

Smart Resume Analyzer

Mangesh Balpande¹, Prem Bhadane², Mohit Pawar³, Bhushan Kapadne⁴ and Harshal Jadhav⁵

¹⁻⁵SVKM's Institute of Technology – Dhule, Maharashtra, India

Email: mangesh.balpande111@gmail.com, prembhadane69@gmail.com

Abstract— Our innovative Resume Analysis and Enhancement Platform, through the use of advanced technology, makes a change in career advancement. It transforms the production and optimization of a resume through the application of advanced methods for machine learning and natural language processing. Using intelligent analysis, it automatically determines and evaluates users' skills, qualifications, and experience levels and maps them against current industry demands. Its sophisticated recommendation engine, which relies on Python frameworks like Streamlit and PyMySQL, offers personalized recommendations for resume improvement ranging from optimizing the structure to writing impactful objective statements. The system analyzes the recent trends in the market and the job requirements and accordingly suggests the most aptly suited jobs and career paths. It further grades each resume into various dimensions such as content quality, professional impact, and industry alignment. The platform's secure infrastructure will ensure protection of information via end-to-end encryption and adherence to data protection regulations. This smart system not only makes creating resumes easy but also makes it a strategic career development tool, helping people get their strengths heard and noticed in the highly competitive job market today. Through continuous analysis and feedback, users get actionable insights toward enriching their professional profiles and hence ensure qualifications are up-to-date with changes in the industry standards.

Keywords— Job Title Suggestions, ML, NLP, Python Tools (Streamlit, PyMySQL), Resume Analysis, Resume Optimization, Skill Identifi.

I. INTRODUCTION

The smart resume analyzer system based on advanced natural language processing identifies the relevant details from resumes that are not structured. The system can correctly identify the candidate's abilities and credentials, enabling the identification of the most appropriate job titles for their profile. Additionally, our system bases its recommendations on this analytical process, enhancing and improving the user's resume in order to make it more attractive to employers. Our resume analyser makes available to the job seekers personalized and effective solutions through the use of ML and natural language processing. We are able to save the extracted data and analyze the gathered data to produce a rating with the help of Python and its modules. Once the rating has been generated, we recommend a few modifications, such as including lines like "my interests are," "my goal is," "objectives," "declaration," and more. Because businesses get for one position, hundreds or thousands of resumes, the outdated resume sorting procedure is time-consuming and ineffective. The goal of NLP is to achieve more possible in order for computers to understand human language, which is challenging undertaking. Computers perform well with structured data, such as databases and spreadsheets, but they struggle to comprehend unstructured data. Computers are not very good at reading speech or text. Since there are many different kinds of data, NLP enables computers to process and interpret them. Job searchers can benefit from the

interface of the suggested method by receiving expert resume evaluations together with suggestions on how to increase their chances of getting hired. Job searchers can increase their chances of landing the job they desire by using the Smart Resume Analyser to ensure that their resumes are tailored to the particular positions they are applying for. Consequently, the interface of the Smart Resume Analyzer, as suggested by this technique, has capacity to transform the profession. application process and achieve effective & efficient for those looking for work.

II. LITERATURE SURVEY

The method of resume analysis software that automatically extracts and analyzes pertinent data from uploaded resumes is represented by Satyaki Sanyal and team [1]. To find keywords, patterns, and trends, the software uses data mining, ML and NLP. Resumes are filtered & ranked in accordance with the job requirements. This method finds the best applicants for a position, streamlines the hiring process, and lessens the workload of recruiters. According to Vaidya and colleagues [2], resume analysts utilize the strategy outlined here to retrieve pertinent data. Stop words are removed, and similar-sounding words are grouped using the Soundex algorithm. After that, keywords are found and kept in a database. An automated resume screening system that uses natural language processing (NLP) to evaluate resumes and score them according to how well they match job requirements is represented by Daryania and team [3]. To get a relevancy score, the system looks for keywords, credentials, and skills and compares them to the job description. Recruiters can save time by ranking resumes appropriately. A method for extracting data from PDF resumes, storing it in a database, and analyzing it for ranking is presented by Nawander and team [4] utilizing NLP and Streamlit modules. The system makes recommendations for enhancements to resume representation in terms of language, formatting, and layout.

Pokhare [5] exemplifies the method of parsing, extracting, and summarizing data from PDF resumes through the use of NLP and ML. Sections like contact details, training, and professional background are identified by the system. NLP and machine learning approaches analyze the language used to describe the candidate and extract keywords, abilities, and qualities. Databases hold data for study and comparison. The method used by Pravesh and colleagues [6] matches resumes with job criteria by using ML and the K-Nearest Neighbors (KNN) algorithm.

A suggested company recommender system for choosing the most qualified applicant is presented by Kelkar and colleagues [7]. The algorithm ranks resumes according to company criteria using machine learning and text mining. Text mining algorithms are used to extract pertinent information from resumes, which is then compared to the job requirements to get a score. To find correlations between candidate qualifications and requirements, a Historical data is used to train the ML model. Recruiters can use a recommendation engine to get a list of suggested applicants & this model scores resumes. A suggested approach for resume screening that makes use of ML and NLP algorithms is presented by Sinha and colleagues [8]. The goal is to educate the machine to analyze and Withdraw pertinent information from written language that is not organized in order to automate the laborious process of CV screening. Using NLP approaches, Shubham Bhor and colleagues [9] have suggested a solution for resume parsing. The goal is to make it easier to identify qualified applicants for available positions. Important details are extracted from resumes that are uploaded to a job platform.

Using some built-in NLP techniques and filtering jobs, Sanjana and colleagues [10] demonstrate resume validation and filtration. Additionally, it validates resumes appropriately. With so many resumes being sent in for available positions, recruiting has become a time-consuming procedure in recent years. Recruiters cannot realistically go through each one by hand. A mechanism for automatically filtering resumes that analyzes & ranks resumes using ML and natural language processing is represented by Gupta and team [11]. The system combines ML algorithms with NLP approaches including named entity identification and part-of-speech tagging. ML-based approaches for resume analysis are represented by Thakur and Goyal [12]. This time-consuming procedure could make use of the Resume Classification System (RCS) for automation by utilizing NLP and ML algorithms. Moreover, with only human involvement, automating this process can significantly accelerate and Showcase the applicants' screening process. This methodology does not suggest any courses that could strengthen an applicant's résumé; instead, it merely aims to lighten the recruiter's workload. According to this research, we can get the necessary results by automating certain tasks utilizing NLP, RCS, and ML approaches.

The authors [12] emphasized the influence of AI on a wide range of domains and the importance of machine learning (ML) in prediction, identifying patterns and minimizing errors throughout several sectors. In addition to exploring the use of ML in phishing attack detection, the author [13] offered text categorization techniques for a variety of applications. With an emphasis on Telugu film industry names, the author [14] explored application of ML and NL, particularly CNN, for handwriting pattern recognition, excellent accuracy (98.3%). Based on

evaluations or comments, the authors of [15] assessed human sentiment analysis feelings and classified them as either favorable, neutral, or negative. In this paper, sentiment analysis architecture and tools for easy-to-use opinion mining are explored. In order to improve the classification of e-commerce products, the method [16] used Advanced Deep Learning with a global threshold, obtaining great accuracy and pushing the boundaries of current technology.

A. Methodology

Our innovative Smart Resume Analyzer system, characterized by innovative design, offers a state-of-the-art approach that transforms the processes of resume development and job compatibility through seven carefully structured phases. The first Employee Onboarding Stage lays the groundwork by providing a comprehensive and secure user registration platform, incorporating sophisticated security measures to safeguard confidential data. This platform collects a vast amount of user data, such as full names, email addresses, and secure authentication credentials, and organizes all this information within a carefully designed system database that guarantees the integrity and availability of the data.

The Resume Data Entry Module should have more user-friendly and organized templates to help in the systematic input of comprehensive professional information. Users are empowered to record their educational credentials, employment history, technical competencies, professional certifications, and project portfolios with ease and efficiency. This module also enhances the document upload functionality for certificates and supplementary materials, thereby guaranteeing a thorough and verifiable professional profile. Maintaining consistency, the framework also allows for flexibility, effectively addressing the needs of various career trajectories through this organized method.

In the job requirement analysis phase, we have established an advanced job database that transcends simple matching capabilities. The system makes use of complex algorithms in keyword extraction and comes up with a multi-dimensional job classification framework. Classification here does not only refer to the skills and experience levels required but includes in-depth assessments of educational qualifications and their corresponding industry sectors. The execution of the machine learning phase reflects the high technical proficiency because modern supervised learning models armed with natural language processing functionalities enhance the analysis of text at a granular level. Our algorithms for classification perform outstandingly well in skill alignment and are evaluated through an accurate framework of pattern recognition. The model has been trained on extensive historical datasets of placement information that have highlighted trends in the success of job placement and the present industry requirements to produce relevant and accurate results.

The Eligibility Assessment System is a very significant development in candidate assessment using a complex multi-criteria algorithm across the entire assessment process. The scoring framework within this system is very complex and accounts for such parameters as: accurate percentage match in skill, comprehensive analysis of the applicability of experience, qualification match, and a comprehensive analysis of additional credentials. All these aspects are integrated into a weighted scoring system that produces fair, exact, and consistent results. It produces two distinguishable categories of output. In the first place, it produces a well-structured list of suitable job positions along with matching percentages and detailed compatibility scores; in the second case, it provides personalized job eligibility recommendations for people along with guidelines to skill shortages, recommended certificates, internship opportunities, and explicitly defined career advancement paths.

The final stage in the system development cycle is the Continuous Learning and Optimization, where we establish an advanced mechanism of a feedback loop. Such a mechanism would ensure that the system reacts to the fluctuations in the job market with continuous retraining of models and input of new data patterns and trends. Rigorous performance metrics monitoring and systematic protocols for improvement ensure that our Smart Resume Analyzer remains effective and relevant in the face of a dynamic job market. In this regard, the overall process works not only as an instrument for aligning with available opportunities but also as a strategic direction for long-term career advancement and achievement.

III. PROPOSED WORK

Collecting participant resumes is the first step in the interface for the suggested approach. Participants are informed that a maximum allowed size for a submission file is 200MB. After gathering resumes, an instant preview of uploaded files is provided to the participants and they can thus see their submissions and modify them if needed. After that, these CV are safely stored in our system. for use in later stages of the suggested method's interface.

We accept the file that the user uploaded. We now proceed to the second stage, which involves extracting the data from the resume that was uploaded. We use the Pyresparser package to assist us extract the data efficiently. The return values of the pyresparser functions are stored in a variable once they are called. When the pyresparser is called, it retrieves the user's name, email address, phone number, abilities, overall experience, and other information from the file that was submitted and brings it back to the location where the function was invoked. In order to perform NLP operations, the pyresparser module internally makes use of the NLTK and spacy modules. Pyresparser's function generates a list of dictionary objects after receiving the uploaded resume as input. Every dictionary key denotes a media that can be used to identify a person; for example, a person's name, email address, and phone number are all distinctive qualities that can be used to identify them. Such information, which can be obtained from a user's curriculum vitae using this module, may contain a person's name, phone number, and similar details. Lastly, a dictionary is defined inside a list structure. The dictionary's keys are a collection of names that serve as distinct identifiers for each individual, such as names and phone numbers; these keys' values correspond to relevant details on the resume of the user.

A. Objectives of the proposed work

select the best applicants for a certain job from any posting. The system must automatically create pertinent résumé data, including education, experience, accomplishments, and abilities. for a quicker hiring process and a less biased selection process that ranks applicants according to a variety of resume elements.

B. Architecture of proposed work

One could think of the architectural diagram as a flow chart. Each element in the architecture diagram has a purpose that aids comprehension. A decision-making or branching point is represented by a diamond or rhombus, whereas the circle denotes the beginning or end. The lines that emerge from the diamond stand for various situations that could result in various sub-processes. The rectangle represents a certain operation. The process flows' direction and order are depicted by the lines.

Next, we use the variable that contains the user's information to display the user's details. Next, we assess the user's skill set and forecast the kind of work that best suits their qualifications. After determining which role best suits the user, we take into account the user's existing skill set and forecast the skills that are lacking from the user's profile. Since the data export was successful, we can proceed to the following phase. The number of pages and the level of expertise required by the user to generate a rating are both measured in this section. Following creation, we show the user this rating before proceeding to the following step. The skills that the user lacks are later identified, and a link to acquire the necessary skill set is provided. We then evaluate the complete resume and provide a score. We advise the customer to include lines such as "include a list of interests," "your professional goal," and more based on that score.

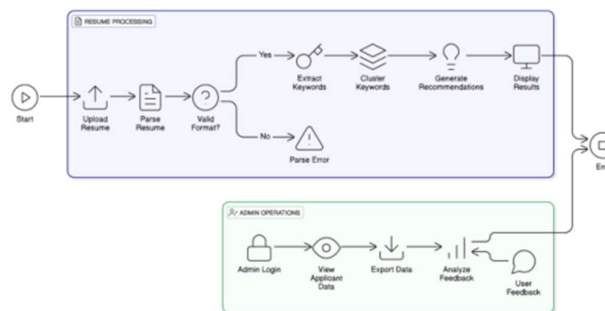


Fig. 1. System Architecture

C. System Workflow

The process begins with users uploading their resumes in supported formats (e.g., PDF, DOCX) through the web interface. The system then utilizes Natural Language Processing (NLP) techniques to extract key information from the resumes, including personal details, skills, education, and work experience. This extracted data is organized into a structured format, such as a table or database, for easy analysis. The system identifies and extracts relevant keywords from the resume content, focusing on skills, qualifications, and experiences. These extracted keywords are grouped into sectors or categories based on their relevance and context, aiding in thematic analysis. The system evaluates the resume against predefined criteria, providing a score or rating that

reflects its quality and alignment with job requirements. Based on the analysis, the system offers personalized recommendations for skill enhancement, course suggestions, and other resources to improve the resume. It generates a detailed feedback report highlighting strengths and areas for improvement in the resume. Users can review the analysis, explore recommendations, and make necessary updates to their resumes. Finally, users have the option to export the analyzed data and feedback in various formats (e.g., CSV, PDF) for further use.

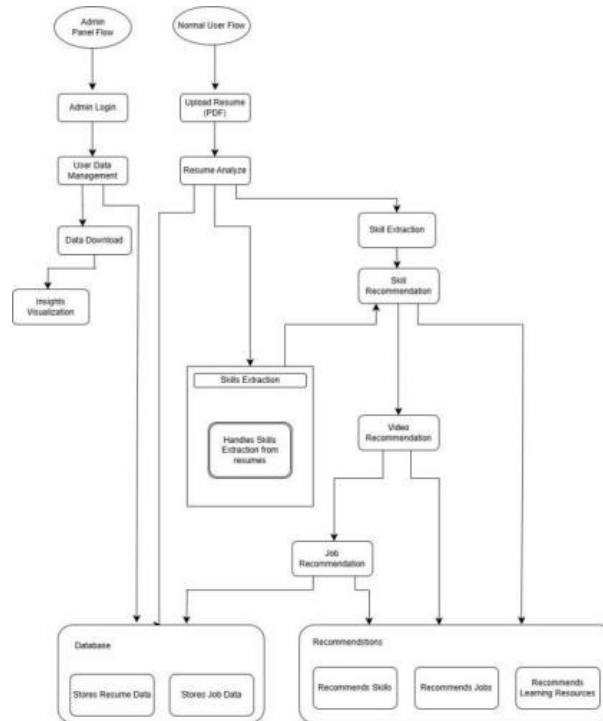


Fig.2 System Workflow

AI Resume Analyzer project, the following algorithms are employed sequentially, each serving a specific function within the system's workflow:

Keyword Matching Algorithm(Naive Bayes Classifier)

This algorithm parses the resume to identify and extract pertinent keywords related to skills, experiences, and qualifications. By matching these keywords against those in job descriptions, the system assesses the relevance of a candidate's profile to specific roles. This ensures that candidates whose resumes contain terms closely aligning with job requirements are highlighted for further consideration.

Uses Bayes' theorem with feature independence assumption:

Application

- $P(Y)$: Prior probability of eligibility.
- $P(X_i | Y)$: Likelihood of features (e.g., "Skill = Machine Learning" given eligibility).

Rule-Based Classification Algorithm(Decision Tree Classifier)

Utilizing predefined "if-then" rules, this algorithm categorizes resumes into specific job sectors or roles. For instance, if a resume mentions certain programming languages and development methodologies, the system may classify the candidate under the "Software Developer" category. This approach enables efficient and transparent classification based on explicit criteria.

Splits data using Gini impurity or entropy to maximize homogeneity.

Gini Impurity:

p_i : Proportion of class i in a node.

Entropy:

Application:

- Splits resumes based on rules like "Years of Experience ≥ 3 " or "Skill = Python".

Content-Based Filtering Algorithm (Support Vector Machine)

This algorithm recommends job roles, courses, or certifications to candidates by analyzing the content of their resumes. It identifies key attributes such as skills and experiences, suggesting opportunities that closely match these attributes. For example, a candidate with extensive data analysis experience might be recommended roles like "Data Analyst" or relevant certifications. This personalized approach enhances the relevance of recommendations.

Finds an optimal hyperplane using maximal margin classification:

Subject to:

- C: Regularization parameter.

Application

- Maps resume features (e.g., skills, education) to a high-dimensional space to separate eligible/ineligible candidates.

Scoring System Algorithm(Logistic Regression Classifier)

This algorithm evaluates and ranks resumes by assigning scores based on various criteria, including skill relevance, experience level, and educational background. Each criterion is weighted according to its importance, and the cumulative score aids in ranking candidates. This facilitates an objective comparison, streamlining the selection process by highlighting profiles that best match job criteria.

Models the probability of a binary outcome (eligible/not eligible) using the sigmoid function:

Application

- $X_1, X_2, \dots, X_n$: Features like education, skills, experience.
- Coefficients () are trained on historical data to predict eligibility.

D. Overview of Data-set

The dataset used for training the Smart Resume Analyzer model consists of multiple data sources, including resume datasets, skill taxonomy datasets, job descriptions, certification mappings, and domain-specific information. These datasets were compiled from publicly available sources such as Kaggle, GitHub, LinkedIn, Indeed, and Coursera APIs. The dataset contains over 10,000 resumes with structured information such as candidate name, contact details, education, skills, certifications, work experience, and projects. Additionally, job descriptions were collected from online job boards to match extracted resume skills with employer requirements.

TABLE I. THE DATASET DESCRIPTION

Dataset Type	Purpose	Features	Sources
Resume Datasets	Train system to parse resumes and extract relevant details	Name, contact info, skills, education, work experience, certifications, projects	Kaggle, GitHub (Resume-NER), scraped resumes (LinkedIn, GitHub, job portals)
Skill Taxonomy Datasets	Build a structured skills database and map them to job roles	List of skills, categories, job roles (e.g., Data Science, Web Development)	Job boards (LinkedIn, Indeed, Monster), ESCO Skills, job descriptions
Job Description Datasets	Match extracted skills with job requirements	Job titles, required skills, qualifications, responsibilities, industry-specific terms	Kaggle job listing datasets, LinkedIn, Glassdoor, Indeed
Certification & Course Mapping Datasets	Recommend courses and certifications for skill improvement	Course title, certification provider, related skills, difficulty level	APIs from Coursera, Udemy, LinkedIn Learning, open datasets
Domain-Specific Datasets	Tailor the system to specific industries (e.g., IT, healthcare)	Industry-specific resumes, specialized job listings, domain-specific keywords	Domain-specific datasets from job boards, customized datasets

IV. RESULTS

A. Overview of results

With this model, we take a basic resume and use it as a basis for the operations we execute, expecting the desired outcomes. A thorough explanation of the experimental outcome. The first resume to be posted is displayed in Figure 3. By looking through the file, The local system is used to upload resumes. resume is examined then it has been uploaded. Figure 5 shows The resume is currently being uploaded.



Fig. 3. The interface of the interface of proposed method



Fig. 4. The PDF that was submitted



Fig. 5. Uploaded resume

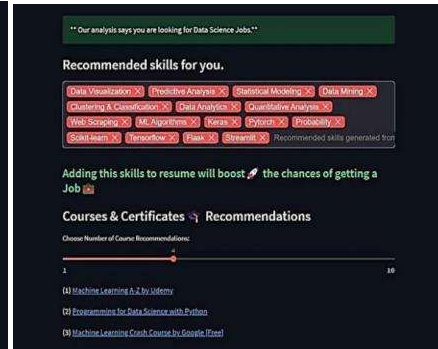


Fig. 6. Recommendation

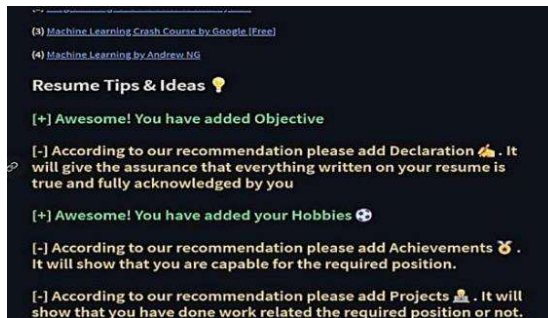


Fig. 7. Tips

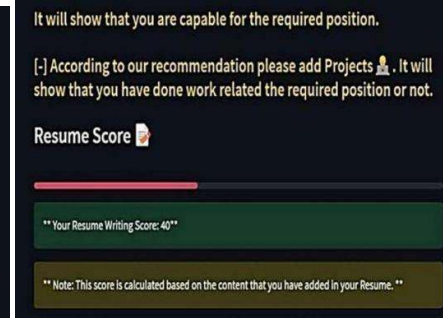


Fig. 8. Score of resume

Predicted Field according to the Skills

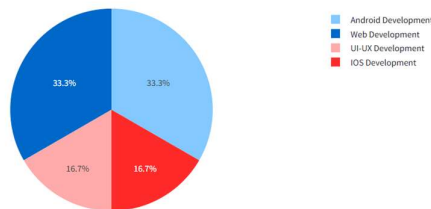


Fig 9. Predicted Field

TABLE II. ALGORITHMS DISCRIPTON

Algorithm	Precision	Recall	F1 Score	Support
Naive bayes	0.85	0.68	0.74	25
Decision Tree	0.78	0.75	0.80	34
Support Vector Machine	0.69	0.88	0.79	28
Logistic Regression	0.65	0.70	0.72	21

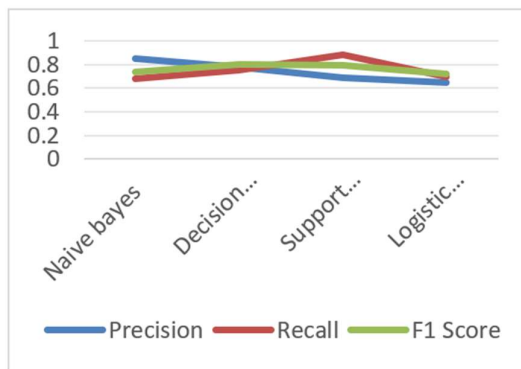


Fig.10 Graph of Accuracy

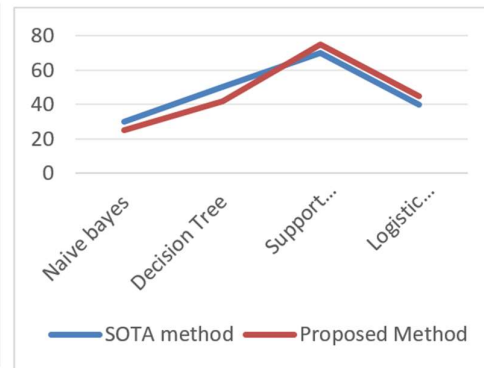


Fig 12. Support Comparison For Various Algorithm with SOTA Method

V. CONCLUSION

We accept resumes in the Word as well as PDF format with the help of file uploader and stream-lit module. Next, we use Pyresparser to extract the data from this uploaded file and keep it as a dictionary list. After that, we display the parsed data to the user and assess their abilities to determine which position would be most appropriate. We determine the user's ability gaps and provide educational materials. A rating is assigned to the resume based on a variety of criteria, including substance and skills. Links for interview preparation are suggested. The résumé is parsed and examined using the NLP and Spacy modules. The interface design is improved by the Streamlit module, and YouTube links are shown by paffy.py to help with acquiring skills that are lacking. The goal of this tool is to assist users in making their resumes more visually appealing. Future improvements might include social media data analysis to reveal work patterns and personality features. Only if the user's competencies are related to information technology professionals or come under the field of computer science will this intelligent resume analysis tool operate well. However, this suggested methodology would not be able to recommend suitable courses or produce a rating when the abilities covered by the resume extend beyond the field of computer science. It is one of the next improvements to fix this.

REFERENCES

- [1] S. Bhor, V. Gupta, V. Nair, H. Shinde, M. S. Kulkarni, Intl. J. Res. Engg. Sci 9, 6 (2021)
- [2] A. S. Vaidya, P. V. Sawant, Intl. J. Engg. Tech. Res. 3, 8 (2015)
- [3] C. Daryania, G. S. Chhabrab, H. Patelc, I. K. Chhabrad, R. Patele, Ethic Infor. Tech 2, 2 (2020)
- [4] V. Nawander, S. Elma, A. Karthikeya, M. K. Yadav, S. K. Kotala, M. D. N. Akash, Intl. Res. J. Engg. Tech 9, 10 (2022)
- [5] S.P. Pokharel, Resume parser using NLP, Tech. Rep. Uni. Wolverhampton, (2018)
- [6] M. L. Pravesh, R. Samhita, M. Mythili, S. Suprith, Intl. Res. J. Comp. Sci 9, 8 (2022)
- [7] B. Kelkar, R. Shedbale, D. Khade, P. Pol, A. Damame, Intl. Res. J. Engg. Tech 11, 5 (2020)
- [8] A. K. Sinha, A. K. Akhtar, A. Kumar, Resume Screening Using Natural Language Processing and Machine Learning: A Systematic Review. In: Swain, D., Pattnaik, P.K., Athawale, T. (eds) Machine Learning and Information Processing. Advances in Intelligent Systems and Computing, Springer, Singapore, 1311 (2020) re, 1311 (2020)