

Resume Analyzer Using Natural Language Processing (NLP)

A major project report submitted in partial fulfillment of the requirement for
the award of degree of

Bachelor of Technology
in
Computer Science & Engineering / Information Technology

Submitted by
Vansh Gulati 201109
Ishita Gupta 201132

Under the guidance & supervision of

Mr. Faisal Firdous
Mr. Ramesh Narwal



**Department of Computer Science & Engineering and
Information Technology**
Jaypee University of Information Technology,
Waknaghat, Solan - 173234 (India)

CERTIFICATE

This is to certify that the work which is being presented in the project report titled '**Resume Analyzer Using Natural Language Processing (NLP)**' in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science And Engineering and submitted to the Department of Computer Science And Engineering, Jaypee University of Information Technology, Waknaghat is an authentic record of work carried out by Vansh Gulati(201109) & Ishita Gupta (201132)" during the period from July 2023 to May 2024 under the supervision of **Mr. Faisal Firdous**, Department of Computer Science and Engineering/Information Technology, Jaypee University of Information Technology, Waknaghat and **Mr. Ramesh Narwal**, Department of Computer Science and Engineering/ Information Technology, Jaypee University of Information Technology, Waknaghat

Vansh.
Vansh Gulati
201109

Ishita
Ishita Gupta
201132

The above statement made is correct to the best of my knowledge.

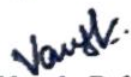
Faisal
16-5-24
Mr. Faisal Firdous
(Supervisor)
Assistant Professor,
Department of Computer Science &
Engineering and Information Technology
Dated:

Ramesh Narwal
10-5-2024
Mr. Ramesh Narwal
(Co-supervisor)
Assistant Professor,
Department of Computer Science &
Engineering and Information Technology
Dated:

CANDIDATE'S DECLARATION

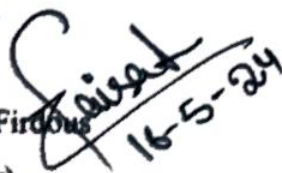
I hereby declare that the work presented in this report entitled '**Resume Analyzer Using Natural Language Processing (NLP)**' in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science & Engineering / Information Technology** submitted in the Department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology, Waknaghat is an authentic record of my own work carried out over a period from August 2023 to May 2024 under the supervision of **Mr. Faisal Firdous** (Assistant Professor, Department of Computer Science & Engineering and Information Technology) and **Mr. Ramesh Narwal** (Assistant Professor, Department of Computer Science & Engineering and Information Technology)

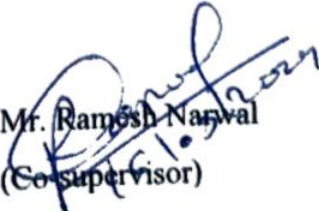
The matter embodied in the report has not been submitted for the award of any other degree or diploma.


Vansh Gulati
201109


Ishita Gupta
201132

This is to certify that the above statement made by the candidate is true to the best of my knowledge.


Mr. Faisal Firdous
(Supervisor)
Assistant Professor,
Department of Computer Science &
Engineering and Information Technology


Mr. Ramesh Narwal
(Co-supervisor)
Assistant Professor,
Department of Computer Science &
Engineering and Information Technology

ACKNOWLEDGEMENT

Firstly, we express my heartiest thanks and gratefulness to almighty God for His divine blessing makes it possible to complete the project work successfully. We are really grateful and wish my profound indebtedness to Supervisors **Mr. Faisal Firdous** (Assistant Professor, Department of Computer Science & Engineering/Information Technology) and **Mr. Ramesh Narwal** (Assistant Professor, Department of Computer Science & Engineering/Information Technology) their endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stages have made it possible to complete this project. We would like to express our heartiest gratitude to Mr. Faisal Firdous, Department of Computer Science & Engineering/Information Technology and Mr. Ramesh Narwal, Department of Computer Science & Engineering/Information Technology for their kind help to finish my project. We would also generously welcome each one of those individuals who have helped me straightforwardly or in a roundabout way in making this project a win. In this unique situation, we might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated my undertaking.

Finally, we must acknowledge with due respect the constant support of our parents.



Vansh Gulati

201109

Computer Science & Engineering and
Information Technology,

Jaypee University of Information
Technology, Wagnaghat



Ishita Gupta

201132

Computer Science & Engineering and
Information Technology,

Jaypee University of Information
Technology, Wagnaghat

CONTENTS

CERTIFICATE	i
DECLARATION	ii
ACKNOWLEDGEMENT	iii
CONTENTS	iv- v
ABSTRACT	x
CHAPTER 1: INTRODUCTION	1-4
1.1 INTRODUCTION	1
1.2 PROBLEM STATEMENT	2
1.3 OBJECTIVE	3
1.4 SIGNIFICANCE AND MOTIVATION	3
1.5 ORGANIZATION	4
CHAPTER 2: LITERATURE SURVEY	5-9
2.1 LITERATURE SURVEY	5
2.2 KEY GAPS	8
CHAPTER 3: SYSTEM DEVELOPMENT	10-26
3.1 REQUIREMENTS	10
3.2 PROJECT DESIGN AND ARCHITECTURE	14
3.3 DATA PREPERATION	19
3.4 IMPLEMENTATION	21
3.5 KEY CHALLENGES	25
CHAPTER 4: TESTING	27-31
4.1 TESTING STRATEGY	27
4.2 TEST CASES AND OUTCOMES	29
CHAPTER 5: RESULTS AND EVALUATION	32-39
5.1 RESULTS	32

CHAPTER 6: CONCLUSION AND FUTURE SCOPE	41-42
6.1 CONCLUSION	41
6.1 FUTURE SCOPE	41
REFERENCES	43-45

LIST OF TABLES

S. No.	Title	Page No.
1	Literature Survey for Resume Analyzer Using NLP	5

LIST OF FIGURES

S. No.	Title	Page No.
1	Flow of the app	15
2	Flow of the Resume Analyzer	16
3	Flow of Personality Insights	16
4	Flow of Resume Builder	17
5	Flow of Resume Matching	18
6	Environment setup	21
7	Data preparation	22
8	Prompts for LLM to execute	22
9	Resume Builder execution	23
10	Resume Matcher Implementation	23
11	Cover Letter generation	23
12	Personality Insights generator	24
13	User Interface	24
14	Result of Test Case no. 1	30
15	Result of Test Case no. 2	31
16	Result of Test Case no. 3	31
17	User uploaded resume	32
18	Details and Skills extraction from resume	33
19	Skills and course recommendation	33
20	Resume Grading and insights	34

21	URL input without providing information	34
22	Generated Resume without any provided information	35
23	URL input with information	35
24	Generated Resume with provided information	36
25	Generated Cover Letter	36
26	Resume and Link input	37
27	Result when resume is not a match	37
28	Result when resume is a match	38
29	Resume Input	38
30	Graph for Behavioral Factors	39
31	Graph for OCEAN Personalities	39
32	Generated personality insights	40

LIST OF ABBREVIATIONS

ABBREVIATIONS	MEANING
NLP	Natural Language Processing
NLTK	Natural Language Toolkit
UI/UX	User Interface/User Experience
LLM	Large Language Model

ABSTRACT

It's becoming more and more important for job searchers to be able to expertly tailor their resumes for every application to further their careers in the fast-paced industry. The purpose of this project is to change the way resume analysis and job profile matching is done by introducing the use of the Natural Language Processing (NLP) techniques, mainly the Large Language Models (LLMs). The conventional methods of resume data extraction and job matching have used only NLP, which is the reason why they have not been able to achieve the best accuracy and diversity. The use of LLMs to improve the quality and the quantity of the insights that are extracted from the resumes, thus, the LLMs will be able to match the candidate profiles with the job requirements more precisely.

Our approach consists of several important aspects. Firstly, we use NLP to extract the necessary data from the resumes, which will guarantee a full data capture. Thus, by applying resume matching, we improve the match of job profiles and candidate can easily find jobs whose skills matches with their profiles. In addition, our application has a feature of resume builder that helps to make resumes suitable for job roles by means of the information extracted and the analysis of the personality. Through the resolution of the existing methodologies gap, our project establishes a new level of resume analysis and job matching and thus, enables the candidates to find their way to their preferred job.

CHAPTER 1

INTRODUCTION

1.1 Introduction

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

Using natural language processing (NLP) and open source LLM, the Resume Analyzer reimagines the resume customization process and situates it at the intersection of cutting-edge technology and professional advancement. The tool makes predictions about a candidate's intended career job using advanced natural language processing (NLP) techniques, going beyond resume analysis. It also looks at what NLP does during evaluation of resumes to gather qualifications, credentials, experience, and skills. It is based on the NLP breakthrough, an advanced paradigm in technological development, which makes it suitable for the Resume Analyzer.

Resume Analyzer employs natural language processing machine learning techniques, which provide detailed analysis of the textual data in resumes. The system shall analyze information related to the applicant's background data to find out compatible job positions. The system is accurate at arriving at the right employment function for an individual based on detailed scrutiny of job descriptions and the incorporation of industry terminologies. Now with the addition of machine learning, the Resume Analyzer is much more than a statistical tool. A prediction machine that the company claims could alter how prospective employees think about their future tracks of careers.

The Resume Analyzer is also forecasted as it promotes specific qualifications that are in line with the required job description. This system takes into consideration a number of factors relevant to the candidate's competencies available in the markets and the general state of the industry. In addition to task prediction, the objective gives users useful information and directs them toward

certifications that might enhance their standing in the field. By positioning the Resume Analyzer as a holistic career development tool, this proactive approach fosters a culture of continuous learning and skill development.

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

1.2 Problem Statement

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

A creative resume analyzer that goes above and beyond conventional procedures is much needed to handle these two problems. This application requires the use of contemporary Natural Language Processing (NLP) techniques in order to assess job obligations and determine the best place to work. In addition to serving as an assessment tool, the Resume Analyzer ought to be a pro-active collaborator for experts, providing astute suggestions for certificates that correspond with the designated job description. Because it enables job searchers to proactively expand their skill sets and manage the intricacies of the modern labor market, this proactive component is essential.

The intended Resume Analyzer turns into a strategic ally for job seekers navigating the modern labor market, rather than just a useful tool. The application addresses the first aspect of the issue by guaranteeing a thorough evaluation of resume content through the use of sophisticated natural language processing (NLP) techniques. The algorithm takes care of the second dimension at the

same time by accurately determining the ideal employment position and recommending credentials that improve the professional profile. Having access to a proactive and all-inclusive solution like these aids in career development and helps job seekers better handle the demands of the modern labor market.

The creation of a proactive and intelligent Resume Analyzer that integrates state-of-the-art natural language processing algorithms, predictive features, and customized certification suggestions is ultimately required to address this issue. In doing so, this instrument turns into a priceless tool that aids job seekers in effectively and mindfully navigating the complexities of the employment market.

1.3 Objectives

- Implement NLP to extract data like personal information, skills, education and work experience from the resume.
- Compare multiple prompts to get the optimal result from the Large Language Model.
- To verify whether the application process offers candidates instant feedback while they upload their resumes.

1.4 Significance and Motivation of the Project Work

The Resume Analyzer project is notable since its objective is to improve the job application process. These days, a tool like this one is needed to accurately assess some of a candidate's skills because many employment marketplaces are very competitive. This project aims to assist people in properly showcasing themselves through polished narrative in order to improve their chances of landing desired positions.

The challenges people have when attempting to prepare resumes served as the inspiration for the notion. Many people find it challenging to identify critical abilities and align them with market demands. This is when the Resume Analyzer comes in handy. By providing tailored guidance and recommendations, it reduces this strain and increases the user's sense of comfort and empowerment during the job application process.

The second objective of the project is to advance an individual's professional growth. The Resume Analyzer helps people advance their careers and stay competitive by recommending relevant qualifications and adapting to changing market conditions. In the end, it'll be like having a really

helpful professional success travel companion, with practical advice that directly affects job seekers' chances of getting hired.

1.5 Organization of Project Report

There are several chapters in this report. Chapter 1 gives a brief introduction of the Resume Analyzer using NLP techniques. Chapter 2 discusses the research work done to get an insight of the work that has been done in the field. Chapter 3 highlights the requirements and system development of the project. Chapter 4 gives the overview of the testing and test cases used during the making of the project. Chapter 5 presents the results of the project and its evaluation. Chapter 6 discusses the conclusion of the work done and the future scope of the project.

CHAPTER 2

LITERATURE SURVEY

2.1 Literature Survey

S. No.	Author and Year	Tools/Techniques/Dataset	Methodology	Results	Limitations
1	Gan, C., Zhang, Q., & Mori, T. (2024a, January 16).	Natural Language Processing, Open-source LLMs, Conditional Random Fields, Vectorization Models	Focuses on leveraging LLM agents for various stages of the automated resume screening process, from sentence classification to decision-making, with the aim of enhancing efficiency and effectiveness in candidate selection.	These results collectively showcase the effectiveness, efficiency, and performance improvements achieved through the implementation of LLM agents in the automated resume screening framework.	The limitations of the automated resume screening framework based on LLM agents include data compromise risk, varying model capabilities, reliance on fine-tuning, resource-intensive human feedback requirement, performance variability, and the need for customization to generalize across industries.
2	Jagwani V, Meghani S, Pai K, Dhage S. Published July 28, 2023.	SpaCy, Latent Dirichlet Allocation model	Uses SpaCy's Named Entity Recognition to extract relevant entities and LDA model to rate resumes by assigning topic probabilities to each entity.	Proposed method successfully extracts relevant entities and assigns topic probabilities to each entity using LDA.	Doesn't provide diversity over multiple career domains
3	Bhor, S., Gupta, V.,	Neural Network,	Utilized Neural Network and	Proposed model aims to	Important to note the

	Nair, V., Shinde, H. and Kulkarni, M.S., 2021. Int J Res Eng Sci (IJRES), 9(6), pp.01-06.	Conditional Random Fields (CRF), Convolutional Neural Network (CNN), Optical Character Recognition (OCR)	CNN for segmentation and extraction of information from resume. Named Entity Recognition (NER) used to understand domain-specific jargon and words	extract details and statistics from resumes and rank them based on company preference and requirements. Output JSON contains various extraction information like personal, educational details etc.	accuracy of the NLP technique used, as the effectiveness of the resume parser heavily relies on it. The performance may also be influenced by the quality and format of the resume being parsed.
4	Daryani, C., Chhabra, G.S., Patel, H., Chhabra, I.K. and Patel, R., 2020. ETHICS AND INFORMATION TECHNOLOGY [Internet]. VOLKSON PRESS, pp.99-103.	Natural Language Processing Toolkit, Term-Frequency-Inverse Document Frequency, Cosine-Similarity matrix	Uses Natural Language Processing to extract relevant information and cosine-similarity matrix to match resume with job description.	Presents an automated resume screening system that simplifies the recruitment process for the organizations.	Does not address the limitations of using NLP in accurately capturing candidate's qualifications
5	Green, N., Liu, M. and Murphy, D., 2020. Information Systems Education Journal, 18(3), pp.28-37.	NLTK, Term-Frequency-Inverse Document Frequency, Cosine-Similarity	Uses Natural Language Processing to extract relevant information and cosine-similarity matrix to match resume with job description.	Presents a tool that utilizes NLP that analyses resumes and produces visual reports which highlights areas for improvement.	Does not address the biases of using NLP techniques
6	Das, P., Pandey, M. and	Big data tools, R language	Research work discusses the text analysis	Text preprocessing methods are	The paper does not mention any specific

	Rautaray, S.S., 2018. IJ Information Technology and Computer Science, 9, pp.21-31.		process and the extraction of entities using different big data tools	used to handle unstructured and ambiguous data	limitations or challenges faced during the implementation of the resume parser.
7	Sanyal, S., Hazra, S., Adhikary, S. and Ghosh, N., 2017. International Journal of Engineering Science, 4484.	Natural Language Processing (NLP), NLTK ToolKit, NoSQL Database	Utilized Natural Language Processing (NLP) techniques to parse and extract information from resume and store it in a NoSQL database for storage and retrieval	Proposed method successfully extracts relevant entities and storing it in a NoSQL database	The paper does not mention any specific limitations or challenges faced during the implementation of the resume parser.

Table 1: Literature Survey for Resume Analyzer Using NLP

Conclusion

The literature review highlights the evolution of resume screening methodologies, emphasizing the integration of Natural Language Processing (NLP) techniques for efficient candidate selection. While various models demonstrate successful extraction and analysis of resume data, limitations persist, including NLP bias, model accuracy, and domain-specific applicability. Future advancements should focus on mitigating biases, enhancing model accuracy, and ensuring adaptability across diverse industries. Overall, the reviewed studies provide valuable insights into the application of NLP in recruitment processes, paving the way for improved candidate evaluation and organizational efficiency.

2.2 Key Gaps in the Literature

The literature on automated resume screening and parsing has a few significant gaps and areas for possible improvement, based on the summaries that have been offered. The following are some observations:

A Basic Discussion of Ethical Issues: While Natural Language Processing (NLP) is mentioned by Chirag Daryania, Gurneet Singh Chhabra, Harsh Patel, Indrajeet Kaur Chhabra, Ruchi Patel [1], the ethical issues surrounding automated resume screening are not discussed. Crucial elements that must be considered and covered in the literature are ethical considerations, such as potential discrimination and bias in NLP models.

Insufficient Diversity in Career Fields: According to the second paper, it doesn't offer diversity across a range of career fields. In order to create inclusive hiring procedures, diversity is crucial. Scholars' ought to investigate strategies for augmenting the generalizability of their approaches across diverse businesses and occupational domains.

Lack of Restrictions Talk: Many publications don't specifically address the restrictions or difficulties encountered while putting their resume parsing algorithms into use. It is vital to acknowledge and deliberate upon the constraints in order to comprehend the pragmatic suitability and possible drawbacks of the suggested techniques.

Comparative analysis and evaluation metrics: While the articles describe the tools, approaches, and methodologies used, they do not go into great length about the assessment criteria that are used to gauge how well the suggested systems operate. A more comprehensive understanding of the efficacy of the suggested models may be obtained by conducting comparative evaluations with current practices or benchmarks.

NLP Technique Biases: Certain publications don't discuss the biases that come with using NLP approaches. Unfair results might arise from biases in language models, particularly in the recruitment process. An essential component of study in this field should be debating methods to reduce biases and enhance fairness.

Combining Big Data Methods and Tools: Papiya Das, Manjusha Pandey, Siddharth Swarup Rautaray[6] talk about using big data tools, but it doesn't go into great detail about how these tools work with the resume parsing paradigm. Researchers and practitioners may find great insights by comprehending the integration process and the advantages of using big data techniques.

Unification of NLP Methodologies: Although NLP approaches are used in many studies, there should be criteria or standards for using NLP in resume processing. This entails dealing with concerns about generalization across various resume formats, training data quality, and model correctness.

Impact throughout Time and Adaptability: Discussions about the long-term effects of automated resume screening systems and how flexible they are to changing patterns in the job market might be beneficial to the literature. Given that job needs are dynamic, it is imperative to investigate the potential for these systems to be updated and modified in the future.

CHAPTER 3

SYSTEM DEVELOPMENT

3.1 Requirements

In this stage we are going to list all the requirements that are needed in the model. The various Tools, Technologies and Libraries which are used.

Python: The Basis for Readability and All-Inclusive Libraries

Python is a popular, advanced, and versatile programming language. Its vast libraries and ease of reading have contributed to its global appeal. It is quite convenient for many operations like data analysis, web-building etc.

- **Readability:** The readability and relative ease at which to write and maintain coding for developers is highly emphasized in the syntax of Python, which plays an important role herein. This feature will be significant in large codebases and collaborative projects.
- **Large Library:** The enormous library and framework ecosystem within python is what makes it possible for the tool to fulfill multiple demands. These include TensorFlow, PyTorch, Django, Flask, NumPy, Pandas and many others. Development is made easier as these libraries are accessible while giving programmers strong instruments.

Streamlit: Making the Development of Web Applications Simpler

It is an open-source python package that was developed to simplify design and development of dashboards as well as web apps. It is good at developing application programs with focus on interaction and data presentation.

- **Ease of Use:** Developers can design web applications with little difficulty thanks to Streamlit's user-friendly interface. It offers a straightforward script-based method that expedites the design and prototyping process.
- **Data Visualization:** Pre-built components available in the library can help enhance the development of dynamic and attractive visuals. Therefore, it becomes ideal for scenarios where sharing information through graphs and bar charts is crucial.
- **Data-Driven Apps:** This allows developers to create data-focused apps much faster thanks to streamlit. Therefore, it is an ideal choice for scenarios that want to show trends and patterns in data.

-

Text Analysis's Untapped Potential: The Natural Language Toolkit (NLTK)

A special Python package for use in NLP known as NLTK, which stands for Natural Language ToolKit. The program provides numerous functionalities, including sentiment analysis, stemming, and tokenization, to name some.

- **Text Analysis capabilities:** NLTK stands out as one of the top NLP tools because it has numerous capabilities for text analysis. For example, developers can employ features such as word tokenization, which breaks down text into words, stemming, reduces words back to their root form, among others.
- **Sentiment Analysis:** NLTK allows developers to determine whether a particular message is positive, negative, or neutral using sentiments analysis. It's especially essential when we talk about apps and the need of understanding user feedback.
- **Natural Language Processing:** Enhanced computational abilities of a computer using NLTK will be possible which makes it possible for programmers to create computers that can understand, read and produce text that sounds as if a human wrote it.

spaCy: Robust Advanced Natural Language Understanding

Another widely used tool in text analysis and natural language processing is the python module spaCy. This software is known for being effective, having pre-trained models, and supporting various tasks such as named entity recognition, tokenization, and part-of-speech tagging.

- **Efficiency and Speed:** SpaCy is commonly recognized to be fast in handling NLP tasks. It has been optimized for production use which will run large amount of text data using minimal resources.
- **Pre-Trained Models:** Developers can use pre-trained models for frequent tasks without requiring extensive training by applying the library's pre-trained models to a variety of NLP tasks. This guarantees consistent performance and shortens the development time.
- **Tokenization and Named Entity identification:** spaCy is a useful tool for applications requiring in-depth linguistic study since it is proficient in both named entity identification (identifying entities such as names, localities, etc.) and tokenization (splitting text into meaningful units).

PDFMiner: Merging Data from PDF Documents

A Python library called PDFMiner is used to extract text and structured data from PDF documents. Since many diverse sectors demand data extraction and analysis from PDF files, it is imperative to make this process easier.

- **Text Extraction:** Textual data can be handled and studied inside PDF files thanks to PDFMiner's ability to extract text content from PDF documents.
- **Structured Data Extraction:** PDFMiner can extract structured data from PDFs in addition to text. This is especially helpful for PDF documents that contain data in tabular or sorted formats.
- **Data analysis from PDFs:** Developers can extract insightful information from PDF documents by integrating PDF data into their analytical pipelines using PDFMiner.

PyResparser: Resume Parsing to Simplify Recruiting Procedures

A custom PyResparser module was created to parse and extract data from CVs and resumes. It is a useful tool for streamlining the hiring process and enhancing the effectiveness of candidate evaluation.

- **Resume Parsing:** PyResparser was developed specifically to parse resumes and retrieve vital data, including contacts, work history, education, and talents. The amount of manual labor required to review resumes is greatly decreased by this automation.
- **Structured Data Output:** Integrating the collected data into databases or application tracking systems is made easy by the library's structured data output. This guarantees unrestricted communication between parties during the recruiting process.
- **Recruitment efficacy:** PyResparser boosts recruitment efficacy by automating the preliminary stages of candidate evaluation. While the library compiles pertinent resume data, recruiters can concentrate on more complex job

Gemini API: LLM Model for better results

The Gemini API is a powerful platform made to enable traders and developers to programmatically include trading features and enable smooth connection with the Gemini cryptocurrency exchange.

- **Access Market Data:** Retrieve real-time market data for informed decision-making.
- **Flexibility:** Offers a versatile solution for building custom trading bots, implementing trading strategies, and creating financial applications.

- **Enhanced Efficiency:** Enables users to navigate the complexities of cryptocurrency trading with ease, improving overall efficiency and productivity.

BeautifulSoup4: Web scrapping for better job matching

BeautifulSoup4 is a flexible Python module made to make online scraping easier and more accessible for developers to extract data from XML and HTML sources.

- **Parse HTML and XML:** Developers can parse web content and extract structured data using BeautifulSoup4's intuitive parsing methods, making it easy to navigate and extract information from complex document structures.
- **Navigate Document Structure:** The library provides tools for traversing the hierarchical structure of HTML and XML documents, enabling developers to locate and extract specific elements or data points efficiently.
- **Extract Information:** BeautifulSoup4 offers powerful methods for extracting information from web pages, including find(), find_all(), and select(), allowing developers to retrieve relevant data with ease.

Humantic API: Personality insights of the candidate according to resume

The Humantic API is designed to analyze text data using advanced artificial intelligence and natural language processing (NLP) techniques, with a specific focus on sentiment analysis and emotion recognition. Here's what the Humantic API does:

- **Sentiment Analysis:** The API evaluates text data to determine the sentiment expressed within it. It can classify text as positive, negative, or neutral based on the language used and the overall tone of the content.
- **Emotion Recognition:** In addition to sentiment analysis, the Humantic API can recognize specific emotions conveyed in the text. It can identify emotions such as happiness, sadness, anger, fear, and others, providing insights into the emotional context of the text.
- **Customizable Parameters:** The API offers customizable parameters that allow users to fine-tune the analysis according to their specific requirements. Users can adjust parameters such as language settings, sentiment thresholds, and emotion categories to optimize results for different use cases and applications.

3.2 Project Design and Architecture

A major development in resume analysis and optimisation is represented by the Resume Analysis Application using Latent Language Models (LLM). The software provides job searchers wishing to improve their resumes and increase their chances of success in the cutthroat job market with a complete solution that combines advanced natural language processing techniques with an intuitive UI.

One of the main advantages of the programme is its intuitive and smooth user interface, which has been meticulously created. Users are led through the resume analysis process by an aesthetically pleasing and tidy interface as soon as they launch the programme. Resumé uploading in PDF, DOC, and DOCX formats is made easy for users by the interface's straightforward drag-and-drop file upload capability. All along the way, users are helped and made to comprehend every stage of the analysis process by the clear explanations and tooltips.

Following an upload of a resume, the programme starts the preparation phase, which involves normalising and cleaning the text to guarantee correctness and consistency. Ensuring uniform resume content and analysis readiness depends on this preprocessing stage. To make more analysis easier, formatting errors and irrelevant material are eliminated, and the text is standardised in design.

Modern natural language processing models trained on enormous volumes of text data, known as BERT or GPT, provide the basis of the analysis engine of the programme. These models let the programme to decipher the semantic background of resume text and get useful information from it. Employing sophisticated methods such entity identification, semantic similarity, and context-aware language understanding, the software can extract important data from the resume including work experience, education, abilities, certifications, and accomplishments.

The application's analysis goes beyond simple keyword matching to give customers thorough understanding of the advantages, disadvantages, and areas for development of their resume. Clients get customised suggestions based on their particular skill set, professional objectives, and market need. Among these suggestions can be ways to highlight important qualifications and experiences, streamline resume material, and make layout changes. To help users fill up skill gaps and progress in their professions, the software also compares extracted abilities and experiences with job criteria

and industry trends and offers suggestions for pertinent courses, certifications, and training programmes.

The analytical algorithms and recommendations are always being updated as people use the programme and offer input. Through this iterative approach, users may optimise their resumes and succeed in their job hunt by knowing that the application is always current and responsive to changing user needs and market trends. All things considered, the Resume Analysis Application with Latent Language Models (LLM) is a major development in resume analysis technology that gives job searchers practical information and customised suggestions for professional growth.

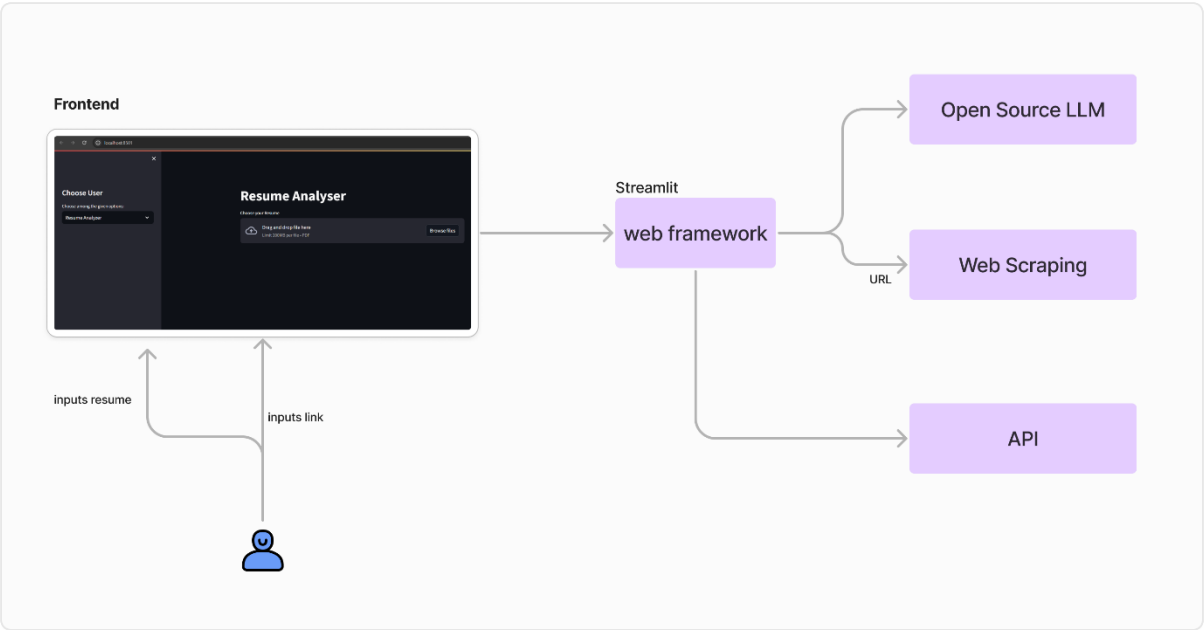


Fig.1: Flow of the app

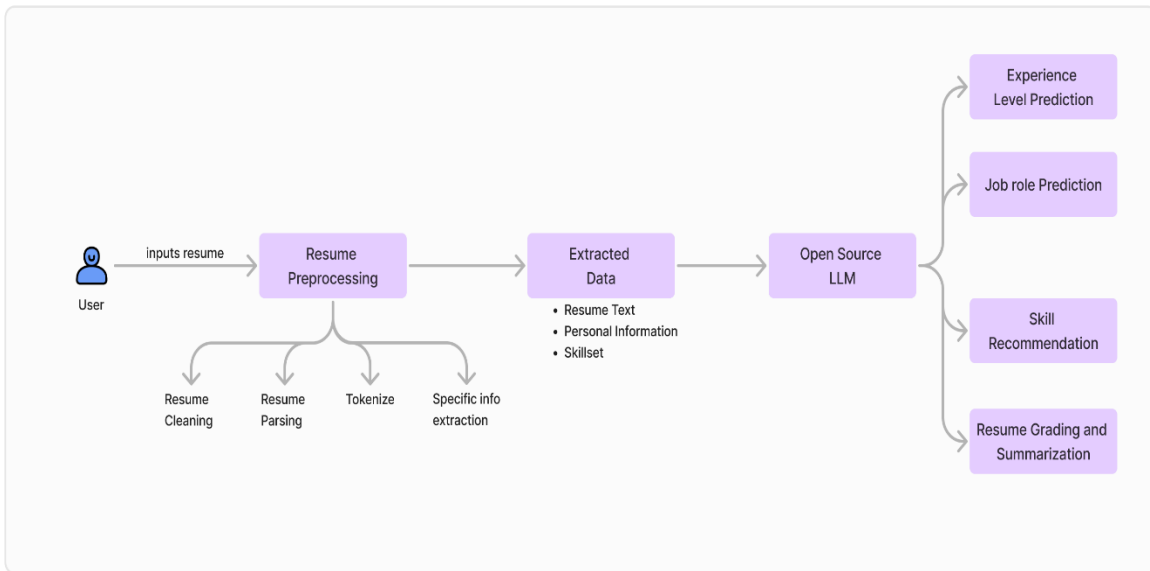


Fig.2 Flow of Resume Analyzer

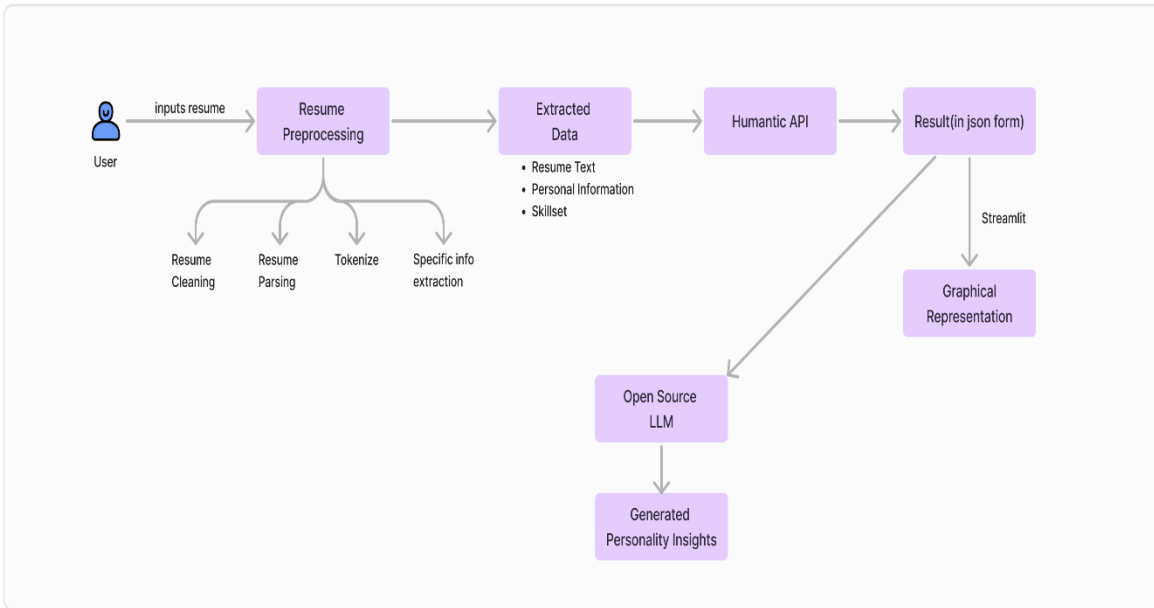


Fig.3 Flow of Personality Insights

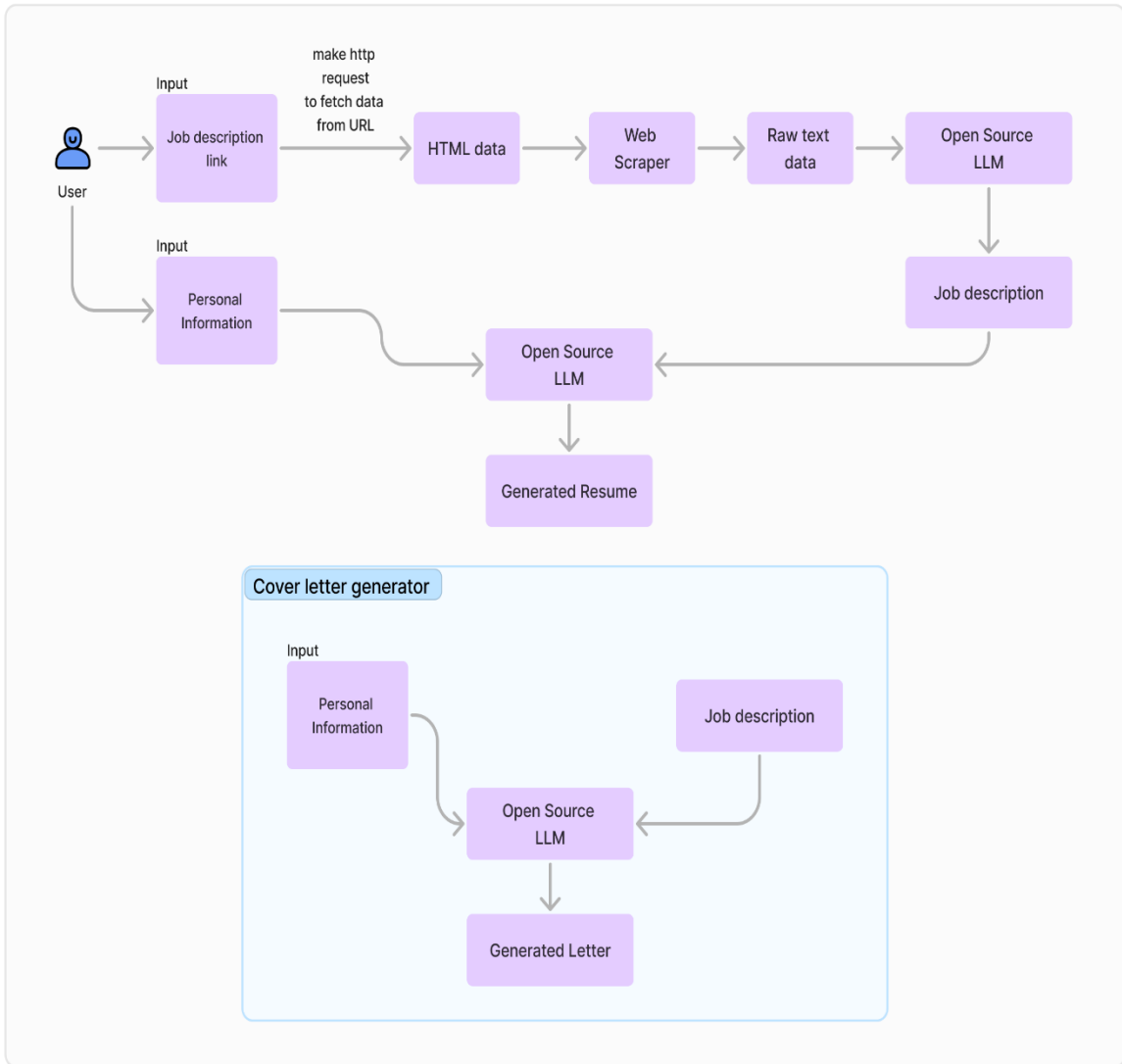


Fig.4 Flow of Resume Builder

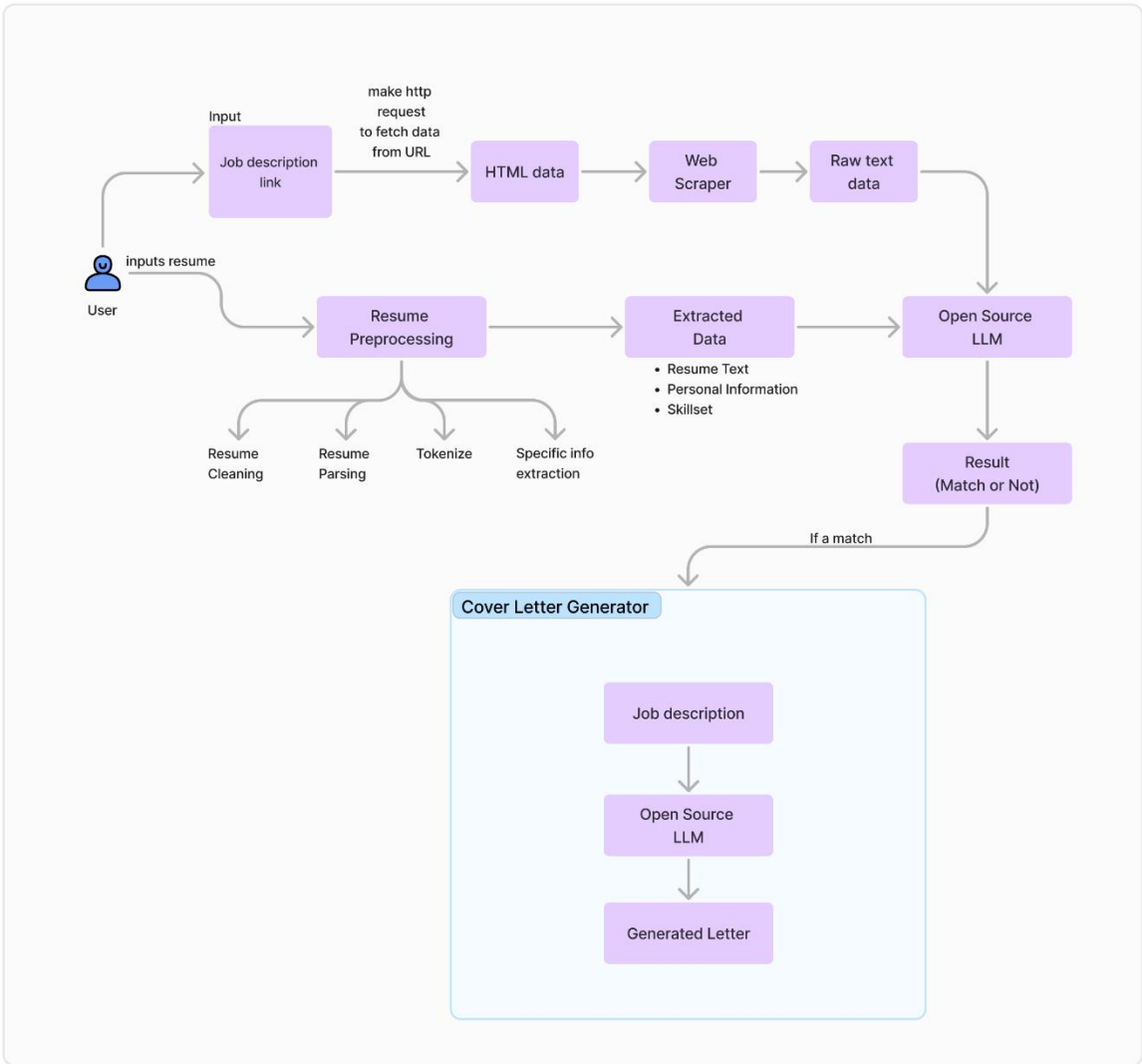


Fig.5 Flow of Resume Matching

3.3 Data Preparation

Automated systems are now essential for quickly processing and evaluating candidate data in the modern job market, when hiring managers and recruiters are deluged with resumes. The resume cleansing and resume parsing are two essential parts of these systems. We'll examine the nuances of both processes in this thorough overview, looking at their goals, important duties, underlying methods, and overall integration inside automated systems.

Resume Cleaning:

The first stage of the automated resume analysis pipeline is resume cleaning, which deals with the variety of formats, structures, and quality variances that are present in resumes. Preprocessing the retrieved data, standardizing the information, and removing inconsistencies are the main goals in order to produce a uniform and clean dataset for further analysis. This is crucial to guaranteeing the automated resume analysis system's efficacy and correctness.

Important Assignments:

1. **Standardization of Formatting:** Resume formats, such as font styles, sizes, and document layouts, are frequently varied. Standardizing these components enables the cleaning procedure to produce a visually uniform and consistent dataset. This entails establishing a uniform structure for all resumes and standardizing date formats and font styles.

2. **Noise Removal:** - Resumes may include extraneous information, graphics, headers, or footers that aren't necessary. Finding and eliminating this kind of noise allows the dataset to be reduced to its most basic parts. This is a crucial step in minimizing irrelevant data that can impede precise analysis.

3. **Typographical Error Correction:** - Inaccuracies in names, addresses, and other pertinent information might make analysis difficult. During the cleaning process, algorithms are used to find and fix typographical problems, guaranteeing that the collected data is error-free.

The process of eliminating duplicate material from resumes involves identifying and removing instances where specific data or phrases are used more than once. To make the dataset more streamlined, the cleaning process entails locating and eliminating these redundancies. This improves productivity and guards against errors resulting from redundancy in later analysis.

5. **Consistent Recognition of Entities:** It is imperative that elements such as talents, education, and work experience be consistently recognized. By improving the extraction algorithms through

cleaning, important information may be accurately recognized and extracted while accounting for linguistic and terminological variances.

Approaches: - Regular expressions for pattern matching, string matching algorithms for phrase recognition, and context-aware noise reduction using natural language processing (NLP) approaches are some of the techniques used during resume cleaning. Moreover, machine learning models can be applied to more difficult jobs like standardizing formatting.

The results of the resume cleaning procedure serve as the basis for the automated system's later phases. The information input into the resume processing step is consistent and free of extraneous noise when the data is cleaned and standardized, which creates the conditions for the precise extraction of crucial information.

Resume Parsing:

The next step in automated resume analysis is called resume parsing, and it involves methodically organizing and structuring the cleaned and standardized data for simple analysis and retrieval. The main goal is to gather precise information in an organized manner regarding a candidate's experiences, abilities, and credentials. During the hiring process, insights are generated, and judgments are made based on this structured data.

1. **Entity Extraction:** - Extracting important entities from the cleansed data is one of the main jobs in resume parsing. This entails locating and retrieving personal data (name, contact information), as well as data on educational background, professional experience, certificates, and other pertinent information. For this objective, Named Entity Recognition (NER) models are frequently used.

2. **Contextual Understanding:** - To comprehend the contextual relationships among various entities in the resume, algorithms are used. For example, contextual comprehension is necessary to discern between a common term and a specific ability. This guarantees that the system understands the text's complex meaning correctly.

3. **The Hierarchical Structure:** A hierarchical structure reflecting the innate relationships between various things is created from the retrieved data. This organization makes it easier to get particular details for later analysis. For example, the hierarchical structure can group abilities, employment history, and educational background into different branches for convenient access.

4. **Normalization:** - Transforming extracted data into a standard format is the process of normalization. This includes maintaining a unified representation of data overall, standardizing

job titles, and making sure date formats are consistent. Normalization is essential to removing discrepancies that could occur from variations in the information that candidates choose to include on their resumes.

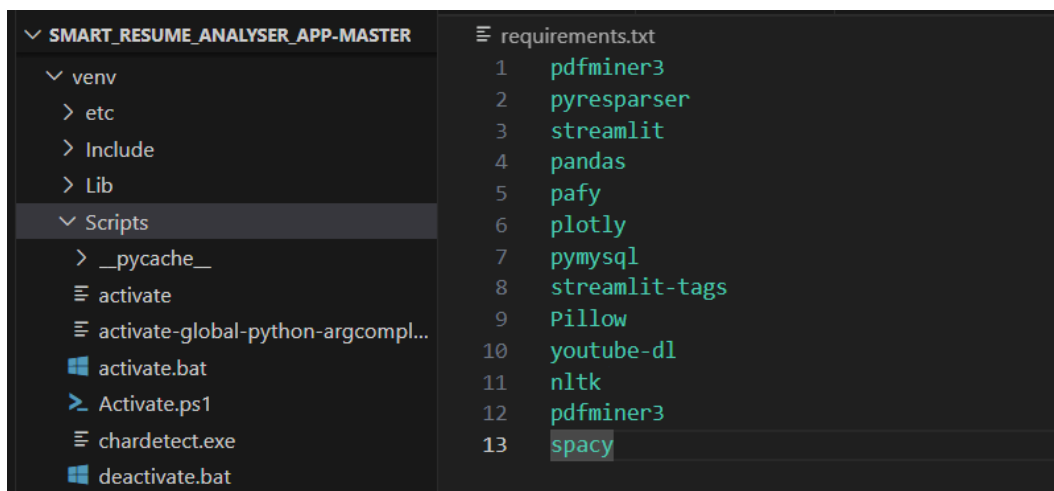
5. Database Integration: - For archiving and retrieval, the parsed data is incorporated into a database. This database facilitates effective searching and analysis by acting as a central store for candidate data. The system's scalability and accessibility are improved by database integration.

Techniques: - During resume parsing, entity extraction is frequently accomplished using Named Entity Recognition (NER) models, rule-based algorithms, and machine learning techniques. Contextual comprehension is enhanced by methods like semantic analysis and pattern matching. Database management systems are used to store and arrange parsed data efficiently.

A structured and ordered dataset is produced as a result of the resume parsing process, which is very helpful for further analysis. Because the structured data is kept in a database, it can be quickly retrieved for searching, creating insights, and helping hiring managers make well-informed decisions.

3.4 Implementation

Setup the environment: Create a python virtual environment and install the required libraries and dependencies.



```
SMART_RESUME_ANALYSER_APP-MASTER
├── venv
│   ├── etc
│   ├── Include
│   ├── Lib
│   └── Scripts
│       ├── __pycache__
│       ├── activate
│       ├── activate-global-python-argcompl...
│       ├── activate.bat
│       ├── Activate.ps1
│       ├── chardetect.exe
│       └── deactivate.bat
└── requirements.txt
    1 pdfminer3
    2 pyresparses
    3 streamlit
    4 pandas
    5 pafy
    6 plotly
    7 pymysql
    8 streamlit-tags
    9 Pillow
    10 youtube-dl
    11 nltk
    12 pdfminer3
    13 spacy
```

Fig. 6: Environment setup

Data Preparation: Load and preprocess the resume data.

```
def pdf_reader(file):
    resource_manager = PDFResourceManager()
    fake_file_handle = io.StringIO()
    converter = TextConverter(resource_manager, fake_file_handle, laparams=LAParams())
    page_interpreter = PDFPageInterpreter(resource_manager, converter)
    with open(file, 'rb') as fh:
        for page in PDFPage.get_pages(fh,
                                      caching=True,
                                      check_extractable=True):
            page_interpreter.process_page(page)
            print(page)
            text = fake_file_handle.getvalue()

    # close open handles
    converter.close()
    fake_file_handle.close()
    return text

def show_pdf(file_path):
    with open(file_path, "rb") as f:
        base64_pdf = base64.b64encode(f.read()).decode('utf-8')
    # pdf_display = f'<embed src="data:application/pdf;base64,{base64_pdf}" width="700" height="1000" type="application/pdf">'
    pdf_display = f'<iframe src="data:application/pdf;base64,{base64_pdf}" width="700" height="1000" type="application/pdf"></iframe>'
    st.markdown(pdf_display, unsafe_allow_html=True)
```

Fig. 7: Data preparation

Instruction Engineering: Prompt for the model to execute the task

```
def get_response(resume_text):
    response = palm.generate_text(prompt=f"Grade this resume: {resume_text} on a scale of 1-10. Grade it on the basis of education")
    # print(response.result)
    return response.result

def get_role(resume_text):
    response = palm.generate_text(prompt=f"Give answer in one word, according to the resume: {resume_text}, candidate has experience in")
    # print(response.result)
    return response.result

def get_level(resume_text):
    response = palm.generate_text(prompt=f"Give answer in one word, according to the resume: {resume_text}, what is the level of education")
    # print(response.result)
    return response.result

def get_skills(resume_text):
    response = palm.generate_text(prompt=f"Give answer in form of string separated by commas, the candidate has following skills: {resume_text}")
    # print(response.result)
    return response.result
```

Fig. 8: Prompts for LLM to execute

Resume Builder: Helps in generating resume for a special job posting

```
response = requests.get(job_link, headers={'User-Agent': random.choice(user_agents)})
print(response.status_code)
content = response.text

soup = BeautifulSoup(content, "html.parser")
job_description_raw_text = soup.get_text().replace('\n', '').strip()
print(job_description_raw_text)

# client = Client()
You, 1 second ago • Uncommitted changes
job_description = palm.generate_text(prompt="Get the job decription from the following text: {job_description_raw_text}")
print(job_description.result)
resume_output = palm.generate_text(prompt="Get the job decription from the following text: {job_description_raw_text}")
# print(resume_output.result)
st.success(resume_output.result)
result = st.button("Cover letter Generator")
if result:
    cv_output = palm.generate_text(prompt="Write a cover letter for the specific job based on the following job desc")
    st.success(cv_output.result)
```

Fig. 9: Resume Builder execution

Resume Matcher: Matching the resume with the job posting to find the perfect find according to the resume

```
response = requests.get(job_link, headers={'User-Agent': random.choice(user_agents)})
print(response.status_code)
content = response.text

soup = BeautifulSoup(content, "html.parser")
job_description_raw_text = soup.get_text().replace('\n', '').strip()
print(job_description_raw_text)

prompt = """"You are a job role description extractor. You will take the text and extract the job role description
model = palm.GenerativeModel('gemini-pro')
response = model.generate_content(prompt + job_description_raw_text)
print(response.text)

# job_description = palm.generate_text(prompt="Return the job decription from the given text: {job_description_r

resume_output = palm.generate_text(prompt="Check if the candidate with the resume : {resume_text} is a good match")
# print(resume_output.result)
st.success(resume_output.result)
if "Yes" in resume_output.result:
    result = st.button("Cover letter generator")
    if result:
        cv_output = palm.generate_text(prompt="Write a Cover letter for the specific job based on the following")
        st.success(cv_output.result)
```

Fig. 10: Resume Matcher implementation

```
if "Yes" in resume_output.result:
    result = st.button("Cover letter generator")
    if result:
        cv_output = palm.generate_text(prompt="Write a Cover letter for the specific job based on the following")
        st.success(cv_output.result)
```

Fig 11: Cover letter generation

```
else:
    pdf_file = st.file_uploader("Choose your Resume", type=["pdf"])
    if pdf_file is not None:
        resume_data, save_image_path = file_uploader(pdf_file)
        if resume_data:
            ## Get the whole resume data
            resume_text = pdf_reader(save_image_path)
            BASE_URL = "https://api.humantic.ai/v1/user-profile/create" # Base URL for create endpoint
            headers = {
                'Content-Type': 'application/json'
            }
            API_KEY = os.getenv("HUMANTIC_API_KEY")
            USER_ID = "201132@gmail.com" # or, any unique identifier

            url = f"{BASE_URL}?apikey={API_KEY}&userid={USER_ID}&analysisstype=talent"

            data = resume_text
            payload = json.dumps(data)

            response = requests.request("POST", url, data=payload, headers=headers)
            print(response.status_code, response.text)
            time.sleep(7)
            FETCH_URL = "https://api.humantic.ai/v1/user-profile" # Base URL for the FETCH endpoint
            url = f"{FETCH_URL}?apikey={API_KEY}&id={USER_ID}"
```

Fig. 12: Personality Insights generator

Integration: Combine the NLP model, job prediction, and certification recommendation modules into a cohesive system by creating a streamlit application.

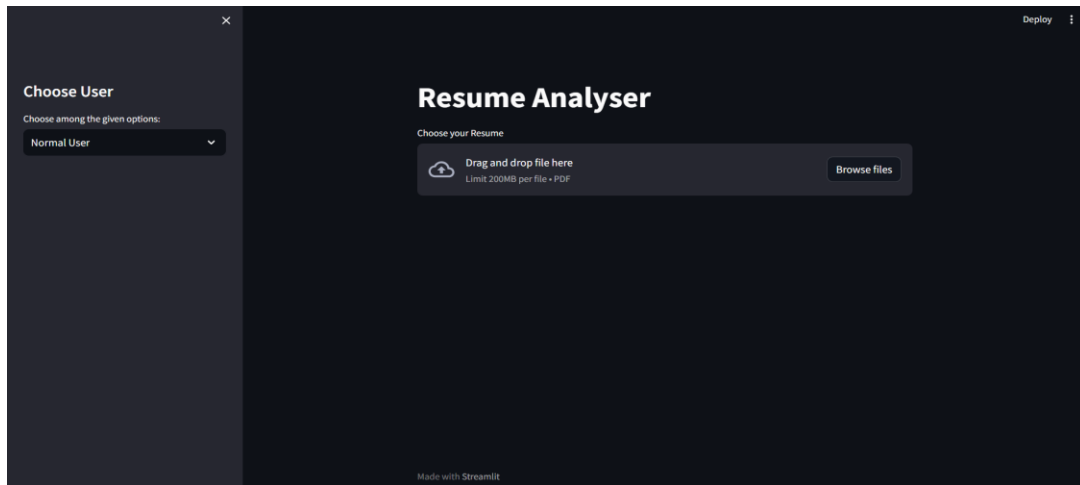


Fig 13: User Interface

3.5 Key Challenges

Creative Resumes and Unstructured Data: For automated systems, managing creative resume forms and unstructured data presents a major difficulty. Certain candidates use eye-catching but non-traditional layouts, like infographics, graphics, or nonlinear structures. It takes sophisticated natural language processing (NLP) methods and machine learning models that can accurately adjust to a variety of resume formats in order to extract pertinent information from such resumes. Finding the right balance in data extraction between precision and flexibility is still a difficult task.

Ambiguities and Nuances in Language: Resume analysis systems are constantly faced with the subtleties and ambiguities found in natural language. Frequently, candidates utilize terminologies that are context-dependent, industry-specific, or abbreviated in ways that differ throughout industries. Comprehending the minute variations in linguistic subtleties is essential to precise entity identification and contextual comprehension. To improve the system's language understanding capabilities, this task calls for constant training on a variety of datasets, semantic analysis, and NLP algorithm improvement.

Algorithmic Decision-Making: Fairness and Bias: Automated resume analysis tools have the potential to unintentionally reinforce prior data biases, producing unfair or discriminating results. Biases can appear at several phases, from the extraction of data to the course recommendation based on talents detected. The introduction of ethical principles, continuous monitoring, and thorough testing are necessary to guarantee fairness in algorithmic decision-making. To reduce bias and advance fairness in the hiring process, tactics including debiasing algorithms, broadening training datasets, and frequent audits are crucial.

Changing Industry Demands and Skill Sets: Resume analysis systems have challenges due to the dynamic nature of skill requirements and the quick evolution of industries. The changing nature of work roles and the emergence of new technology may make traditional keyword-based techniques obsolete. Systems must be flexible, constantly upgrading their knowledge of vocabulary and expertise unique to a certain business. It becomes essential to integrate with real-time job market data and industry professional feedback methods to stay up to date with evolving trends and guarantee that the system's recommendations are still applicable.

Data security and privacy concerns: Strong data security measures are necessary since resumes contain sensitive personal information, which causes privacy concerns. Resumes contain personal information that should be protected from unwanted access, including as contact details and

occasionally even sensitive identifiers. Addressing privacy concerns and building user trust in automated resume analysis systems require implementing encryption techniques, adhering to data protection legislation, and adopting safe data storage practices.

Resume Formats Are Not Standardized: The diversity of resume forms and structures makes standardization difficult. Different titles, templates, and information structures might make it difficult to extract and parse data accurately. It is difficult to establish a standard resume format that is approved by all employers because resumes are customized. Accuracy and consistency must be preserved in automated systems when they are adjusted to various structures. The creation of sophisticated parsing algorithms that can see and classifying data even in the lack of a standard format is necessary to find solutions.

Integrating with People Who Make Decisions: Achieving synergy between technology and human expertise is a difficulty when it comes to effective integration between automated systems and human decision-makers. Improving the automated analytical process's ability to make decisions requires an understanding of and integration with human insights, preferences, and subjective assessments. Creating interfaces that promote teamwork and utilizing user input to continuously enhance the system are key components in striking a balance between automated efficiency and human intuition.

CHAPTER 4

TESTING

4.1 Testing Strategy

Unit Tests: The NLP Model Unit Test, which verifies the precision and efficacy of the Natural Language Processing (NLP) model, is a crucial component of the Resume Analyzer project's overall testing approach. This test seeks to ascertain whether the NLP model accurately extracts pertinent data from a range of resumes. In this context, it entails presenting the model with many templates that depict diverse conventions, styles or forms adopted by industries. In the unit test, the model determines if it is able to process input items in text format correctly, detect key elements as well as generate useful information about work experience, academic qualification, and other related details. Throughput assessment on such a microscopic level makes sure that the interpretation of the NLP model matches the complex subtleties of actual practice. The successful execution of the NLP Model Unit Test demonstrates the validity of the model and acts as the basis for the competence of the Resume Analyzer to analyze different resumes correctly.

Integration Tests: To test the reliability of the Resume Analyzer system, the end-to-end testing should be carried out to ensure that interfaces match perfectly. Integration is necessary for the NLP model, certification recommendation, and job prediction modules to function as a cohesive unit. This entails ensuring that data flows seamlessly between modules and that consistency and correctness are always upheld. Next, end-to-end testing is used to evaluate the system's functionality in detail. This testing begins with the resume input and concludes with the outputs of job prediction and certification recommendation. This test simulates real-world environments to confirm that the related parts work well together and produce correct results. A well-functioning system that assesses resumes, predicts job openings, and integrated recommends relevant certificates is one that integrates and tests well from start to finish. Regular validation and refinement of these procedures improves the Resume Analyzer project's overall dependability and efficacy.

Hardcoded Data Validation: Based on the job prediction and the predefined hardcoded data, this crucial Resume Analyzer project component verifies that the certifications that are suggested are valid and pertinent. This validation process includes a careful comparison of the anticipated employment roles with the relevant credentials in the hardcoded data. The system does this in an

attempt to confirm that the recommended certifications satisfy the standards and industry norms pertaining to the expected work positions. In addition to increasing the accuracy of the certification recommendations, this thorough verification procedure guards against mistakes that could arise from discrepancies between the hardcoded data and the dynamic nature of labor markets. The automated resume analysis process is made more reliable and effective overall by the Resume Analyzer system's ongoing validation and updating of its capacity to offer pertinent and helpful certification recommendations.

User-Interface Tests: The success of the project depends on efficient input validation, which ensures that the Resume Analyzer system handles a range of resume formats and input styles with grace. This means setting up dependable procedures to interpret and preprocess different resume formats while considering variations in document types, layouts, and content configurations. Its capacity to manage this variability makes the system more flexible and easier to use, allowing it to easily accept the diverse range of resumes that are submitted for review. But output validation is equally crucial because it verifies that the system displays certification and job projections accurately. This validation process contributes to the development of trust in the system's analytical capabilities by guaranteeing that end users receive reliable and accurate information. With its emphasis on thorough input and output validation, the Resume Analyzer distinguishes itself as a reliable tool that assists users in customizing their resumes for specific job vacancies. This ensures excellent user satisfaction.

Tests of Robustness: The robustness of the Resume Analyzer is assessed by examining how effectively it manages resumes containing ambiguous or imprecise information and by verifying the strength of its error-handling techniques. The system should be able to analyze resumes with unique forms or vague job descriptions with ease, displaying a flexible approach to language use. In addition, a thorough evaluation of error handling is required to ensure that users receive helpful error messages that aid in issue solving and that the system adjusts seamlessly to unanticipated events. This dual evaluation emphasizes the importance of the system's versatility and user-friendliness. The system's ability to navigate through intricate resume information and accurately highlight any issues improves the Resume Analyzer project's overall dependability and usability.

Cross-platform and cross-browser tests: These are critical to confirming the system compatibility of the Resume Analyzer. To ensure that the system functions consistently and faultlessly across a range of devices and web browsers, extensive testing is required. The project

aims to confirm that users can consistently access and use the Resume Analyzer regardless of their preferred browser (Chrome, Firefox, Safari, or Edge) or operating system (Windows, macOS, or Linux) through extensive compatibility testing. This comprehensive assessment guarantees that the application will be viewed by a larger audience and enhances the user experience, which emphasizes the system's accessibility and adaptability even more. Regardless of the technical environment in which it operates, the Resume Analyzer project places a high priority on cross-browser and cross-platform compatibility to deliver a dependable and consistent user experience.

4.2 Test Cases and Outcomes

In this section test cases are used to assess the system and the anticipated results are covered in this section.

Test Case no. 1

Test Case No.	1
Objective:	To test whether the user is able to upload file from the file explorer.

Test Data	Expected Result	Actual Result
Click Upload Button and upload file	Open file explorer and upload file.	Open file explorer and upload file.

Test Result	The above result indicates that the admin is able click upload button and upload files.
-------------	---

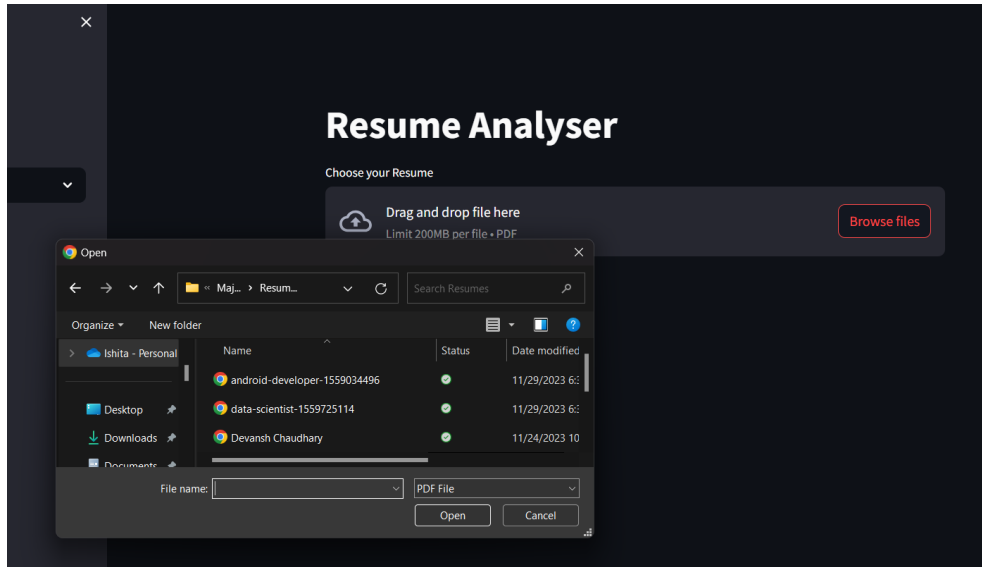


Fig. 14: Result of Test Case no.1

Test Case no. 2

Test Case No.	2
Objective:	To test whether the necessary information from the resume is parsed.

Test Data	Expected Result	Actual Result
Upload the resume	Show name, email, mobile number, technical skills, soft skills.	Show name, email, mobile number, technical skills, soft skills.

Test Result	The above result indicates that the necessary information from the resume has been parsed.
-------------	--

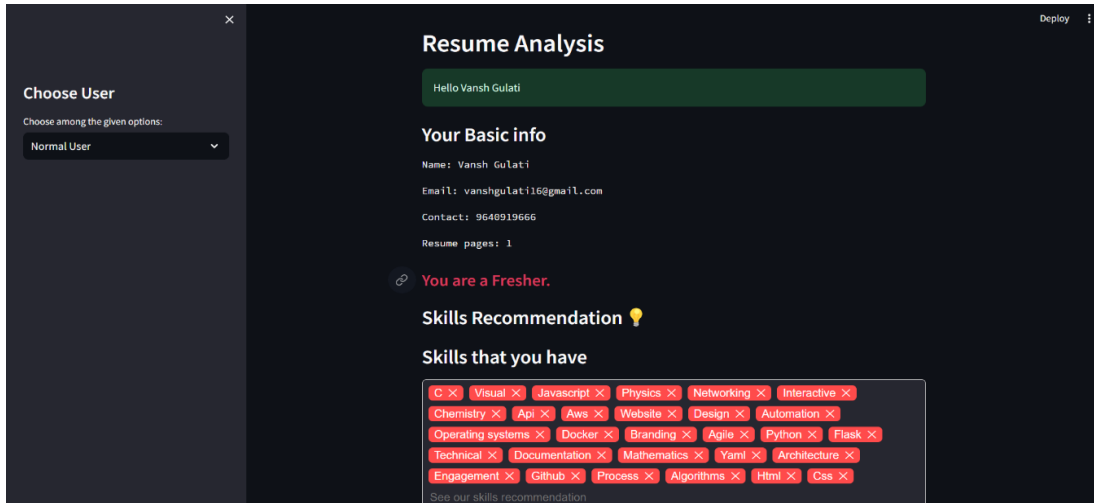


Fig. 15: Result of Test Case no. 2

Test Case no. 3

Test Case No.	3
Objective:	To test whether the job description is being scraped or not.

Test Data	Expected Result	Actual Result
Input the link	Shows job description.	Shows job description.

Test Result	The above result indicates that the job description is scraped.
-------------	---

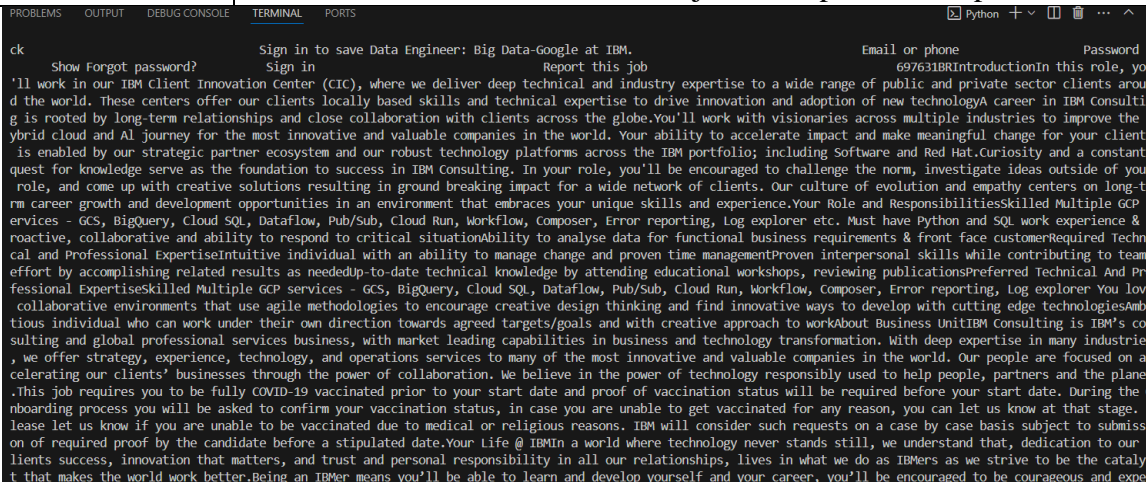


Fig. 16: Result Test Case no. 3

CHAPTER 5

RESULTS AND EVALUATIONS

5.1 Results

Resume Analyzer

The results obtained from the resume analyzer show that the system can extract resume data from the PDF, of any format (alignment, text style, fonts etc.), given as an input. It can successfully extract personal information of the user like name, contact list and the number of pages in the resume.

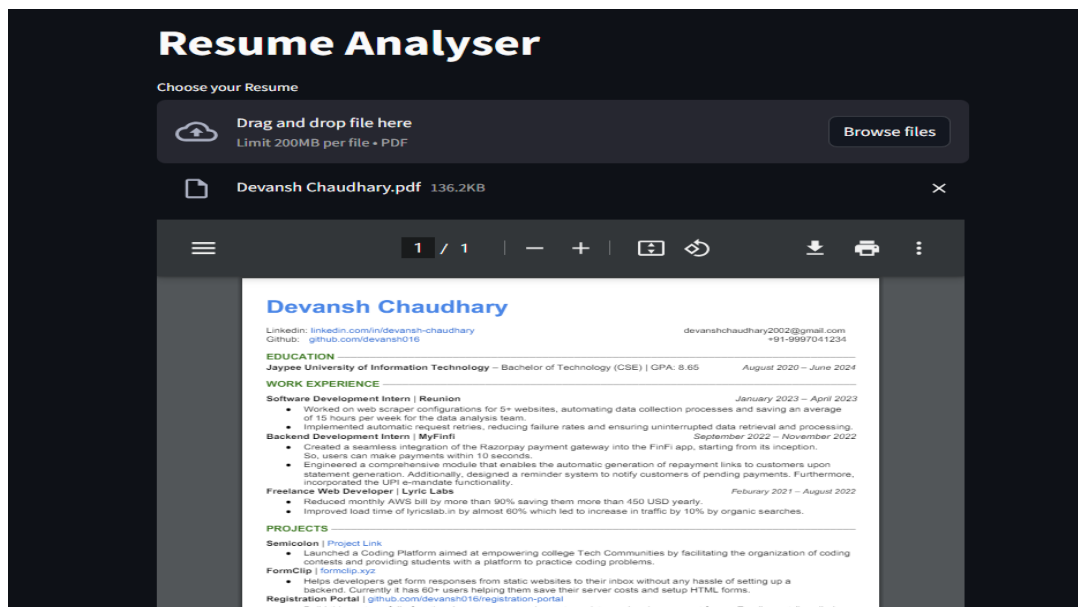


Fig. 17: User uploaded resume

In addition to the above results, it can highlight the skill set of the user, predict their targeted job role, and recommend skills that can be added to their resume to increase their chances to land their targeted job.

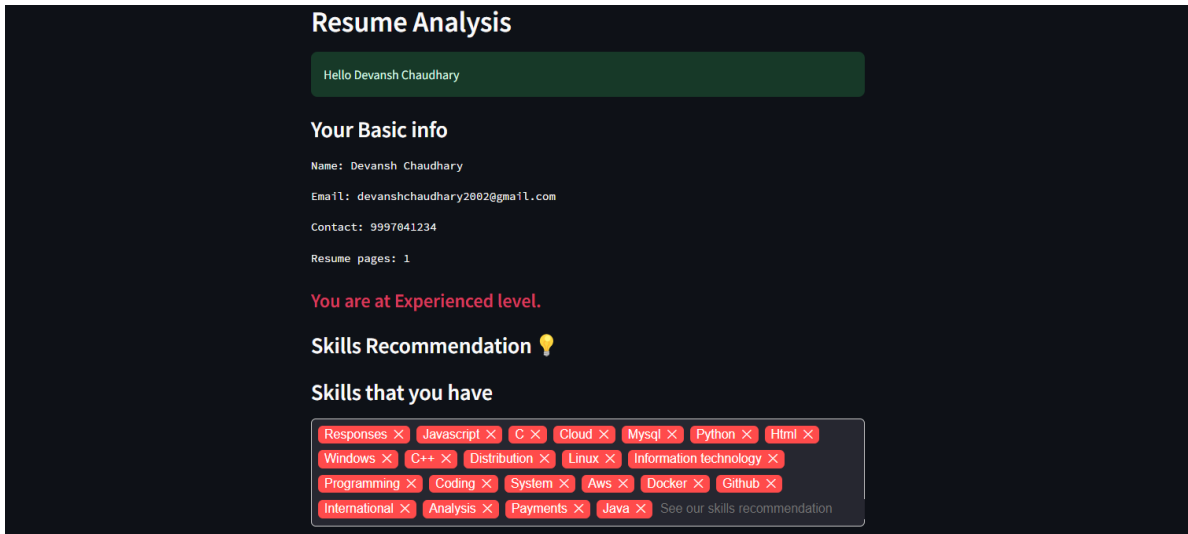


Fig. 18: Details and Skills extraction from resume

The system also has the capability to recommend certifications that outlines the users' current skills from the hardcoded dataset that has been created. The number of certifications suggested can also be increased by using the slider provided in the user interface.

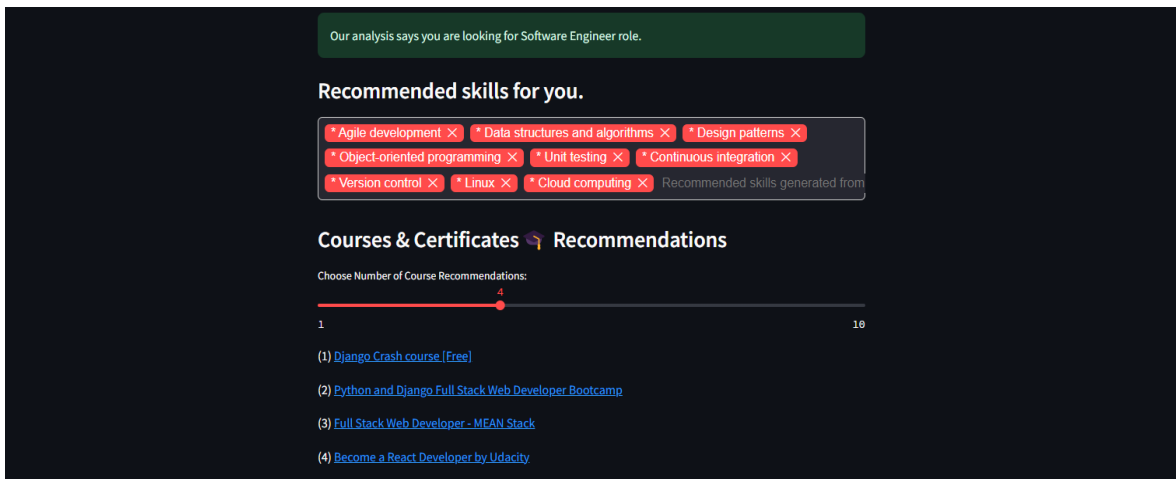


Fig. 19: Skills and course recommendation

It also provides an elaborate grading of all the components in the resume i.e. education, skills etc. and overall grading of the resume. In addition, it provides recommendations to make required changes in the resume to make it better.

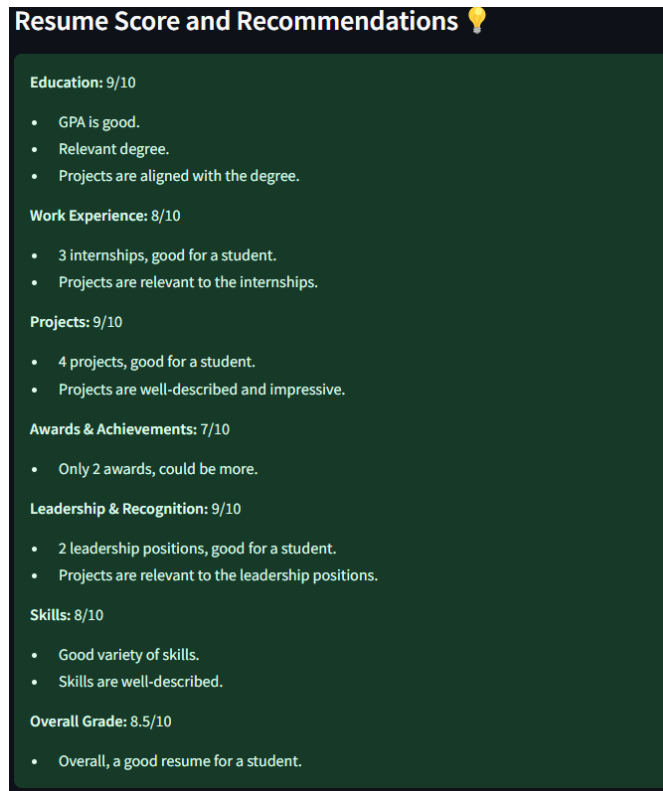


Fig. 20: Resume Grading and insights

Resume Builder

Results from resume builder feature shows that it successfully scrapes data from the URL provided for the job description/role and generates a resume that is suitable to apply for the provided role.

experience

education(with duration)

work_Experience-with duration and work done(if any)

link

<https://www.linkedin.com/jobs/view/3909460771/>

Fig. 21: URL input without providing information

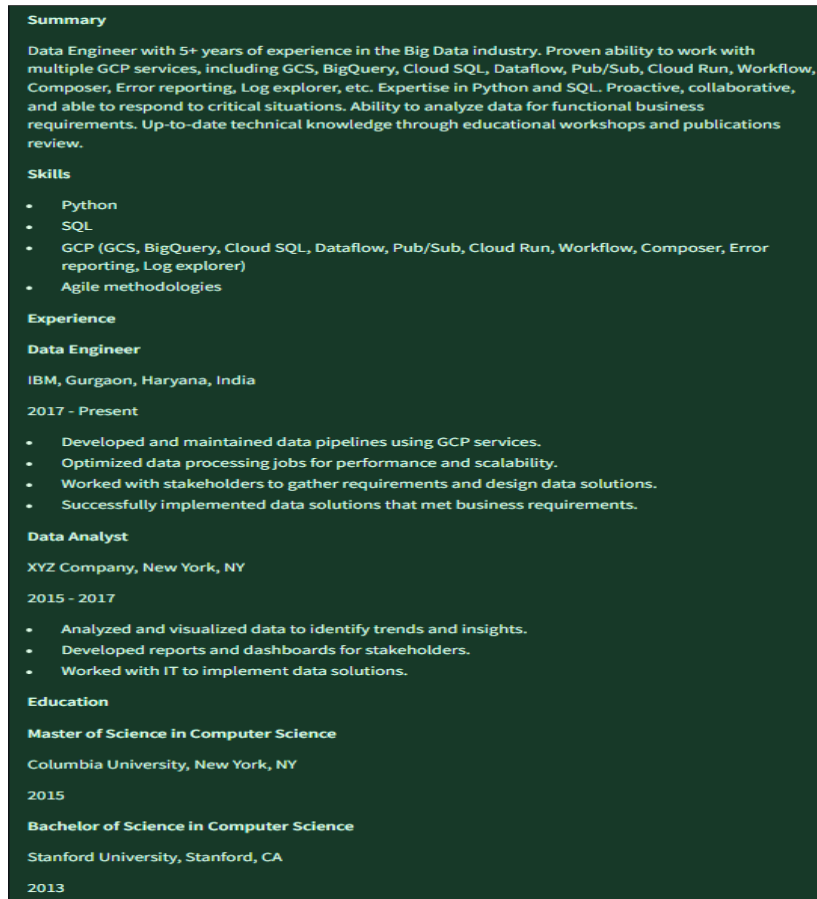


Fig. 22: Generated Resume without any provided information

In addition, it also is capable to generate a resume when the user provides their own information to add into the resume.

experience

Fresher

education(with duration)

Jaypee University of Information Technology(2020-2024) B.Tech

work_Experience-with duration and work done(if any)

None

link

<https://internshala.com/internship/details/work-from-home-php-development-internship-at-smartcoo>

Fig. 23: URL input with information

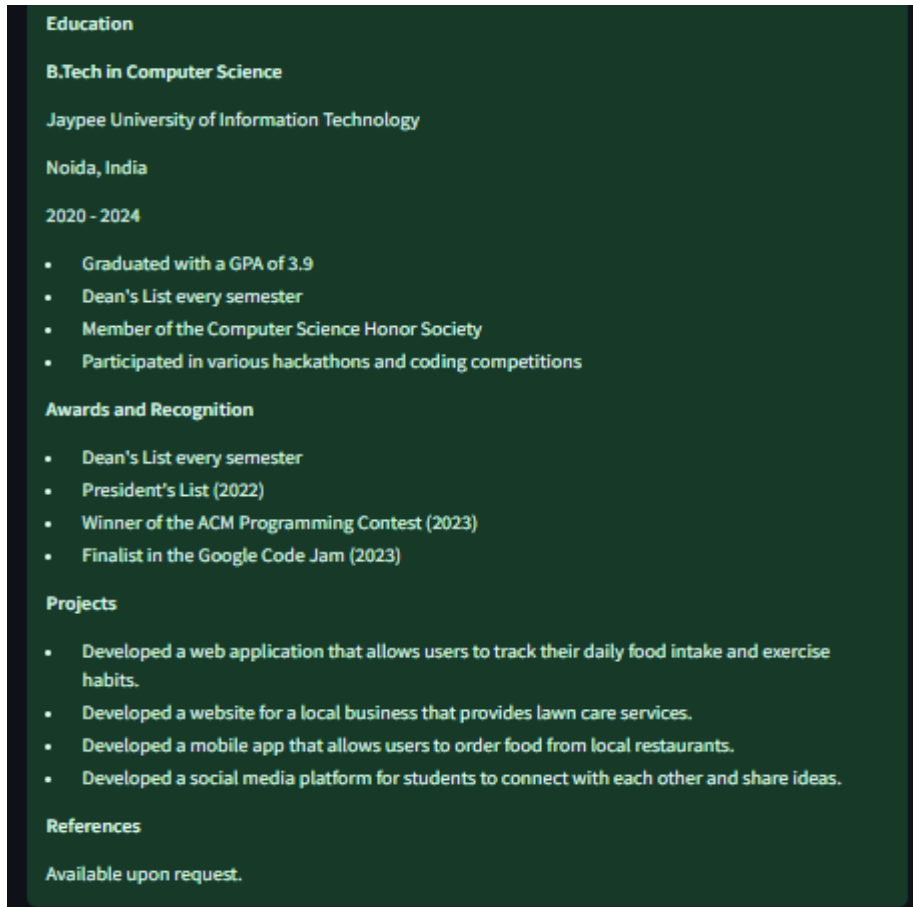


Fig. 24: Generated Resume with provided information

This feature also provides the option to generate a cover letter for the job description provided.



Fig. 25: Generated Cover Letter

Resume Matching

This feature successfully uses the capability of resume data extraction and web scraping to extract the data and then match the resume to the job description.

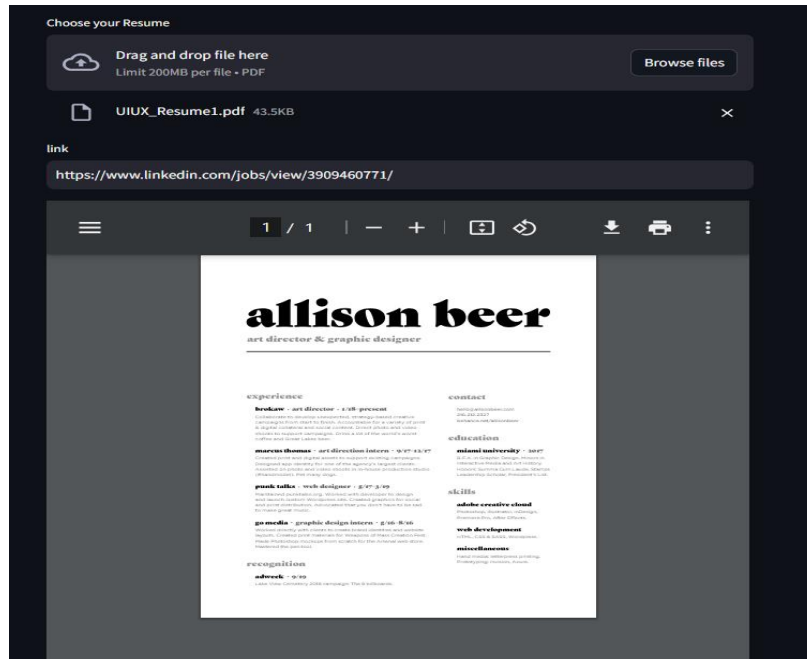


Fig. 26: Resume and Link input

If the resume is not a match, it generates a brief explanation of the reasons.

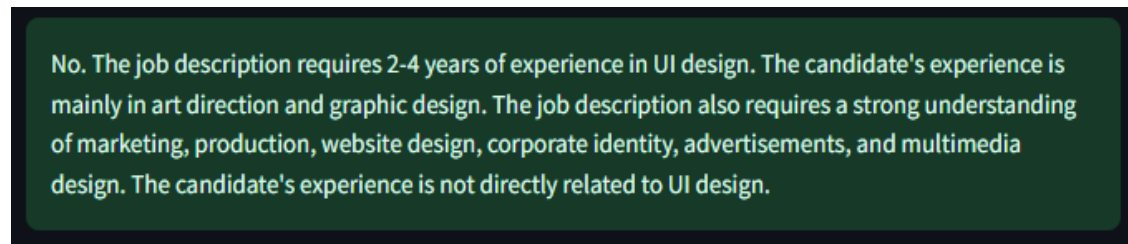


Fig. 27: Result when resume is not a match

If resume is a match, it generates the result and a button to generate a cover letter also appears which then can be used by the user, if needed.

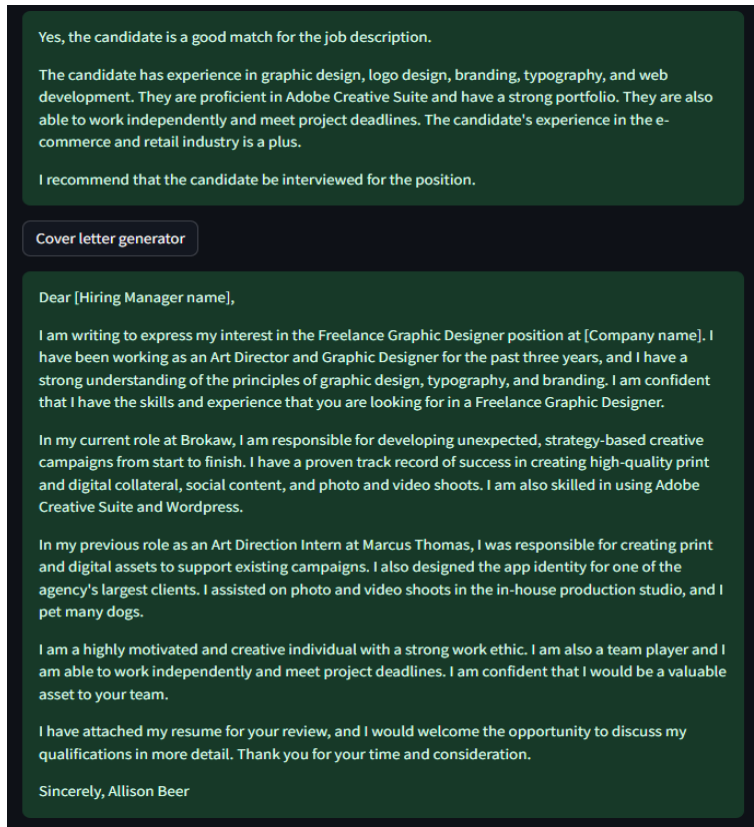


Fig. 28: Result when resume is a match

Personality Insights

The feature successfully uses data from the resume to provide a personality and behavioral factors of the user. The result is in form of bar graphs.

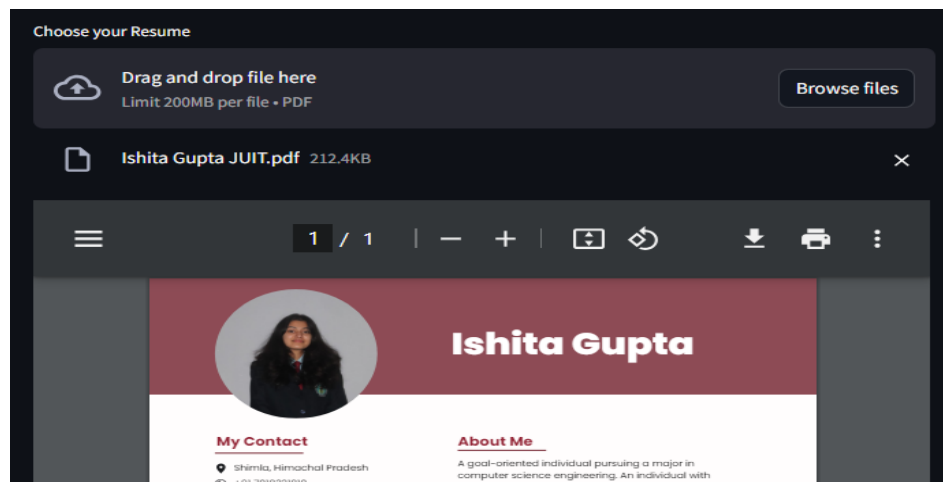


Fig. 29: Resume Input

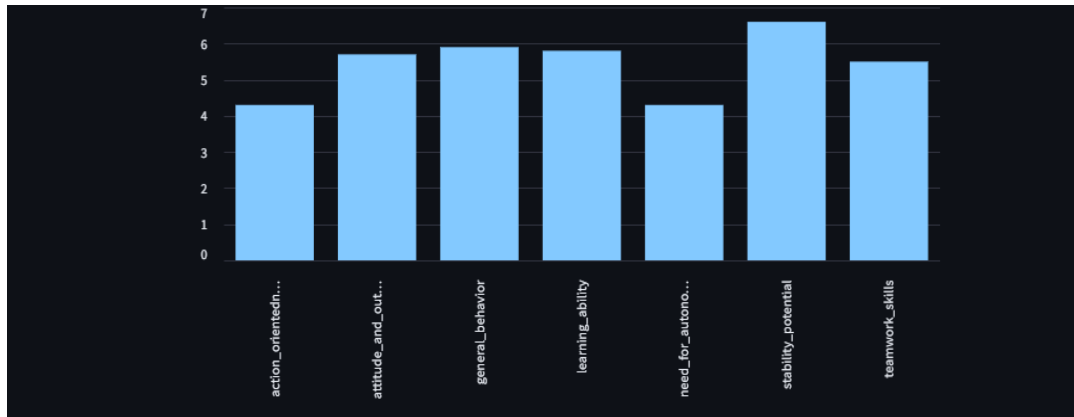


Fig. 30: Graph for Behavioral Factors

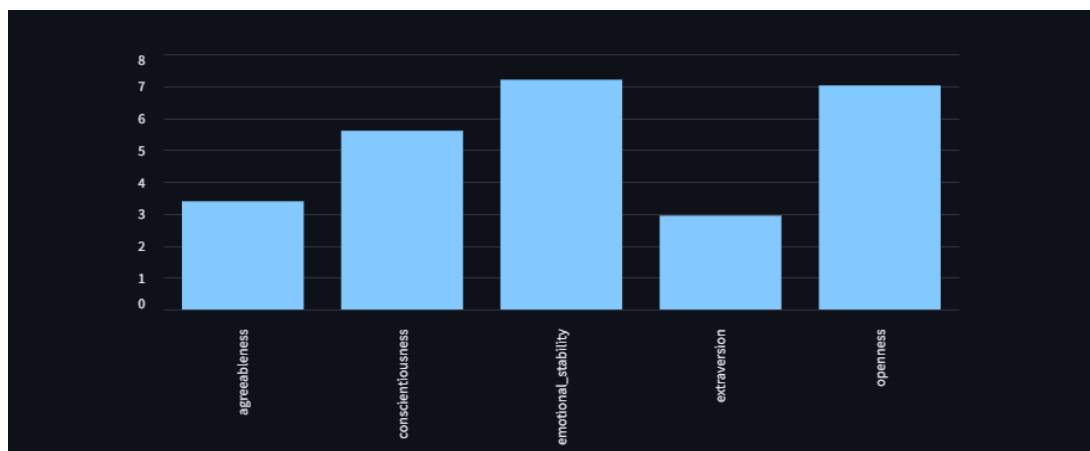


Fig. 31: Graph for OCEAN Personalities

In addition to the above result, it also generates a detailed explanation about the personality insights and how these insights may affect a job role.

The user is a disagreeable, somewhat conscientious, balanced, introverted, and open individual. They have a high stability potential, medium learning ability, sometimes friendly general behavior, a realist attitude and outlook, good teamwork skills, medium need for autonomy, and medium action-orientedness.

This personality profile suggests that the user would be well-suited for a job role that requires independent work and critical thinking. They may also be well-suited for a job role that requires teamwork and cooperation. However, they may not be as well-suited for a job role that requires a lot of social interaction or leadership.

Here is a more detailed explanation of each trait and its potential impact on a job role:

- **Disagreeable:** This trait suggests that the user may be more likely to challenge authority and be less likely to conform to social norms. This could be a positive or negative trait in a job role, depending on the specific requirements of the role.
- **Somewhat Conscientious:** This trait suggests that the user is generally organized and reliable, but they may not be as detail-oriented as some other individuals. This could be a positive or negative trait in a job role, depending on the specific requirements of the role.
- **Balanced:** This trait suggests that the user is emotionally stable and has a positive outlook on life. This is a positive trait in any job role.
- **Introverted:** This trait suggests that the user prefers to work independently and may be less comfortable in social situations. This could be a positive or negative trait in a job role, depending on the specific requirements of the role.
- **Open:** This trait suggests that the user is curious and open to new experiences. This is a positive trait in any job role.

Fig. 32: Generated personality insights

CHAPTER 6

CONCLUSIONS AND FUTURE SCOPES

6.1 Conclusion

In conclusion, the Resume Analyzer using NLP is a smart, innovative, and interactive tool that can analyze resumes automatically. Natural Language Processing Techniques enables the tool to predict the profession the user is interested in. On top of that, the ability to recommend skills and certifications with the help of the hardcoded dataset that can be added to the user's resume helps the individual in ensuring that they are fulfilling the industry requirements. By providing personalized inputs and recommendations this tool not only guides the user to get better opportunities but also becomes a very useful career development partner.

The NLP model, certification recommendation modules, and job prediction algorithms' smooth integration demonstrate the project's comprehensive methodology. The system's capacity to adapt to different resume formats, proactive skill suggestions, and dynamic certification recommendations are what make it useful in a competitive job market. Thus, the Resume Analyzer is not just a sophisticated tool for resume analysis; it is also a catalyst that helps people effectively communicate their professional stories and enhances their prospects in the dynamic job market.

Limitations

While the Resume Analyzer with NLP is a great tool, it does have some points for consideration. However, if the system cannot manage the difference of languages and culture in CVs, the efficiency of such will reduce. Using static data for certification purposes could compromise the agility and slow down industry developments. Challenges arise when using even sophisticated NLP algorithms, such as an uncertainty within skills' extraction or the limitation of checking soft skills. Strong privacy rules within the system are important in dealing with issues relating to data mining. Besides, it shows vulnerability in terms of resumption structure context. It is important to acknowledge these limitations and to strive towards optimizing the system's effectiveness for use in different professional settings.

6.2 Future Scope

The incorporation of intelligent solutions, such as a scoring system, resume data storage in a database, and an administrative dashboard, is how hiring processes are expected to operate in the

future. This proposed system centralizes and organizes information while introducing a data-driven approach to resume evaluation. One essential element is the Administrator Dashboard, which offers a centralized area for in-depth analysis and tactical decision-making. Recruiters can gain useful insights into the makeup and progress of the applicant pool with its features, which include Diversity Metrics, Distribution of Skills and Keywords, Application Progress Overview, and Resume makeup Analysis.

The database-based resume data storage solution guarantees a seamless and well-organized hiring process concurrently. An organized schema, scalability, data security, and interface with existing systems are important factors to consider when designing a database. These factors lay the groundwork for more sophisticated analysis and retrieval. By standardizing assessments, the Resume Evaluation Scoring Mechanism improves the procedure even more. Candidates are assessed objectively and quantitatively using the following criteria: Skills Match, Experience Level, Educational Requirements, and Customizable Criteria. When taken as a whole, these developments offer a revolutionary approach to hiring that promises quicker procedures, well-informed decision-making, and a just and effective hiring system for businesses utilizing technology.

REFERENCES

- [1] Daryani, Chirag & Chhabra, Gurneet & Patel, Harsh & Chhabra, Indrajeet & Patel, Ruchi. (2020). AN AUTOMATED RESUME SCREENING SYSTEM USING NATURAL LANGUAGE PROCESSING AND SIMILARITY. 99-103. 10.26480/etit.02.2020.99.103.
- [2] Jagwani, Vidhita & Meghani, Smit & Pai, Krishna & Dhage, Sudhir. (2023). Resume Evaluation through Latent Dirichlet Allocation and Natural Language Processing for Effective Candidate Selection.
- [3] Sanyal, Satyaki & Hazra, Souvik & Ghosh, Neelanjan & Adhikary, Soumyashree. (2017). Resume Parser with Natural Language Processing. 10.13140/RG.2.2.11709.05607.
- [4] Shubham Bhor, Vivek Gupta, Vishak Nair, Harish Shinde, S Manasi, Kulkarni. Resume Parser Using Natural Language Processing Techniques.
- [5] Nathan Green, Michelle Xiang, Liu, Diane Murphy, "Using an Electronic Resume Analyzer Portal (e-RAP) to Improve College Graduates Employability," Information Systems Education Journal (ISEDJ), vol. 18, no. 3, pp. 1545-679X, June 2020.
- [6] Das, Papiya & Pandey, Manjusha & Rautaray, Siddharth. (2018). A CV Parser Model using Entity Extraction Process and Big Data Tools. International Journal of Information Technology and Computer Science. 10. 21-31. 10.5815/ijitcs.2018.09.03.
- [7] Resume Recommendation using Machine Learning. Author: Ramba S Naik, Shrinivas R Dhotre. Published in International Journal of Creative Research Thoughts (IJCRT), Volume 10, Issue 7 July 2022.
- [8] A Review Paper on Resume scanning using python. Author: Panil Jain, Nandkishor Kamble, Risha Nadar, Shaikh Luqman. Published in International Research Journal of Engineering and Technology (IRJET), Volume: 09 Issue: 04 — Apr 2022.
- [9] NLP based Extraction of Relevant Resume using Machine Learning. Author: Nirali Bhaliya, Jay Gandhi, Dheeraj Kumar Singh. Published in International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075 (Online), Volume-9 Issue-7, May 2020.
- [10] Resume Screening Using LSTM. Author: Navale Sakshi, Doke Samiksha, Mule Divya3, Prof. Said S. K. Published in International Journal of Research Publication and Reviews, April 2022.

- [11] Singh, Moninder, Karthikeyan Natesan Ramamurthy, and Shrihari Vasudevan. "Propensity modeling for employee Re-skilling." 2017 IEEE Global Conference on Signal and Information Processing(GlobalSIP). IEEE, 2017.
- [12] Ayishathahira, C. H., C. Sreejith, and C. Raseek. "Combination of Neural Networks and Conditional Random Fields for Efficient Resume Parsing." 2018 International CET Conference on Control,Communication, and Computing (IC4). IEEE, 2018.
- [13] JIANQIANG, Z., AND XIAOLIN, G. Comparison research on text pre-processing methods on twitter sentiment analysis. IEEE Access 5 (2017), 2870–2879.
- [14] JOSE, M., KURIAN, P. S., AND BIJU, V. Progression analysis of students in a higher education institution using big data open source predictive modeling tool. In Big Data and Smart City (ICBDSC), 2016 3rd MEC International Conference on (2016), IEEE, pp. 1–5.
- [15] MANDAL, B., SETHI, S., AND SAHOO, R. K. Architecture of efficient word processing using hadoop mapreduce for big data applications. In Man and Machine Interfacing (MAMI), 2015 International Conference on (2015), IEEE, pp. 1–6.
- [16] NARASIMHAN, R., AND BHUVANESHWARI, T. Big dataa brief study. Int. J. Sci. Eng. Res 5, 9 (2014), 350–353.
- [17]JULUSOY, O` . Research issues in real-time database systems: survey paper. Information Sciences 87, 1-3 (1995), 123–151.
- [18] VIJAYARANI, S., AND JANANI, M. R. Text mining: open source tokenization tools–an analysis. Advanced Computational Intelli- gence 3, 1 (2016), 37–47.
- [19] Ravindranath, Vinodh Kumar, et al. "Inferring Structure and Meaning of Semi-Structured Documents by using a Gibbs Sampling Based Approach." 2019 International Conference on Document Analysis and Recognition Workshops (ICDARW). Vol. 5. IEEE, 2019.
- [20] GARCIA, T., AND WANG, T. Analysis of big data technologies and method-query large web public rdf datasets on amazon cloud using hadoop and open source parsers. In Semantic Computing (ICSC), 2013 IEEE Seventh International Conference on (2013), IEEE, pp. 244–251
- [21] FERGUSON, M. Architecting a big data platform for analytics. A Whitepaper prepared for IBM 30 (2012).
- [22] Gugnani, Akshay, Vinay Kumar Reddy Kasireddy, and Karthikeyan Ponnalagu. "Generating unified candidate skill graph for career path recommendation." 2018 IEEE International Conference on Data Mining

Workshops (ICDMW). IEEE, 2018.

[23] BAMNOTE, G., AND JOSHI, H. Distributed database: A survey. *International Journal of Computer Science and Applications* 6, 2 (2013), 0974–1011

APPENDIX

Vansh Report

ORIGINALITY REPORT

5 %	5 %	2 %	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.researchgate.net Internet Source	2 %
2	web.archive.org Internet Source	1 %
3	github.com Internet Source	<1 %
4	iieta.org Internet Source	<1 %
5	tudr.thapar.edu:8080 Internet Source	<1 %
6	Xiaoguang Tian, Robert Pavur, Henry Han, Lili Zhang. "A machine learning-based human resources recruitment system for business process management: using LSA, BERT and SVM", Business Process Management Journal, 2022 Publication	<1 %
7	www.irjmets.com Internet Source	<1 %

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT
PLAGIARISM VERIFICATION REPORT

Date: 16/05/24

Type of Document (Tick): PhD Thesis M.Tech Dissertation/ Report B.Tech Project Report Paper

Name: Ishita Gupta, Vansh Gulati Department: CSE Enrolment No 201132, 201109

Contact No. 7018321810 E-mail. 201132@juitsoan.in

Name of the Supervisor: Mr. Saikal Firdous, Mr. Ramesh Narwal

Title of the Thesis/Dissertation/Project Report/Paper (In Capital letters): RESUME
ANALYZER USING NLP

UNDERTAKING

I undertake that I am aware of the plagiarism related norms/ regulations, if I found guilty of any plagiarism and copyright violations in the above thesis/report even after award of degree, the University reserves the rights to withdraw/ revoke my degree/report. Kindly allow me to avail Plagiarism verification report for the document mentioned above.

Complete Thesis/Report Pages Detail:

- Total No. of Pages = 61
- Total No. of Preliminary pages = 10
- Total No. of pages accommodate bibliography/references = 3

Ishita Vansh
 (Signature of Student)

FOR DEPARTMENT USE

We have checked the thesis/report as per norms and found Similarity Index at 5.....(%). Therefore, we are forwarding the complete thesis/report for final plagiarism check. The plagiarism verification report may be handed over to the candidate.

(Signature of Guide/Supervisor)
16-5-24

[Signature]
 Signature of HOD

FOR LRC USE

The above document was scanned for plagiarism check. The outcome of the same is reported below:

Copy Received on	Excluded	Similarity Index (%)	Generated Plagiarism Report Details (Title, Abstract & Chapters)	
Report Generated on	<ul style="list-style-type: none"> • All Preliminary Pages • Bibliography/Images/Quotes • 14 Words String 		Word Counts	
			Character Counts	
		Submission ID	Total Pages Scanned	
			File Size	

Checked by
 Name & Signature

Librarian

Please send your complete thesis/report in (PDF) with Title Page, Abstract and Chapters in (Word File) through the supervisor at plagcheck.juit@gmail.com