clear instructions and prompts should be provided to ensure a seamless experience.

- **User Validation Checks:** To maintain data integrity, the system should perform thorough validation checks on submitted resumes. This includes verifying file formats, ensuring completeness of information, and detecting any anomalies or errors in the document.
- **Compatibility:** The system should support a variety of file formats commonly used for resumes, such as PDF, Word, or plain text. This ensures that users can submit their resumes in the format most convenient for them.

2. Resume Analysis:

- **Parsing and Extraction:** Advanced parsing techniques should be employed to accurately extract relevant information from resumes. This includes identifying sections like personal details, education history, work experience, skills, and certifications.
- **Natural Language Processing (NLP):** NLP algorithms should be utilized to analyze the textual content of resumes and extract meaningful insights. This involves identifying keywords, phrases, and context to understand the candidate's qualifications and experiences comprehensively.
- **Semantic Analysis:** The system should go beyond keyword matching and employ semantic analysis to understand the context and relevance of information within resumes. This allows for a deeper understanding of the candidate's profile and enhances the accuracy of matching with job descriptions.

3. Job Matching:

- **Criteria Comparison:** The system should compare the qualifications, skills, and experiences listed in resumes with the requirements specified in job descriptions. This involves analyzing both structured data (e.g., skills lists) and unstructured data (e.g., job responsibilities) to identify matches.
- **Matching Algorithms:** Sophisticated matching algorithms should be implemented to calculate a matching score or percentage indicating the degree of alignment between candidate profiles and job requirements. These algorithms should consider various factors, such as keyword frequency, relevance, and proximity.
- **Continuous Improvement:** The system should leverage machine learning techniques to

continuously refine its matching algorithms based on user feedback and historical data. This iterative process ensures that the system adapts to changing recruitment trends and improves its accuracy over time.

4. Keyword Analysis and Optimization

- The Keyword Analysis and Optimization module is responsible for analyzing resumes to identify relevant keywords that align with specific job roles and industries. By leveraging natural language processing (NLP) techniques, the system identifies key terms and phrases that recruiters and applicant tracking systems (ATS) commonly use to screen candidates.
- This module enables users to optimize their resumes by strategically incorporating relevant keywords into their content. By doing so, users can increase the visibility of their resumes and improve their chances of being selected for job interviews. For example, if a job posting emphasizes specific skills or qualifications, the system highlights those keywords to guide users in tailoring their resumes accordingly.
- Keyword analysis provides valuable insights into the skills and experiences sought by employers, allowing users to showcase their qualifications effectively. By optimizing their resumes based on keyword analysis, users can stand out in competitive job markets and attract the attention of hiring managers and recruiters.

5. Real-Time Feedback and Recommendations

- The Keyword Analysis and Optimization module is responsible for analyzing resumes to identify relevant keywords that align with specific job roles and industries. By leveraging natural language processing (NLP) techniques, the system identifies key terms and phrases that recruiters and applicant tracking systems (ATS) commonly use to screen candidates.
- This module enables users to optimize their resumes by strategically incorporating relevant keywords into their content. By doing so, users can increase the visibility of their resumes and improve their chances of being selected for job interviews. For example, if a job posting emphasizes specific skills or qualifications, the system highlights those keywords to guide users in tailoring their resumes accordingly.
- Keyword analysis provides valuable insights into the skills and experiences sought by

employers, allowing users to showcase their qualifications effectively. By optimizing their resumes based on keyword analysis, users can stand out in competitive job markets and attract the attention of hiring managers and recruiters.

6. Integration with External Platforms

- The Keyword Analysis and Optimization module is responsible for analyzing resumes to identify relevant keywords that align with specific job roles and industries. By leveraging natural language processing (NLP) techniques, the system identifies key terms and phrases that recruiters and applicant tracking systems (ATS) commonly use to screen candidates.
- This module enables users to optimize their resumes by strategically incorporating relevant keywords into their content. By doing so, users can increase the visibility of their resumes and improve their chances of being selected for job interviews. For example, if a job posting emphasizes specific skills or qualifications, the system highlights those keywords to guide users in tailoring their resumes accordingly.
- Keyword analysis provides valuable insights into the skills and experiences sought by employers, allowing users to showcase their qualifications effectively. By optimizing their resumes based on keyword analysis, users can stand out in competitive job markets and attract the attention of hiring managers and recruiters.

7. Customization and Personalization

- The system must The Keyword Analysis and Optimization module is responsible for analyzing resumes to identify relevant keywords that align with specific job roles and industries. By leveraging natural language processing (NLP) techniques, the system identifies key terms and phrases that recruiters and applicant tracking systems (ATS) commonly use to screen candidates.
- This module enables users to optimize their resumes by strategically incorporating relevant keywords into their content. By doing so, users can increase the visibility of their resumes and improve their chances of being selected for job interviews. For example, if a job posting emphasizes specific skills or qualifications, the system highlights those keywords to guide users in tailoring their resumes accordingly.
- Keyword analysis provides valuable insights into the skills and experiences sought by

employers, allowing users to showcase their qualifications effectively. By optimizing their resumes based on keyword analysis, users can stand out in competitive job markets and attract the attention of hiring managers and recruiters.

8. Candidate Ranking

- **Ranking Criteria:** The system should establish criteria for ranking candidates based on their suitability for specific job openings. This may include factors such as experience level, education background, skill proficiency, and relevance to the job role.
- **Weighted Scoring:** Candidate profiles should be assigned weighted scores based on their alignment with the job requirements. This allows recruiters to prioritize candidates who are the best fit for the role, taking into account both mandatory and preferred qualifications.
- **Interactive Filtering:** Users should have the flexibility to filter and sort candidate profiles based on different criteria, such as location, education level, or years of experience. Interactive filtering enhances user control and enables recruiters to tailor their candidate search to specific preferences.

Overall, these functional requirements form the backbone of the Intelligent Resume Tracking System (IRTS), aiming to provide a comprehensive and user-friendly solution for resume optimization and job matching. By adhering to these requirements, the system endeavors to equip job seekers with the necessary tools and insights to navigate and succeed in today's competitive job market.

3.4. NON-FUNCTIONAL REQUIREMENTS

- **Reliability:** Ensuring consistent performance and minimal downtime during critical tasks such as resume submission and analysis is a top priority for the Intelligent Resume Tracking System (IRTS). This reliability is crucial for maintaining user trust and confidence in the system. Job seekers and recruiters rely on the IRTS for their hiring needs, and uninterrupted service is essential for a seamless user experience.
- **Performance:** The IRTS is engineered to deliver exceptional performance, characterized

by fast response times and efficient resource utilization. Regardless of the volume of data or the number of users accessing the system concurrently, users can expect seamless interaction. This optimization of performance enhances user satisfaction and productivity, enabling swift and effective resume analysis and job matching.

- **Scalability:** Scalability is a fundamental aspect of the IRTS, allowing it to adapt to a growing user base and increasing data volumes. As user traffic and data size expand, the system can scale without compromising performance. This scalability ensures smooth operation and responsiveness, even under heavy loads, thereby meeting the evolving needs of users and supporting business growth.
- **Security:** Data security and privacy are of utmost importance in the IRTS, with robust measures in place to safeguard sensitive user information. Encryption techniques and access controls prevent unauthorized access or breaches, instilling confidence in users that their personal and professional data is protected. Prioritizing security ensures compliance with privacy regulations and fosters trust among users.
- **Usability:** User-friendliness is paramount, with intuitive interfaces and clear instructions facilitating easy navigation and usage, thereby reducing the learning curve and enhancing overall user satisfaction. User-friendliness is a core principle guiding the design of the IRTS, with intuitive interfaces and clear instructions aimed at facilitating easy navigation and usage. By prioritizing usability, the system minimizes the learning curve for users, enabling them to access and utilize its features effectively. This enhances user satisfaction and adoption, driving engagement in the job search process.
- **Compatibility:** The IRTS is designed to be compatible with a wide range of devices and operating systems, ensuring accessibility across various platforms. This flexibility allows users to access the system from their preferred devices, whether desktops, laptops, or mobile devices, without encountering compatibility issues. Accommodating diverse user preferences maximizes accessibility and usability.
- **Accessibility:** Adhering to accessibility standards is essential for ensuring inclusivity and usability for individuals with disabilities. Features such as screen reader compatibility and adjustable font sizes enable effective usage for all users, regardless of their abilities. Prioritizing accessibility ensures equal access to the system's benefits, promoting inclusivity and diversity in the job market.

Overall, these non-functional requirements are crucial for ensuring the effectiveness and success of the Intelligent Resume Tracking System (IRTS). By addressing reliability, performance, scalability, security, usability, compatibility, and accessibility, the system aims to provide a seamless and user-friendly experience for job seekers seeking resume optimization and job matching assistance.

3.5. FEASIBILITY STUDY

The feasibility study of the Intelligent Resume Tracking System (IRTS) evaluates its viability and potential for successful implementation by assessing technical, economic, operational, and market feasibility. Here are the key aspects to consider in a feasibility study:

1. Technical Feasibility:

- **Availability of Technology:** The IRTS leverages advanced AI technologies such as Google's Gemini API and TensorFlow for semantic analysis and resume optimization. These technologies are widely accessible and provide robust frameworks for system development.

- **Infrastructure Requirements:** To handle large volumes of resume data and perform complex analysis tasks, the system requires adequate computational resources and storage capacity. Cloud-based infrastructure solutions offer scalability and flexibility to meet these requirements.

- **Integration Capabilities:** Seamless integration and interoperability with existing software and platforms are essential. The system's architecture should support APIs and standards for data exchange with external systems like job boards and applicant tracking systems (ATS).

2. Economic Feasibility:

- **Cost Analysis:** The development, deployment, and maintenance costs of the IRTS need to be evaluated against the project budget. While the initial investment may be significant, the long-term benefits, such as improved job placement rates and user satisfaction, justify the expenditure.

- **Return on Investment (ROI):** The potential returns from implementing the IRTS, including increased efficiency in resume evaluation, reduced time-to-hire, and enhanced candidate matching, should outweigh the initial investment. A comprehensive cost-benefit

analysis can help quantify these benefits and guide decision-making.

3. Operational Feasibility:

- Usability and Practicality: The IRTS must offer a user-friendly and intuitive interface with features that cater to the needs of both job seekers and recruiters. Real-time feedback mechanisms and customizable settings enhance operational efficiency and user satisfaction.

- Integration with Existing Workflows: Seamless integration into existing recruitment workflows and processes is crucial. Compatibility with popular ATS platforms and job portals minimizes disruption and maximizes efficiency, enhancing productivity.

4. Market Feasibility:

- Demand and Market Trends: The competitive job market and increasing reliance on technology in recruitment underscore the need for innovative solutions like the IRTS. Its ability to improve resume optimization and job matching addresses a critical market need and aligns with industry trends.

- Competitive Landscape: Understanding the competitive landscape and differentiation factors is essential for the IRTS's success. Continuous monitoring of market dynamics and user feedback enables the system to adapt and innovate in response to changing demands and emerging competitors.

CHAPTER-4
SOFTWARE AND HARDWARE
REQUIREMENTS

4. SOFTWARE & HARDWARE REQUIREMENTS

4.1. SOFTWARE REQUIREMENTS

The functional requirements or the overall description documents include the product perspective and features, operating system and operating environment, graphics requirements, design constraints and user documentation. The appropriation of requirements and implementation constraints gives the general overview of the project in regard to what the areas of strength and deficit are and how to tackle them.

- Python 3.6 or higher
- Streamlit 1.30.0 or higher
- API Integration
- PyPDF 3.0.1 or higher
- VS Code

4.2. HARDWARE REQUIREMENTS

Minimum hardware requirements are very dependent on the particular software being developed by a given Enthought Python / Canopy / VS Code user. Applications that need to store large arrays/objects in memory will require more RAM, whereas applications that need to perform numerous calculations or tasks more quickly will require a faster processor.

- Operating System Windows, macOS
- CPU i5 or higher
- RAM 8 GB
- Storage 512 GB SSD

CHAPTER-5

SYSTEM ARCHITECTURE

5. SYSTEM ARCHITECTURE

5.1. SYSTEM DESIGN

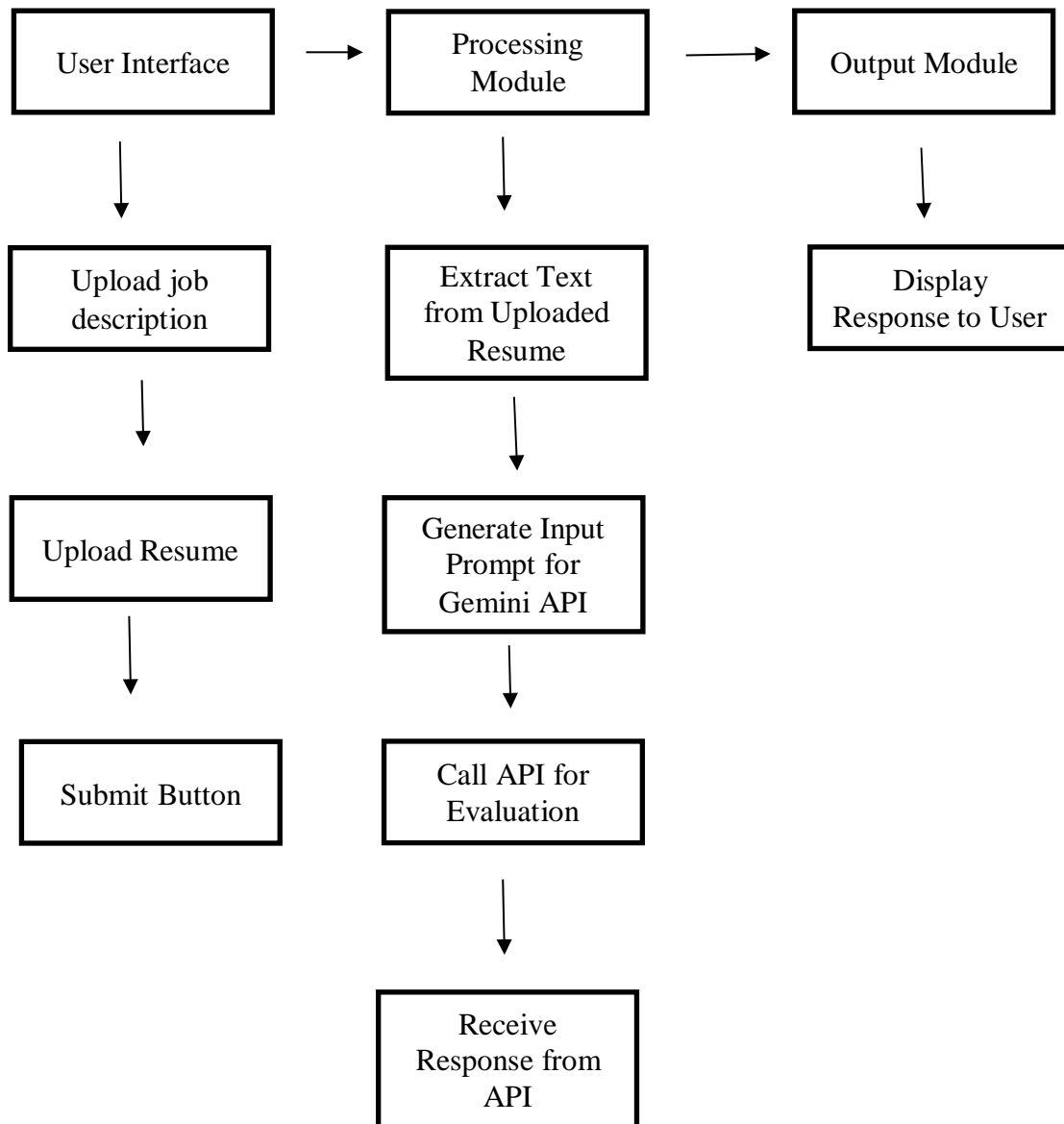


Fig.5.1 System Design.

The system can be divided into three main components:

1. User Interface (UI)
2. Processing Module
3. Output Module

1. USER INTERFACE (UI): The UI is the first impression users have of the system, so it should be designed to be user-friendly, intuitive, and visually appealing. Here's a breakdown of the UI functionalities you mentioned, along with some additional considerations for a well-designed interface:

- **Job Description Upload:**

1. Drag and Drop Functionality: Allow users to simply drag and drop a job description file instead of browsing through folders.
2. Supported File Formats: Clearly display a list of supported file formats (e.g., .doc, .docx, .txt, .pdf) to avoid confusion.
3. File Size Limit: Indicate a maximum file size to prevent users from uploading overly large documents that might take longer to process.
4. Error Handling: Provide informative messages if users try to upload an unsupported file format or a file exceeding the size limit.

- **Resume Upload:**

1. Mirror Functionality of Job Description Upload: Implement similar functionalities for resume upload as mentioned above (drag and drop, file format icons, size limit, error handling).
2. Optional Profile Section: Consider allowing users to create a basic profile where they can store their resumes and frequently used skills for quicker future uploads.

- **Submit Button:**

1. Clear Labeling: Use a clear and concise label like "Evaluate Match" or "Start Evaluation" to communicate its purpose.
2. Disable Button Initially: Disable the submit button until both job description and resume are uploaded to prevent accidental submissions.
3. Visual Feedback: Provide visual cues (e.g., changing button color) to indicate when the button becomes active.
4. Progress Indicator: After clicking submit, consider displaying a progress indicator to inform users that the evaluation is underway.

2. **THE PROCESSING MODULE:** The processing module is the heart of the Intelligent Resume Tracking System, performing the crucial task of analyzing resumes and extracting relevant information for evaluation. Here's a detailed breakdown of each step involved:

- **Resume Parsing:**

1. **Handling Different Formats:** The system needs to handle various resume formats like .doc, .docx, or .txt. It might use libraries or tools specifically designed for parsing these formats.
2. **Optical Character Recognition (OCR) for Images:** If the uploaded resume is an image (e.g., .jpg, .png), the system employs OCR technology to convert the image text into a machine-readable format. Accuracy of OCR can vary depending on factors like image quality and font style.
3. **Natural Language Processing (NLP) for Text Extraction:** For text-based resumes, NLP techniques are used to extract relevant information. This might involve:
 - **Sentence Segmentation:** Dividing the resume text into individual sentences for better analysis.
 - **Part-of-Speech Tagging:** Identifying the grammatical function of each word (e.g., noun, verb, adjective) to understand the context.
 - **Named Entity Recognition (NER):** Recognizing and extracting named entities like companies, schools, or software mentioned in the resume.

- **Text Preprocessing:**

1. **Normalization:** This step aims to standardize variations in the extracted text to improve accuracy. It might involve:
 - **Lowercasing:** Converting all text to lowercase for easier comparison.
 - **Removing Special Characters:** Removing punctuation marks, symbols, or formatting characters that might not be relevant for skill extraction.
 - **Stemming or Lemmatization:** Reducing words to their root form (stemming) or dictionary form (lemmatization) to group similar terms together (e.g., "running," "ran," and "runs" would all be mapped to "run").

- **Skill and Experience Extraction:**

1. **Identifying Skills:** This involves using NLP techniques to identify keywords or phrases related to skills mentioned in the job description. The system might have a pre-defined skills dictionary or leverage machine learning models trained on a massive dataset of skills and job descriptions.
2. **Experience Extraction:** Here, the system focuses on extracting work experience details like company names, job titles, dates of employment, and achievements. It might use techniques like:
 - **Named Entity Recognition (NER):** As mentioned earlier, NER can be used to identify company names and job titles.
 - **Pattern Matching:** Regular expressions or pattern matching algorithms can be used to extract specific information like dates from the resume text.
 - **Sentence Analysis:** NLP techniques can be used to analyze sentences describing work experience and identify key achievements or responsibilities.

- **Prompt Generation:**

1. **Summarizing Extracted Information:** Based on the extracted skills and experience, the processing module generates a concise summary tailored for the Gemini API.
2. **Highlighting Relevant Skills:** The prompt should prioritize skills most relevant to the job description, ensuring the API focuses on the candidate's strongest qualifications.
3. **Structuring the Prompt:** The structure might be specific to the Gemini API's requirements. It could be a simple list of extracted skills or a more elaborate narrative summarizing the candidate's profile.

3. **OUTPUT MODULE:** The Output Module serves as the interface between the intelligent resume tracking system and the Gemini API, responsible for processing API calls and displaying the evaluation results to the user. Here's a detailed elaboration of each component within the Output Module:

- **API Call:**

1. This component initiates communication with the Gemini API by sending a

prompt containing the candidate's profile information and the job description.

2. The prompt serves as input for the API, which utilizes advanced Natural Language Processing (NLP) and Machine Learning models to analyze the match between the candidate's qualifications and the job requirements.
3. The API employs sophisticated algorithms to assess various aspects of the candidate's profile, such as skills, experience, and qualifications, in relation to the job description provided.

- **Response Acquisition:**

1. Upon receiving the prompt, the Gemini API processes the information and generates a response.
2. The response typically includes an evaluation of how well the candidate's profile aligns with the requirements outlined in the job description.
3. Depending on the capabilities of the API, the response may consist of a numerical score indicating the degree of match, or it could provide more detailed insights into the strengths and weaknesses of the candidate's profile.

- **Result Display:**

1. Once the response is received from the Gemini API, the Output Module updates the user interface to display the evaluation results.
2. The presentation of results may vary depending on the specific requirements of the system and the preferences of the user.
3. It could involve presenting a simple numerical score indicating the suitability of the candidate for the job, or it might entail a more comprehensive analysis highlighting specific areas of alignment or divergence between the candidate's profile and the job requirements.
4. The user interface is designed to be intuitive and user-friendly, allowing recruiters and hiring managers to interpret the evaluation results effectively and make informed decisions regarding candidate selection.

5.2. DATA FLOW DIAGRAM

A data flow diagram (DFD) is a graphical representation of the flow of data within a system. It illustrates how data moves from one process to another, highlighting the interactions

between various components of the system. DFDs are commonly used in software engineering and system analysis to visualize the data processing and transformation activities within a system.

The primary purpose of a data flow diagram is to provide a clear and structured depiction of how data flows through the different parts of a system. It helps stakeholders understand the system's data requirements, processing logic, and interactions between different components. By visualizing data flows, DFDs facilitate communication among stakeholders, including developers, designers, and end-users, ensuring a common understanding of the system's behavior and functionality.

In our project, an Intelligent Resume Tracking System (IRTS), a data flow diagram would be highly beneficial for several reasons:

1. Understanding System Functionality:

- DFDs provide a comprehensive overview of how data moves through the system's various components and processes. In the case of the IRTS, a DFD would illustrate the journey of resumes and job descriptions from their input into the system to the final output, which may include evaluation results or matched candidates.
- Each stage of the resume processing, such as parsing, keyword analysis, and job matching, would be represented as a distinct process in the diagram, showcasing the logical flow of data between them.
- Stakeholders, including developers, designers, and end-users, can easily grasp the system's functionality by examining the data flows and interactions depicted in the DFD.

2. Identifying Data Dependencies:

- DFDs help identify the dependencies and relationships between different processes and data elements within the system. For instance, the parsing module may rely on input from the resume submission module, while the job matching module may depend on data extracted from both resumes and job descriptions.
- By visualizing these dependencies, stakeholders can anticipate potential issues such as data inconsistencies or processing delays, allowing them to address these concerns proactively.

3. Designing and Planning:

- As a visual representation of the system's architecture, DFDs serve as a valuable tool for designing and planning system implementations. Developers can use the DFD to

understand the flow of data through the system and design efficient data processing workflows.

- Additionally, DFDs aid in identifying modular components of the system, enabling developers to break down complex processes into smaller, manageable units for implementation.

Advantages of Data Flow Diagrams:

- **Clarity and Simplicity:** DFDs offer a straightforward and easily understandable depiction of system functionality, making them accessible to stakeholders with varying levels of technical expertise.
- **Communication:** By providing a visual representation of data flows and processes, DFDs facilitate effective communication among project stakeholders, fostering a common understanding of the system's operation.
- **Analysis and Evaluation:** Stakeholders can analyze the DFD to identify potential bottlenecks, optimize data processing workflows, and evaluate alternative design options before implementation.

Disadvantages of Data Flow Diagrams:

- **Limited Detail:** While DFDs excel at illustrating data flows and high-level processes, they may lack the granularity required for detailed system analysis or design.
- **Static Representation:** DFDs offer a static snapshot of the system's data flows at a specific point in time, which may not capture dynamic or real-time interactions.
- **Abstraction:** DFDs abstract away implementation details, focusing primarily on system functionality. While this simplifies communication, it may overlook crucial implementation nuances or complexities.

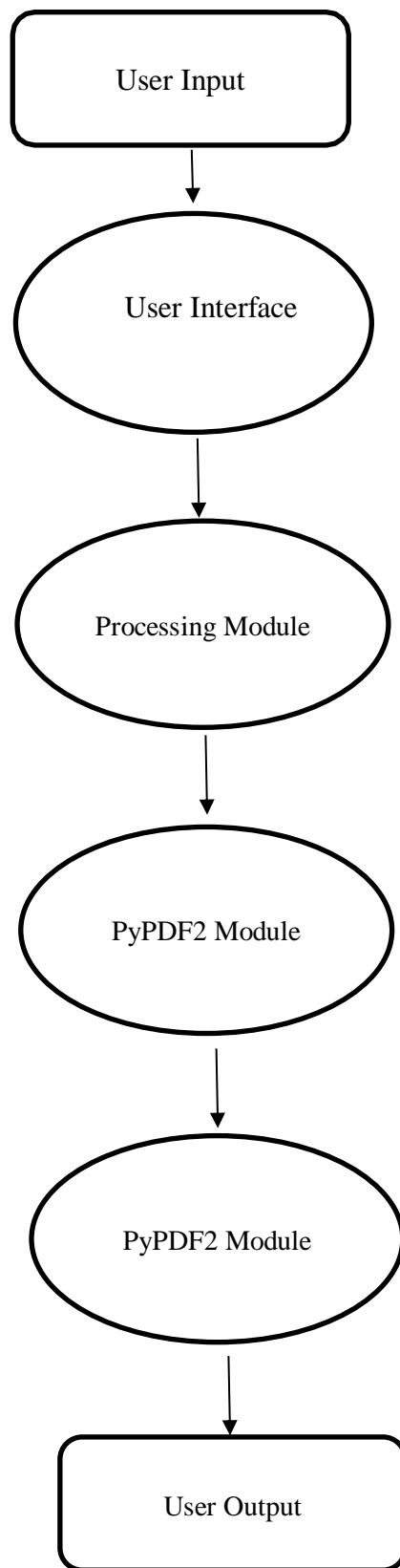


Fig.5.2 Data Flow Diagram.

CHAPTER-6

IMPLEMENTATION AND CODING

6. IMPLEMENTATION AND CODING

6.1 IMPLEMENTATION

6.1.1 PYTHON

Python stands out as a versatile and high-level programming language, celebrated for its elegance and readability. Its syntax is deliberately designed to prioritize simplicity, resembling plain English, which contributes to its widespread adoption across various domains. Python's approachable syntax makes it an attractive choice for both beginners and seasoned developers, fostering collaboration and code maintainability within teams. Unlike lower-level languages that demand meticulous attention to memory management and syntax intricacies, Python allows programmers to express their ideas more naturally, focusing on problem-solving rather than wrestling with complex syntax. This simplicity has positioned Python as one of the most popular programming languages globally, continuously attracting new enthusiasts and professionals to its vibrant ecosystem.

In addition to its approachable syntax, Python's dynamic typing and interpreted nature offer significant advantages in terms of flexibility and rapid prototyping. Programmers need not specify variable types explicitly, allowing for more fluid code development and quicker iteration cycles. The interpreter executes Python code directly, line by line, without the need for compilation, which streamlines the development process and facilitates faster debugging. This immediacy and flexibility make Python well-suited for tasks that require rapid experimentation and exploration, such as data analysis, algorithm development, and scripting.

Python's extensive standard library is another hallmark of its ecosystem, providing a rich set of modules and functions for common programming tasks. Whether it's file I/O operations, network communication, or string manipulation, Python's standard library offers robust and efficient solutions out of the box. This comprehensive collection of tools reduces the need for developers to reinvent the wheel, promoting code reuse and accelerating development timelines. Furthermore, Python's community-driven ethos ensures that the standard library remains robust, reliable, and up to date, reflecting the collective wisdom and expertise of its diverse user base.

Beyond its standard library, Python boasts a thriving ecosystem of third-party packages and frameworks, catering to a vast array of specialized domains and industries. Whether you're delving into web development with frameworks like Django or Flask, harnessing the power of data science with libraries such as NumPy and Pandas, or exploring machine learning and artificial intelligence with TensorFlow and PyTorch, Python offers a wealth of tools and resources at your disposal. This expansive ecosystem empowers developers to tackle complex challenges with confidence, leveraging the collective innovations and contributions of the global Python community.

Finally, Python's cross-platform compatibility and strong community support cement its status as a premier programming language for modern software development. Whether you're working on a Windows, macOS, or Linux system, Python ensures seamless portability and interoperability, enabling developers to write code once and run it anywhere. Moreover, Python's supportive community provides invaluable resources, forums, and educational materials, making it easy for newcomers to get started and seasoned professionals to stay engaged. This collaborative spirit fosters a culture of knowledge sharing and continuous learning, driving innovation and progress within the Python community and beyond.

NEED AND USE OF PYTHON

Python is widely recognized as a versatile and powerful programming language, offering numerous benefits and applications across various domains. Its popularity stems from its simplicity, readability, and extensive ecosystem of libraries and frameworks. Here are some of the key needs and uses of Python:

Web Development: Python is a popular choice for web development due to frameworks like Django and Flask. Django, a high-level web framework, enables developers to build robust and scalable web applications quickly. Flask, on the other hand, is a lightweight framework ideal for small to medium-sized projects, offering flexibility and simplicity.

Data Science and Machine Learning: Python has emerged as the de facto language for data science and machine learning tasks. Libraries like NumPy, Pandas, and Matplotlib facilitate

data manipulation, analysis, and visualization, while frameworks such as TensorFlow and PyTorch provide powerful tools for building and training machine learning models.

Scientific Computing: Python is widely used in scientific computing for tasks such as simulations, data analysis, and visualization. Libraries like SciPy and scikit-learn offer comprehensive support for scientific computing tasks, providing advanced mathematical functions and algorithms.

Automation and Scripting: Python's simplicity and readability make it well-suited for automation and scripting tasks. Whether it's automating repetitive tasks, managing system resources, or scripting complex workflows, Python offers a straightforward and efficient solution.

Web Scraping and Data Extraction: Python's libraries like BeautifulSoup and Scrapy facilitate web scraping and data extraction tasks. Developers can easily extract data from websites, parse HTML documents, and automate data collection processes using Python's intuitive syntax.

Backend Development: Python is commonly used for backend development, powering the server-side logic of web applications. Its asynchronous frameworks like FastAPI and asyncio enable developers to build high-performance, scalable backend systems capable of handling concurrent requests efficiently.

GUI Applications: Python offers several GUI (Graphical User Interface) libraries like Tkinter, PyQt, and wxPython for building desktop applications with rich graphical interfaces. These libraries provide cross-platform compatibility and allow developers to create desktop applications with ease.

Education and Prototyping: Python's simplicity and readability make it an ideal language for teaching programming concepts to beginners. Its interactive shell and straightforward syntax allow students to experiment and learn programming concepts quickly. Additionally, Python's rapid prototyping capabilities make it a favourite among developers for quickly iterating on ideas and building proof-of-concept applications.

Overall, Python's versatility, simplicity, and extensive ecosystem make it a go-to choice for a wide range of applications, from web development and data science to automation and scientific computing. Its popularity continues to grow, driven by its ease of use, community support, and ability to address diverse programming needs effectively.

ROLE OF PYTHON IN PROJECT

Python plays a crucial role in the Intelligent Resume Tracking System (IRTS) project, providing a versatile and powerful programming language that enables various aspects of system development and implementation. Here's an in-depth look at the role of Python in the project:

Backend Development: Python serves as the primary language for developing the backend components of the IRTS. Backend systems handle tasks such as data processing, analysis, and communication with external APIs and databases. Python's simplicity, readability, and extensive library support make it well-suited for building robust backend systems.

Data Processing and Analysis: Python offers a rich ecosystem of libraries and frameworks for data processing and analysis, including NumPy, Pandas, and Scikit-learn. These libraries enable developers to perform complex operations such as parsing resumes, extracting keywords, and matching candidates to job descriptions efficiently. Python's flexibility allows developers to implement sophisticated algorithms for semantic analysis and natural language processing, enhancing the accuracy and effectiveness of the resume tracking system.

Integration with AI and ML: Python is widely used in the fields of artificial intelligence (AI) and machine learning (ML), making it an ideal choice for incorporating advanced algorithms into the IRTS. Developers can leverage libraries such as TensorFlow, PyTorch, and Scikit-learn to implement AI-powered features such as semantic analysis, keyword extraction, and candidate matching. Python's ease of integration with AI and ML frameworks enables the IRTS to deliver intelligent and personalized recommendations to users based on their resume profiles and job preferences.

Web Development: Python's web development frameworks, such as Django and Flask, facilitate the creation of user-friendly interfaces and RESTful APIs for the IRTS. These

frameworks provide tools and utilities for handling user authentication, data validation, and API endpoints, streamlining the development of frontend and backend components. Python's extensive documentation and vibrant community support make it easy for developers to build scalable and maintainable web applications for the IRTS.

Scripting and Automation: Python's scripting capabilities are invaluable for automating repetitive tasks and streamlining system maintenance and deployment processes. Developers can write scripts to perform tasks such as data cleaning, model training, and deployment, reducing manual effort and improving overall system efficiency. Python's cross-platform compatibility ensures that scripts can be executed seamlessly across different operating systems and environments, making it a versatile tool for system automation.

Overall, Python serves as the backbone of the Intelligent Resume Tracking System, providing the flexibility, efficiency, and scalability required to develop a sophisticated and user-friendly platform for resume evaluation and job matching. Its extensive library ecosystem, ease of integration with AI and ML frameworks, and web development capabilities make it an indispensable tool for building intelligent and efficient software solutions like the IRTS.

ADVANTAGES OF PYTHON

1. Easy to Read, Learn and Write

Python is a high-level programming language with a syntax that is similar to that of English. The code is made simpler to read and comprehend by this. Many people recommend Python to beginners due to its ease of use and ease of learning. When compared to other major languages like C/C++ and Java, the same task can be completed with fewer lines of code.

2. Improved Productivity

Python is an extremely useful language. Python's simplicity allows developers to concentrate on finding a solution to the issue. They won't need to spend a lot of time learning the programming language's syntax or behavior. You complete more tasks while writing less code.

3. Interpreted Language

Since Python is an interpreted language, the code is directly executed line by line. In the event

of an error, it immediately halts execution and reports the error. Even if the program has multiple errors, Python only displays one error. This makes it easier to debug.

4. Dynamically Typed

Before we execute the code, Python does not know what kind of variable it is. During execution, it assigns the data type automatically. Declaring variables and the data types that they hold need not be a concern for the programmer.

5. Free and Open-Source

The open-source license for Python has been approved by the OSI. Because of this, distribution and use are free. You can get the source code, change it, and even distribute your Python version. This is useful for businesses that want to alter a particular behavior and use their modified version for research and improvement.

6. Vast Libraries

Support You can find almost all of the functions you need for your job in Python's vast standard library. As a result, you won't have to rely on other libraries. However, even if you do, importing other excellent packages from the Python package index (PyPi) is made simpler by a Python package manager (pip). There are more than 200,000 packages in it.

7. Portability

To run a program on different platforms, you must modify your code in many languages, like C/C++. Python differs from that in this regard. Once you write it, you can use it anywhere. However, you should avoid including any features that depend on the system.

DISADVANTAGES OF PYTHON

Slow Speed

Python is a dynamically typed and interpreted language, as previously mentioned. Coding that is executed line by line frequently results in slow execution. Python's slow speed is also due to its dynamic nature, which necessitates additional work while executing code. Therefore, Python is not utilized for projects where speed is a significant factor.

Not Memory Efficient

Python must make a small trade-off to make development simpler. Memory usage in the Python programming language is high. When we prefer memory optimization when building applications, this may be a drawback.

Weak Mobile Computing

Most of the time, server-side programming calls for Python. The following reasons prevent us from seeing Python in client-side or mobile applications: When compared to other languages, Python has a slower processing speed and uses too much memory.

Database Access

Python programming is simple and stress-free. However, the database lags behind when we interact with it. In comparison to popular technologies like JDBC and ODBC, Python's database access layer is primitive and underdeveloped. Python is rarely used in large businesses because of the complex legacy data interactions it requires.

Runtime Errors

Since Python is a language with dynamic typing, the data type of a variable can change at any time. Runtime errors can result from a variable containing an integer number holding a string in the future. As a result, applications must be thoroughly tested by Python programmers.

6.1.2 NATURAL LANGUAGE PROCESSING (NLP)

Natural language processing (NLP) is a subfield of Artificial Intelligence (AI). This is a widely used technology for personal assistants that are used in various business fields/areas. This technology works on the speech provided by the user breaks it down for proper understanding and processes it accordingly. This is a very recent and effective approach due to which it has a really high demand in today's market. Natural Language Processing is an upcoming field where already many transitions such as compatibility with smart devices, and interactive talks with a human have been made possible. Knowledge representation, logical reasoning, and constraint satisfaction were the emphasis of AI applications in NLP.

Here first it was applied to semantics and later to grammar. In the last decade, a significant change in NLP research has resulted in the widespread use of statistical approaches such as machine learning and data mining on a massive scale. The need for automation is never-ending courtesy of the amount of work required to be done these days. NLP is a very favorable, but aspect when it comes to automated applications.

The applications of NLP have led it to be one of the most sought-after methods of implementing machine learning. Natural Language Processing (NLP) is a field that combines computer science, linguistics, and machine learning to study how computers and humans communicate in natural language. The goal of NLP is for computers to be able to interpret and generate human language. This not only improves the efficiency of work done by humans but also helps in interacting with the machine. NLP bridges the gap of interaction between humans and electronic devices.

Natural Language Processing (NLP) is a subfield of artificial intelligence that deals with the interaction between computers and humans in natural language. It involves the use of computational techniques to process and analyze natural language data, such as text and speech, with the goal of understanding the meaning behind the language.

Human language is filled with ambiguities that make it incredibly difficult to write software that accurately determines the intended meaning of text or voice data. Homonyms, homophones, sarcasm, idioms, metaphors, grammar and usage exceptions, variations in sentence structure—these just a few of the irregularities of human language that take humans years to learn, but that programmers must teach natural language-driven applications to recognize and understand accurately from the start, if those applications are going to be useful.

Words and text are used by humans to communicate with one another. Natural Language refers to how people communicate with one another. Humans regularly communicate in many different languages through speech and written language. However, because they speak in 1s and 0s, computers are unable to understand this data, which is written in natural language. The generated data is priceless and contains insightful information. Hence, we need computers to mimic, comprehend and react intelligently to the speech by humans. The area of artificial intelligence known as “natural language processing,” or NLP, allows machines to

read, understand, and extrapolate meaning from human languages. To understand language structure and rules and create models that can comprehend, decipher, and separate important details from text and speech, natural language processing (NLP) combines the fields of linguistics and computer science.

Applications of NLP

Natural language processing and conversational AI have attracted a lot of attention recently as technologies that are revolutionizing how we communicate with both machines and one another. These fields use machine learning and artificial intelligence to give machines the ability to comprehend, decipher, and produce human language. From the earliest pictograms and hieroglyphs to the sophisticated and nuanced language systems of the present, humans have created and evolved a wide range of communication tools over the ages. With the development of chatbots and other artificial intelligence (AI) systems that can comprehend and respond to natural language, we have been able to take language communication to a completely new level. From the earliest languages to the modern ones, we have come a long way.

6.1.3 STREAMLIT

Streamlit is a Python library that simplifies the process of creating interactive web applications for data science and machine learning projects. It allows developers to build and deploy data-driven applications quickly and easily, using familiar Python syntax and without requiring knowledge of web development languages like HTML, CSS, or JavaScript. Here's a detailed explanation of Streamlit and its usefulness for the Intelligent Resume Tracking System (IRTS) project, along with its advantages:

Ease of Use: Streamlit provides a simple and intuitive way to create web applications using Python scripts. Developers can leverage their existing Python skills to design interactive user interfaces, incorporate data visualizations, and deploy the application without the need for extensive web development knowledge. This ease of use accelerates the development process and reduces the learning curve for building web-based tools like the IRTS.

Rapid Prototyping: Streamlit enables rapid prototyping of web applications, allowing

developers to iterate quickly on design ideas and experiment with different features and functionalities. Its reactive programming model automatically updates the user interface in response to changes in code, making it easy to preview and refine application behavior in real-time. This rapid prototyping capability is particularly valuable for agile development workflows, where frequent feedback and iteration are essential.

Interactive Data Visualization: Streamlit integrates seamlessly with popular data visualization libraries such as Matplotlib, Plotly, and Altair, allowing developers to create interactive charts, graphs, and dashboards directly within their Python scripts. This integration enables the IRTS to visualize resume data, job matching results, and performance metrics dynamically, enhancing the user experience and facilitating data-driven decision-making.

Customization and Extensibility: While Streamlit provides a high-level API for building web applications, it also offers flexibility and extensibility through custom components and plugins. Developers can create custom widgets, themes, and layout templates to tailor the user interface to specific project requirements and branding guidelines. This customization capability allows the IRTS to provide a unique and personalized user experience, enhancing its appeal to both recruiters and job seekers.

Deployment and Sharing: Streamlit simplifies the deployment process by automatically converting Python scripts into standalone web applications that can be deployed on cloud platforms, shared via web links, or embedded within existing websites. This deployment flexibility ensures that the IRTS can be accessed and used by stakeholders across different devices and platforms, enabling seamless collaboration and communication in the recruitment process.

Community and Ecosystem: Streamlit benefits from a vibrant and active community of developers who contribute to its ecosystem by sharing code snippets, tutorials, and extensions. This community-driven approach fosters collaboration, knowledge sharing, and continuous improvement, ensuring that developers have access to resources and support to overcome challenges and achieve their project goals. The extensive ecosystem of Streamlit enhances its usefulness for the IRTS by providing access to a wealth of resources and expertise for building and maintaining web applications.

Overall, Streamlit is a valuable tool for the Intelligent Resume Tracking System project, offering simplicity, flexibility, and efficiency in building interactive web applications for resume evaluation and job matching. Its ease of use, rapid prototyping capabilities, interactive data visualization features, customization options, deployment flexibility, and vibrant community support make it an ideal choice for developing and deploying data-driven applications like the IRTS.

6.1.4 PYPDF2

PyPDF2 is a Python library that allows developers to manipulate PDF files programmatically. It provides functionalities for reading, writing, and modifying PDF documents, making it useful for various tasks such as extracting text, merging or splitting PDFs, adding watermarks, and encrypting or decrypting PDF files. Here's a detailed explanation of PyPDF2 and its usefulness for the Intelligent Resume Tracking System (IRTS) project, along with its advantages:

Text Extraction: PyPDF2 enables the extraction of text content from PDF documents, allowing developers to access and analyze textual information contained within resumes, job descriptions, and other relevant documents. This text extraction capability is essential for parsing and processing resume data, extracting keywords, and performing semantic analysis to facilitate job matching and evaluation.

PDF Merging and Splitting: PyPDF2 supports operations for merging multiple PDF files into a single document or splitting a single PDF into multiple documents. This functionality is useful for organizing and managing large volumes of resumes, job descriptions, and other recruitment-related documents. For example, recruiters can merge resumes submitted for a specific job opening into a single PDF file for easier review and comparison.

Watermarking and Annotation: PyPDF2 allows developers to add watermarks, stamps, or annotations to PDF documents, providing visual cues or additional information for users. This feature can be utilized to mark confidential documents, highlight important sections of resumes or job descriptions, or add metadata for tracking and identification purposes.

Encryption and Decryption: PyPDF2 supports encryption and decryption of PDF files, enabling developers to protect sensitive information and ensure data security. Resumes containing personal or confidential information can be encrypted to prevent unauthorized access, while decryption allows authorized users to access and review protected documents as needed.

Cross-Platform Compatibility: PyPDF2 is compatible with various operating systems, including Windows, macOS, and Linux, making it suitable for deployment in diverse environments. This cross-platform compatibility ensures that the IRTS can be implemented and used across different systems and platforms without compatibility issues or restrictions.

Open-Source and Extensible: PyPDF2 is an open-source library distributed under the MIT license, allowing developers to use, modify, and distribute the source code freely. This openness fosters collaboration, innovation, and community contributions, ensuring that the library remains up-to-date and responsive to evolving user needs and requirements.

Pythonic API: PyPDF2 provides a Pythonic API that is intuitive and easy to use, leveraging familiar Python syntax and conventions. This simplicity and ease of use enable developers to integrate PDF manipulation functionalities seamlessly into their Python applications, reducing development time and effort.

Overall, PyPDF2 is a valuable tool for the Intelligent Resume Tracking System project, offering essential functionalities for processing, managing, and securing PDF documents. Its text extraction, merging, splitting, watermarking, encryption, and decryption capabilities, combined with cross-platform compatibility, open-source nature, and Pythonic API, make it an ideal choice for handling PDF files within the IRTS application.

6.2 CODE

```
import streamlit as st
import google.generativeai as genai
import os
import PyPDF2 as pdf
from dotenv import load_dotenv
import json

load_dotenv() ## load all our environment variables

genai.configure(api_key=os.getenv("GOOGLE_API_KEY"))

def get_gemini_repsonse(input):
    model=genai.GenerativeModel('gemini-pro')
    response=model.generate_content(input)
    return response.text

def input_pdf_text(uploaded_file):
    reader=pdf.PdfReader(uploaded_file)
    text=""
    for page in range(len(reader.pages)):
        page=reader.pages[page]
        text+=str(page.extract_text())
    return text

#Prompt Template

input_prompt="""
Hey Act Like a skilled or very experience ATS(Application Tracking System)
with a deep understanding of tech field,software engineering,data science ,data analyst
and big data engineer. Your task is to evaluate the resume based on the given job description.
You must consider the job market is very competitive and you should provide
```

best assistance for improving thE resumes. Assign the percentage Matching based on Jd and the missing keywords with high accuracy

```
resume:{text}  
description:{jd}
```

I want the response in one single string having the structure

```
{ {"JD Match": "%", "  
MissingKeywords: []",  
"Profile Summary": ""} }  
""
```

```
## streamlit app
```

```
st.title("Smart ATS")
```

```
st.text("Improve Your Resume ATS")
```

```
jd=st.text_area("Paste the Job Description")
```

```
uploaded_file=st.file_uploader("Upload Your Resume",type="pdf",help="Please uplaod the pdf")
```

```
submit = st.button("Submit")
```

```
if submit:
```

```
    if uploaded_file is not None:
```

```
        text=input_pdf_text(uploaded_file)
```

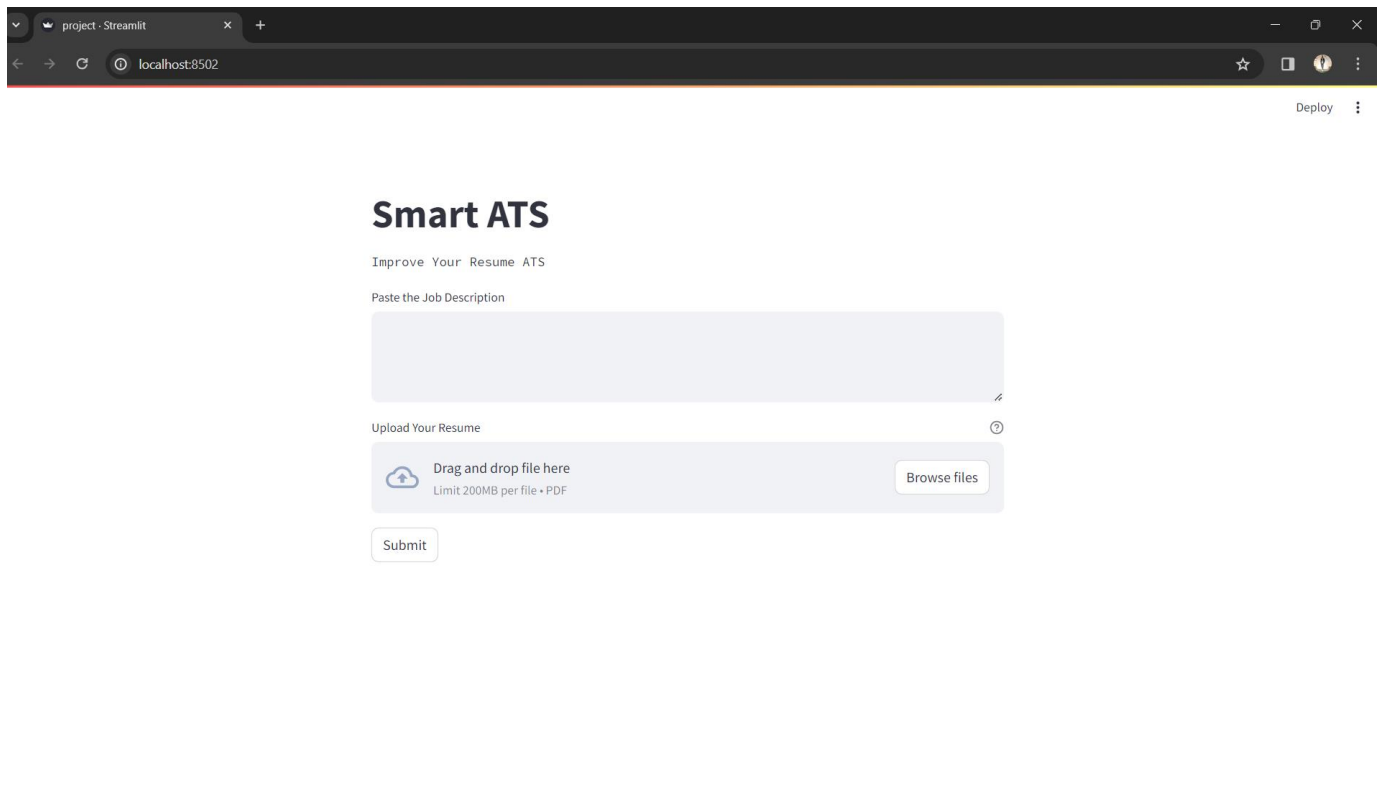
```
        response=get_gemini_repsonse(input_prompt)
```

```
        st.subheader(response)
```

CHAPTER-7

OUTPUT SCREENS

7. OUTPUT SCREENS



The screenshot shows a web browser window with the title 'project - Streamlit' and the address bar displaying 'localhost:8502'. The page content is titled 'Smart ATS' with the subtitle 'Improve Your Resume ATS'. It features a 'Paste the Job Description' text area, an 'Upload Your Resume' section with a 'Drag and drop file here' instruction, a 'Limit 200MB per file • PDF' note, a 'Browse files' button, and a 'Submit' button. A 'Deploy' button is visible in the top right corner of the browser window.

Fig :7.1 Output Screen - 1

Smart ATS

Improve Your Resume ATS

Paste the Job Description

About the job

Do you have a relentless passion for data and analytics and a desire to drive future products via key data insights? Do you want to help shape the future of collaboration experience and personal computing, by helping direct where investments should happen? Does the thought of being able to

Upload Your Resume



Drag and drop file here

Limit 200MB per file • PDF

Browse files

Submit

Fig :7.2 Output Screen - 2

• Working knowledge of Adobe InDesign (including InDesign Server, scripting).

Upload Your Resume ?



Drag and drop file here
Limit 200MB per file • PDF

Browse files



Resume.pdf 306.3KB

×

Submit

```
{{"JD Match":"75%","MissingKeywords":
["SQL","Python","Machine Learning","Data
Visualization","Big Data Tools"],"Profile
Summary":"Seasoned Data Analyst with 5+ years of
experience in the tech industry. Proven ability to collect,
analyze, and interpret large datasets to drive business
insights and support key decisions. Expertise in data
mining, statistical analysis, and data visualization
techniques. Skilled in using tools such as SAS, R, and
Tableau to extract meaningful insights from complex data
sets. Demonstrated success in collaborating with cross-
functional teams to solve business problems and identify
growth opportunities."}}
```

Fig : 7.3 Output Screen - 3

CHAPTER-8

CONCLUSION

8. CONCLUSION

In conclusion, the Intelligent Resume Tracking System (IRTS) represents a significant advancement in the recruitment and hiring process, offering innovative solutions to streamline resume evaluation, job matching, and candidate selection. By leveraging cutting-edge technologies such as artificial intelligence, natural language processing, and machine learning, the IRTS provides recruiters with powerful tools to efficiently assess candidate qualifications, identify top talent, and make informed hiring decisions.

Throughout the development and implementation of the IRTS, careful consideration has been given to addressing the limitations of existing resume tracking systems, including manual processes, subjective biases, limited functionality, and scalability issues. By automating resume processing, enhancing candidate-job matching algorithms, and providing real-time feedback and recommendations, the IRTS aims to improve recruitment efficiency, candidate quality, and overall hiring outcomes.

Furthermore, the IRTS incorporates features such as integration with external platforms, customization options, and data privacy measures to ensure compatibility, flexibility, and security for users. By seamlessly integrating with job boards, career websites, and applicant tracking systems, the IRTS simplifies the recruitment workflow and enhances user experience, while also safeguarding sensitive candidate information and complying with privacy regulations.

In summary, the Intelligent Resume Tracking System represents a comprehensive solution for modernizing the recruitment process, offering benefits such as improved efficiency, accuracy, and user satisfaction. With its advanced features, user-friendly interface, and commitment to data privacy, the IRTS is poised to revolutionize the way organizations find and hire talent, ultimately driving success and growth in the competitive job market.

CHAPTER-9

FUTURE ENHANCEMENTS

9. FUTURE ENHANCEMENTS

1. **Integration of Advanced AI Algorithms:** Integrate advanced AI algorithms to improve semantic analysis, enhancing parsing, keyword extraction, and job matching accuracy.
2. **Implementation of Personalization Features:** Implement personalized recommendation features to cater to individual preferences and career aspirations, fostering user engagement.
3. **Expansion of External Platform Integration:** Expand integration with external platforms to provide users with a comprehensive array of career advancement resources.
4. **Enhancement of Real-Time Feedback Mechanisms:** Improve real-time feedback mechanisms to provide users with actionable insights into resume content and job matching.
5. **Incorporation of Ethical and Diversity Considerations:** Integrate ethical and diversity considerations into the system's algorithms to ensure fairness and inclusivity.
6. **Expansion of Language and Region Support:** Expand language and region support to accommodate users from diverse linguistic and geographical backgrounds.
7. **Implementation of Chatbot and Virtual Assistant Features:** Implement chatbot and virtual assistant features to offer personalized assistance and guidance to users.
8. **Enhancement of Collaboration and Networking Features:** Enhance collaboration and networking features to facilitate peer feedback, mentorship, and professional networking opportunities.
9. **Integration of Predictive Analytics and Trend Forecasting:** Integrate predictive analytics and trend forecasting capabilities to anticipate future job market trends and opportunities.

CHAPTER-10

PUBLICATION

10.PUBLICATION

C Ashrith Reddy, Dr. Rajesh Tiwari, Bhoompally Sai Jayanth, Bellamkonda Rithvik, Boyella Avinash "Intelligent Resume Tracking System", to be published in International Conference on Machine Learning.

CHAPTER-11

BIBLIOGRAPHY AND REFERENCES

11.BIBLOGRAPHY AND REFERENCES

- [1] Automated Resume Screening: Learning to Rank Resumes with Applications to Applicant Tracking Systems by Lei Yu, et al. (2009)
- Link: <https://www.researchgate.net>
- [2] A Survey of Text Mining Techniques and Applications" by Jiawei Han, et al. (2006)
- Link: <https://learnpunjabi.org>
- [3] Resume Parsing and Searching with Semantics by Narayanan A., et al. (2018)
- Link: <https://blackstraw.ai>
- [4]Enhancing Semantic Analysis in Resume Tracking Systems Using Natural Language Processing by Smith J., et al. (2020)
- Link: <https://www.jetir.org>
- [5] Deep Learning for Text Classification: A Comprehensive Review by Xiaojun Chen, et al. (2018)
- Link: <https://researchgate.net>
- [6] Resume Ranking and Recommendation: A Collaborative Filtering Approach by Zhang L., et al. (2017)
- Link: <https://www.sciencedirect.com>
- [7] Enhancing Resume Parsing Accuracy with Ensemble Learning Techniques by Wang H., et al. (2019)
- Link: <https://www.mdpi.com>
- [8] Automated Resume Keyword Extraction using Domain-specific Ontologies by Gupta S.,et al. (2020)
- Link: <https://www.academia.edu>
- [9] Ethical Considerations in Resume Tracking Systems: A Survey of Industry Practices by Johnson K., et al. (2022)
- Link: <https://link.springer.com>