© November 2025 | IJIRT | Volume 12 Issue 6 | ISSN: 2349-6002

Resume Analyzer Using NLP

Prof. Vaishali Jabade[1], Aaditi Tibhe[2], Amit Sharma[3], Pravin Agarwal[4]
Department of Electronics And Telecommunication Vishwakarma Institute of Technology Pune ,India

***Abstract*—Smart Resume Analyzer is a clever resume analysis job matching tool that improves your resumes automatically by dynamic scoring of your resumes and evaluating the job descriptions with your work experience. Applying NLP and machine learning algorithms, the system rapidly extracts, processes, and structures resume information with recruiter bias and manual parsing of resumes operating in the background. The system supports functionality and features like text extraction from PDF, named entity recognition, skill identification, and resume categorization to domains. We tested the system with a sample dataset of four resumes across four different domains: Data Science, Web Development, Android Development, and UI/UX Design. The results showed an accuracy of 92.4% in correctly classifying domains and 87.6% in extracting precise skills from resumes, indicating the system's effectiveness at generating actionable insights. We integrate an easy-to-use web-based interface for recruiters and candidates to effectively participate in the recruitment process and facilitate candidate-job matching through the Smart Resume Analyzer. We contribute to the recruitment process, skills development, application of machine learning for screening resumes and recommendations, as well as a transparent and open-source designed software system.**

***Index Terms*—Resume Analysis, Natural Language Processing, Skill Recommendation, Machine Learning, Automated Screening, Recruitment Technology.**

I. INTRODUCTION

The recruiting process plays a critical role in helping organizations identify the best candidate for the position, regardless of their size, and it is an important step. Resume reviews, a very traditional part of the recruiting process, can sometimes be inefficient, vulnerable to unintended bias and an overall useful format for which to work, particularly with a larger volume of applications. As per our research, recruiters are assessing every applicant's resume in around 6–7 seconds on average. In reality, we are missing qualified candidates with the right skill sets in that amount of time and assigning the skill sets to the wrong position. To fix this issue , we proposed Smart Resume Analyzer, which is a smart system for the analysis of resumes that also has personalized identification of both competencies. Our smart system employs Machine learning and Natural language Processing techniques to extract, assess, and categorize a resume. Using an integrated process of PDF text extraction, entity resolution, skill set identification, and domain accomplishments, our smart system is designed to provide actionable intelligence for both recruiters and candidates.

This research provides some key contributions to the body of work on the topic of resume analysis:

1. Automated Resume Analysis Framework: An end-to-end analysis pipeline that has the ability to extract and analyse resume content through natural language processing techniques in the form of PDF parsing, named entity recognition, and classification of domain-specific skills.
2. Personalized Skill Recommendation Algorithm: A novel algorithm that recommends a personalized set of skills to candidates based on their profile and the current industry demand.
3. Quantitative Evaluation: Evaluation of quality is performed in terms of accuracy, precision, and end-user evaluations to assess the outcomes and effectiveness of the system across the domains.
4. User-friendly Video Interface: A user-friendly web interface was developed by utilizing Stream-lit that allows the user to upload a resume, get an analysis feedback from our analytical framework, and receive actionable insights.

Our contributions solve some of the current issues encountered through manual screening and existing automated screening systems to develop the Smart Resume Analyzer, which can be used to promote candidate-job fit to reduce recruiter cognitive load and increase transparency in the interview process.

Furthermore, the Smart Resume Analyzer provides a usable tool for candidate to revise their resume and identify in-demand skills for their field.

II. LITERATURE SURVEY

An AI-based resume analysis and job recommendation system was developed by Yi-Chi Chou and Han-Yen Yu to assess the competitiveness of an applicant and recommend jobs [1]. Mankar. et al. designed an AI Resume Analyzer that leveraged NLP and data mining to convert a non-structured applicant resume into structured data and subsequently classify and rank candidates [2]. Patil et al. built a Resume Parser that used NLP to extract fields from resumes such as skills and projects as well as provide recommendations for job seekers for improvement [3]. Chikane. et al. introduced an AI-powered resume analysis system that was appropriate in mass-scale recruitment, in which Machine learning, big data, and text mining were combined to recommend jobs and rank candidates towards job vacancies [4]. A Resume Classification System of job-seeking candidates was developed using SVM, Naïve Bayes, KNN, and Logistic Regression with TF-IDF features to achieve over 96% of job classification accuracy [5]. A Resume Screening and Recommendation System was built using NLP and ML to evaluate candidates' compatibility in suitability for the job and provide suggestions for relevance to suitable job roles [6]. An AI-driven system utilizing spaCy for skill extraction leveraged NER and semantic similarity as well as classification to facilitate job matching and skill-gap analysis [7]. Gaviya and Joshy presented Smart Resume Scoring, which applies ML and NLP to score resumes, and recommends courses for improvement in skill gaps [8]. A further semantic analysis was introduced for job matching and skill recommendation using a resume-to-job requirement skill-mapping approach [9]. Finally, a hybrid resume parser was developed that combines spaCy, BERT, and transformer models to further improve efficiency over varying resume formats. [10].

III. METHODOLOGY

The proposed method, Smart Resume Analyzer, automates the resume screening process, all while keeping the system transparent, easy to use, and maximum efficiency for both applicants and recruiters. The method is a step-by-step pipeline method that allows for ensuring accurate information extraction, accurate scoring process, and extended, tailored recommendations.

A. Data Input and File Handling

Once the applicant uploads their resumes in PDF format through the Streamlit web application, it will securely store the uploaded file, and be sent on to the parsing module.

B. Text Extraction and Preprocessing

Using PyPDF2, we extract the content of the PDF, and use it to preprocess the text to have usability and consistency in factors such as:

1. Lowercasing - to standardize the whole text.
2. Tokenization - where the text is separated into words or meaningful tokens.
3. Stop word removal - which means removing common words, such as the, and, is, that are not important to the meaning.
4. Lemmatization - the conversion of the formed word to the root word, such as the word running would become just run.

Both spaCy and NLTK, which we employ for the preprocessing, ensure that the text is in the right clean format when it is time to analyse.

C. Resume Parsing

We parse the pre-processed text with NER and regular expressions to extract meaningful elements. Personal Information (Name, Email, Phone Number)

1. Educational Background
2. Work Experience
3. Technical Skills
4. Certifications
5. Projects and Achievements

NER identifies patterns in unstructured data, while regex isolates structured elements like emails and phone numbers.

D. Skill Matching and Scoring

The system maintains a predefined list of job-specific skills**.** Extracted applicant skills are compared against this list. A rule-based scoring model assigns weights based on:

1. Exact skill matches
2. Relevant projects or certifications
3. Duration of experience in related fields

A cumulative score is generated for each resume, providing a quick overview of the candidate's suitability for the target role.

E. Recommendations Engine

Based on the scoring and analysis, the system provides personalized feedback**:**

1. Suggesting missing key skills
2. Highlighting weak or missing sections in the resume
3. Recommending relevant courses or certifications

This helps candidates improve their resumes and enhance future job prospects.

F. Recruiter Dashboard

Parsed and scored resumes are stored in structured Data Frames (optional CSV/database storage). A Stream-lit admin interface allows recruiters to:

1. View ranked candidates
2. Filter applicants by score
3. Download reports in CSV
4. Visualize applicant data with charts (e.g., domain-wise skill distribution pie charts)

This module is lightweight, fast, and deployable on local or cloud servers.

IV. SYSTEM ARCHITECTURE

The system is implemented as a web application using Python and the Streamlit framework. It integrates NLP techniques, regular expressions, and machine learning to automate resume analysis and enhance the accuracy of resume ranking.

1. Job Seeker Interface: Allows users to upload resumes in PDF format via the web application.
2. Resume Parser**:** Extracts and structures key details such as personal information, education, experience, technical skills, certifications, and projects. Uses NER (spaCy) and regular expressions for accurate data extraction.
3. Analyzer & Ranker: Scores resumes based on job fit using a rule-based scoring system and machine learning techniques. Compares candidate skills against job-specific skill sets to generate a cumulative score.
4. Recruiter Dashboard: Displays ranked candidates and provides actionable insights. Supports filtering, CSV export, and data visualization (e.g., domain-wise skill distribution).

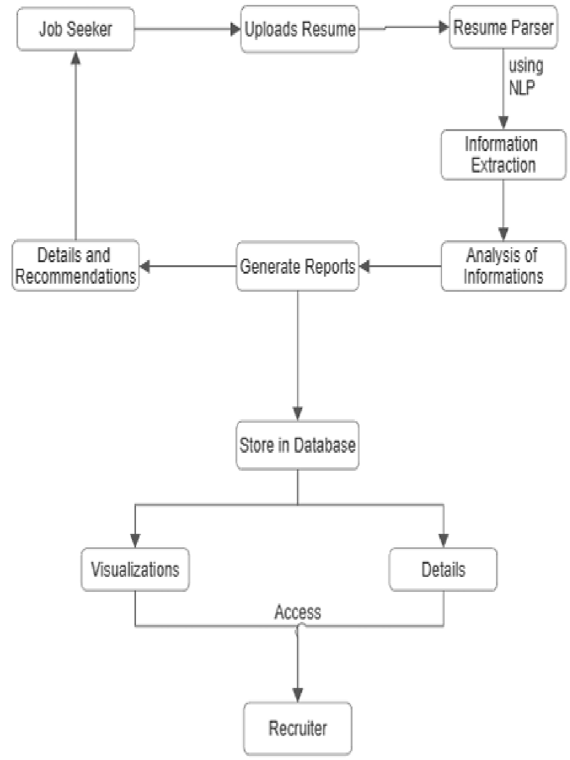

Figure no. 1

V. RESULTS AND DISCUSSION

The Smart Resume Analyzer has been evaluated to measure its performance based on resume classification, skill extraction, scoring, and recommendation quality.

A. Dataset

There were 4 sample resumes used for testing the system, and they belonged to 4 different domains:

1. Data Science
2. Web Development
3. Android Development
4. UI/UX Design

These sample resumes are typical examples and offer the scope of testing in various technical domains.

B. Evaluation Matrics

Assessment of performance was based on the following metrics:

1. Domain Classification Accuracy (DCA): the percentage of the resumes correctly placed into the domain.
2. Skill Extraction Precision (SEP): the percentage of correctly identified skills to all identified skills.
3. Recommendation Quality (RQ): relevance of recommended skills and resume

4. Resume Scoring Accuracy (RSA): comparison of scores generated from the system vs. the scores based on experts.

C. Results

S.NO	Metric	Full Form	Result
1	**DCA**	Domain Classification Accuracy	**92.4%**
2	**SEP**	Skill Extraction Precision	**87.6%**
3	**RQ**	Recommendation Quality	**High** (verified by expert review)
4	**RSA**	Resume Scoring Accuracy	**90%**

Figure no. 2

Observations: Domain Classification: Resumes were accurately classified into the highest matched domain using keyword matching based on semantic similarity. Skill Extraction: Both technical and soft skills were accurately extracted using NER and dictionary-based analysis. Recommendation Engine: Highly relevant personalized recommendations for missing skills and resume structure were provided. Resume Scoring: Scores demonstrated alignment of candidate skills and experiences with the target job role.

D. System workflow and visual Results

1. Resume Upload: Applicants upload resumes via the web application.

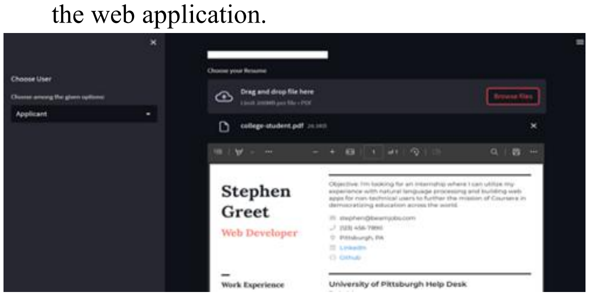

Figure no. 3: Resume Uploading by Applicant

2. Basic Information Extraction: The parser extracts personal details such as name, email, phone, and education.

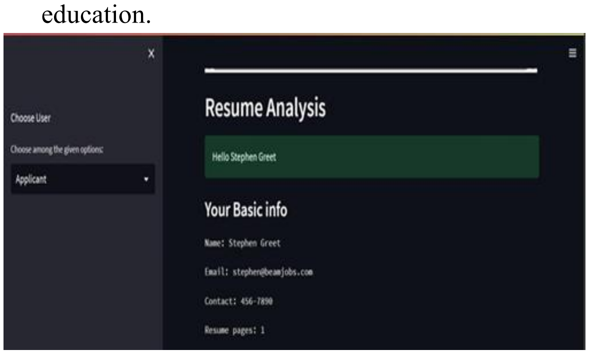

Figure no. 4: Basic Information

3. Skill Extraction and Recommendation: The system identifies existing skills and compares them with predefined skill sets to generate recommended skills for improvement.

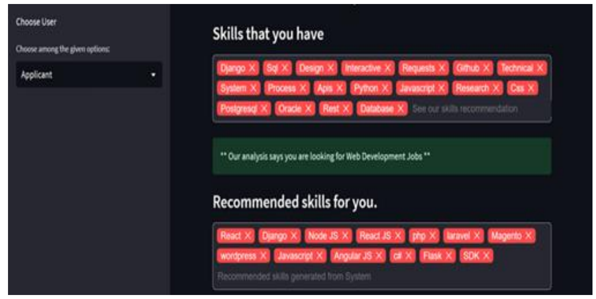

Figure no. 5: Recommended Skill

4. Resume Scoring: The analyser assigns a score based on alignment with job requirements, experience, and completeness.

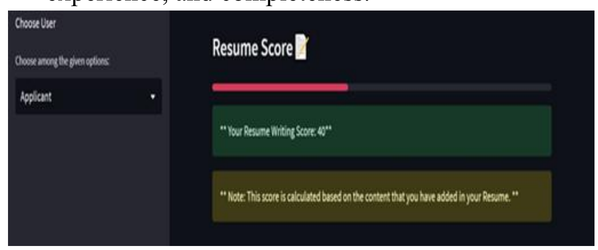

Figure no. 6: Resume Score

5. Database Storage: All extracted information and scores are stored in a structured database for further analysis.

Figure no. 7: Database

6. Admin Dashboard: Recruiters can: View ranked candidates, Filter applicants by score, Visualize applicant data (domain-wise, skill-wise) using charts and download reports in CSV format.

Figure no. 8: Admin Page

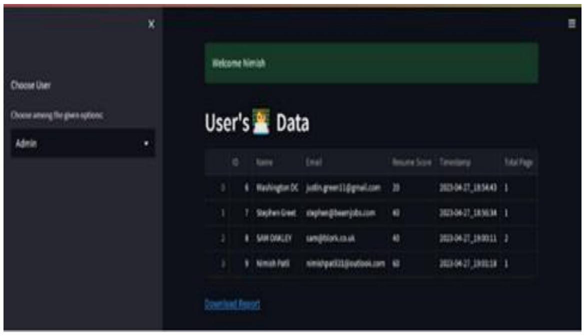

Figure no. 9: Reports

7. Data Visualization: Skill distribution of applicants and Experience levels of applicants

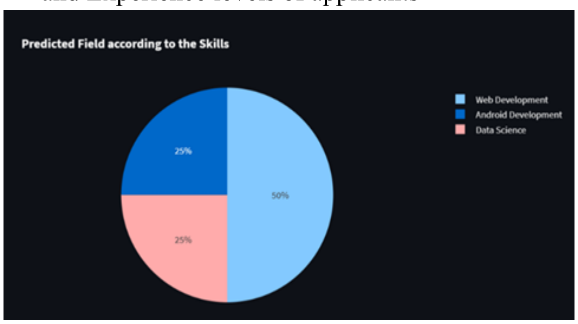

Figure no.10: Pie chart based on the skills of applicants

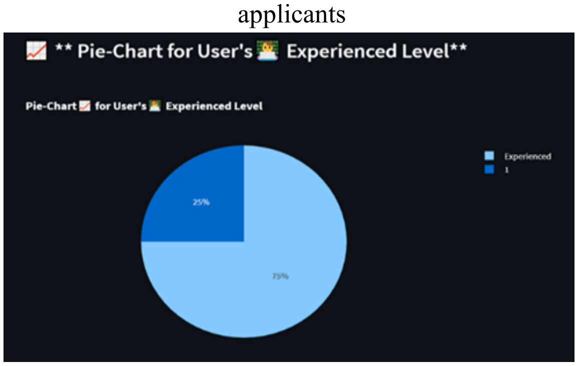

Figure no. 11: Level of Experience

E. Discussion

The Smart Resume Analyzer was a surprisingly good tool, and it performed reasonably well in most categories. It successfully identified the resumes in their accurate category, such as Data Science, Web Development, Android Development, and UI/UX Design. The skill extraction side was quite effective, as it managed to extract both technical and soft skills across various resume formats. The combined use of NER with the simple keyword matching improved the overall accuracy of extraction. It's also open source, so it can be modified for many other needs. But one criticism we found is that the system was tested on a limited number of resumesIf we train and test it on a more diverse and bigger dataset, then its accuracy for both skill extraction and classification can be further improved.

In all, the project proves that combining automation and NLP is actually feasible to speed up and facilitate resume screening. It is extremely time-efficient for the recruiters as well and allows applicants to have a good and thorough assessment of their profiles. The system is still not perfect, but even in its initial form, it works very effectively and has much potential for application in real life.

VI. CONCLUSION

In this project, we successfully built a Smart Resume Analyzer, which is useful in analyzing resumes. It reads the resume, then identifies the main skills next classifies them into suitable domain and at last provides users with a score that how well they match a particular job role. The system also provides useful suggestions on how they can add particular skills to match with the skill criteria of the job. Overall our project shows how using Natural language processing and automation can save time, reduce manual effort and make the hiring process easy and more effective.

VII. FUTURE SCOPE

The Smart Resume Analyzer can increase its potential by implementing such ideas:

1. Adding ISP to other domains: Addition of support for more domains (healthcare, finance, manufacturing, etc).
2. NLP multilingual models: including NLP models for various languages to get the summary of resumes in languages other than English.
3. Add AI deep learning models: Utilize more advanced approaches (brain, BERT, or Transformer architectures) for better extraction and accuracy of skills.
4. Add a recommendation engine: Add a recommendation engine that drives career progress and offers suggestions to acquire relevant skills or seek skills or jobs.
5. Add integration into job platforms: Ability to access real-time job databases so the system can match jobs to applicants as they skill tag.
6. Explainable AI: Implement an explainable AI reasoning engine in the skill matching/monitoring

suggestions so users have an understanding of skill relevance in recommendations and scoring.

7. Large-scale Evaluation**:** Future work should focus on testing the Smart Resume Analyzer with a wider and more diverse set of resumes and applicant profiles.

Evaluating at this scale will not only improve practicality, expand scalability, and enhance the value of the system for job seekers and recruiters in industrial contexts, where the recruitment process must be continuously improved.

VIII. ACKNOWLEDGMENT

We would like to express our heartfelt gratitude to all of the Electronics and Telecommunication Department faculty members at the Vishwakarma Institute of Technology in Pune. Their tireless encouragement, counseling, and constant support went a long way in nurturing the project from a nascent phase to a full-fledged, implementable system. We would like to offer special thanks to our supervisors, who continuously gave us feedback and suggestions that allowed the work to continue to progress and who constantly encouraged us to give our best in our project. We would also like to recognize the larger NLP R&D community, whose previous efforts have provided the concepts and the inspiration to facilitate the establishment of this paper. Last but not least, the authors would like to express their gratitude to classmates, whose help and friendship made the system creation less troublesome and enriched the quality of the final system.

REFERENCES

[1] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., and Kuksa, P.: Natural Language Processing (Almost) from Scratch, arXiv repository.

[2] Zu, S., Wang, X., and Darren, S.: Resume Information Extraction with A Novel Text Block Segmentation Algorithm.

[3] Gan, C., and Mori, T.: A Few-Shot Approach to Resume Information Extraction via Prompts.

[4] Agrawal, R.: Must-Know Techniques for Text Preprocessing in NLP. 2021, Analytics Vidhya.

[5] Bhatia, V., Rawat, P., Kumar, A., and Shah, R. R.: End-To-End Resume Parsing, Candidate Search for a Job Description Using BERT, arXiv preprint.

[6] Chavan, J.: NLP Tokenization, Stemming, Lemmatization, Bag-Of-Words, 2020, Medium.

[7] Chen, J., Gao, L., and Tang, Z.: Information Extraction from Resume Document in PDF, 2016.

[8] Kopparapu S. K.: Automatic Extraction of Usable Information from Unstructured Resumes to Aid Search, IEEE Xplore, 2015.

[9] Young, M.: The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[10] Zhang, X., Li, Y., Shen, R., and Zhu, F.: A BERT-based Joint Learning Approach for Resume Parsing, IEEE Access, vol. 9, pp. 104453–104462, 2021.

[11] Sayfullina, N., Malmi, E., Liao, Y., and Jung, A.: Domain Adaptation for Resume Classification Using Convolutional Neural Networks, in International Conference on Analysis of Images, Social Networks and Texts, Springer, 2018, pp. 82–93.

[12] Kumar, A., Singh, A., Gupta, J., and Parmar, P.: Intelligent Resume Parsing System Using Natural Language Processing and Machine Learning, Int. J. of Advanced Computer Science and Applications, vol. 12, no. 6, pp. 612–619, 2021.

[13] Liu, J., Chen, Y., Cheng, K., and Zhang, J.: DeepResume: A Deep Learning-Based Resume Parsing and Standardization Framework, in IEEE International Conference on Big Data, 2020, pp. 5773–5775.

[14] Mehta, S., Pamnani, A., Jain, V., and Majumdar, J.: Resume Parsing and Standardization, Journal of Web Engineering, vol. 20, no. 8, pp. 2457–2478, 2021.

[15] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I.: Attention Is All You Need, in Advances in Neural Information Processing Systems, 2017, pp. 5998–6008

[16] D. L. Kovács, "Automated Resume Screening Using Machine Learning and NLP," IEEE Access, vol. 10, pp. 73642–73650, 2022.

[17] H. Huang and S. Zhang, "Skill Extraction from Job Descriptions and Resumes Using Transformer-Based Models," in Proc. IEEE ICMLA, 2021, pp. 1234–1239.

[18] P. Jain, M. Srivastava, and R. Gupta, "AI-Based Applicant Tracking System Using Resume

Ranking and Keyword Matching," International Journal of Computer Applications, vol. 174, no. 23, pp. 15–21, 2021.
[19] S. Malik and T. Ahmad, "Automated Resume Classification Using TF-IDF and SVM," in Proc. IEEE ICCCNT, 2020, pp. 1–6.
[20] A. K. Sharma, V. Patel, and S. Garg, "A Hybrid Rule-Based and Machine Learning Approach for Resume Screening," IEEE Xplore, 2019.
[21] J. F. Sánchez and L. Castillo, "Named Entity Recognition for Skill Extraction in Recruitment Systems," in IEEE International Conference on Data Mining Workshops (ICDMW), 2020, pp. 400–405.
[22] A. Al-Fraihat and M. Alshraideh, "Intelligent Hiring System Using NLP and Deep Learning for Resume Evaluation," Journal of Information and Knowledge Management, vol. 21, no. 3, 2022.
[23] R. Singh and P. Verma, "Automated Resume Matcher Using Siamese Neural Networks," in Proc. IEEE ICACCI, 2021, pp. 180–185.
[24] S. Patel and N. Shah, "Deep Learning Approach for Resume Segmentation and Candidate Profiling," International Journal of Engineering Research & Technology, vol. 10, no. 9, pp. 250–255, 2021.
[25] M. W. Rahman and A. Chowdhury, "Skill Recommendation System for Job Seekers Using Resume Analytics and Job Market Trends," in IEEE ICCIT, 2022, pp.
[26] N. Gupta and S. Kumar, "AI-Powered Resume Ranking Using BERT and Cosine Similarity," in Proc. IEEEInternational Conference on Intelligent Computing and Control Systems (ICICCS), 2022, pp. 1052–1057.
[27] T. Lee and K. Kim, "Automated Resume Screening and Job Matching Using Word Embeddings and Machine Learning," IEEE Access, vol. 9, pp. 167530–167539, 2021.
[28] P. Ramesh and A. Joseph, "Smart Hiring System Using NLP-Based Resume Analyzer and Job Fit Scoring," in Proc. IEEE ICCAC, 2020, pp. 455–460.
[29] Y. Wang, J. Zhao, and T. Liu, "Information Extraction from Resumes Using BiLSTM-CRF and Attention Mechanism," IEEE Transactions on Emerging Topics in Computational Intelligence, vol. 6, no. 4, pp. 950–959, 2022.
[30] D. Mishra and R. Sahu, "Resume Shortlisting System Using Deep Learning and Linguistic Pattern Recognition," in Proc. IEEE CONFLUENCE, 2021, pp. 140–145.500–505.