Resume Parser and Job Description Matcher

1st Spandana Anilkumar Kamkar
Dept. of Computer Science and Engineering
BNM Institute of Technology, Affiliated to VTU,
Bangalore, India
spandana.kamkar@gmail.com

2nd Srushti Sanjay
Dept. of Computer Science and Engineering
BNM Institute of Technology, Affiliated to VTU,
Bangalore, India
srushtisanjay313@gmail.com

3rd Vaishnavi P S
Dept. of Computer Science and Engineering
BNM Institute of Technology, Affiliated to VTU,
Bangalore, India
vaishnavips2020@gmail.com

4th Chaitra M
Dept. of Computer Science and Engineering
BNM Institute of Technology, Affiliated to VTU,
Bangalore, India
chaitram@bnmit.in

2024 8th International Conference on Computational System and Information Technology for Sustainable Solutions (CSITSS) | 979-8-3315-0546-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/CSITSS64042.2024.10816751

***Abstract*—Resume Parser and Job Description Matcher is an innovative project aimed at enhancing the efficiency and accuracy of job candidate screening and matching processes. Leveraging natural language processing (NLP) techniques, particularly using SpaCy and machine learning models, the system parses resumes and job descriptions to extract relevant skills, experience, and qualifications. By analyzing textual data and applying advanced algorithms, such as Support Vector Machines (SVM), the system learns to match candidates with job postings based on keyword relevance and contextual understanding. Our paper demonstrates improved matching results compared to traditional methods, facilitating streamlined recruitment processes, and enhancing workforce management strategies.**

Keywords—Resume parsing, Job description matching, Natural language processing (NLP), Machine learning, spaCy, Support Vector Machines (SVM), Skills extraction, Experience recognition, Recruitment automation.

I. Introduction

The Resume Parser and Job Description Matcher is revolutionizing the hiring process by employing advanced NLP techniques and machine learning, particularly Support Vector Machines (SVM), to assess and compare candidates more effectively. This innovative system addresses common challenges found in traditional recruitment methods, such as inefficiency, reliance on manual labor, and potential bias in candidate selection. By automatically extracting key skills and qualifications from both resumes and job descriptions, the project streamlines the comparison of candidates, making it easier for recruiters to match candidates with job roles based on relevant skills and experience [1].

SVM models are trained on large datasets to identify patterns in candidate profiles and job descriptions, enabling the system to make more accurate and efficient matches between candidates and job openings. This automation reduces the manual effort required to sift through large volumes of resumes, significantly speeding up the initial screening process while minimizing the risks of human errors and bias. By eliminating these bottlenecks, recruiters can focus their efforts on more strategic activities, such as engaging candidates and conducting interviews, thereby increasing overall hiring efficiency [2].

Through the integration of NLP and SVM technology, the system enhances the hiring process by ensuring that recruitment is more efficient, fair, and aligned with an organization's goals. This approach not only enables smarter decision-making but also leads to higher-quality hires, lower turnover rates, and reduced recruitment costs for companies. As a result, this paper represents a significant breakthrough in modernizing recruitment practices and improving the overall effectiveness of hiring processes [3].

II. Problem Statement

Issues faced by the traditional system:

Traditional recruitment processes face several significant challenges that hinder their effectiveness and efficiency. One major issue is the reliance on manual resume screening. This process is inherently time-consuming, as HR professionals must sift through numerous resumes, leading to substantial delays in candidate evaluation. Moreover, manual screening is error-prone; human errors can result in the overlooking of qualified applicants. As organizations often receive large volumes of resumes, managing this influx exacerbates the problem, causing further delays and increasing the likelihood of oversights.

Additionally, the subjective nature of manual reviews can introduce biases into the recruitment process. These biases can affect the fairness and consistency of candidate assessments, as different reviewers may have varying opinions on what constitutes a strong candidate. This subjectivity can lead to inconsistency in evaluations, making it difficult to ensure that the best candidates are selected. The lack of standardized evaluation methods further complicates this issue, resulting in discrepancies between candidate profiles and job requirements. This inconsistency can lead to a mismatch between the skills and experiences of the candidates and what is needed for the job, thereby affecting the overall quality of hires. Keyword-based matching, which is a common method used in traditional recruitment, also presents significant limitations. This approach tends to be rigid, focusing narrowly on specific keywords in resumes and job descriptions. As a result, it may miss relevant skills and experiences that are not explicitly stated in the job descriptions but are nonetheless valuable [4].

Issues faced by the current system:

Current recruitment systems, while an improvement over purely manual processes, still face limitations. These systems often struggle with handling diverse resume formats and extracting nuanced information accurately. They may also have difficulty processing unstructured data, which is common in resumes and job descriptions. Additionally, many existing systems lack robustness in handling multi-language resumes, limiting their effectiveness in global recruitment efforts. Furthermore, these systems are not always adaptive to evolving job market demands, leading to outdated or irrelevant matching criteria. These challenges highlight the urgent need for more advanced, adaptive, and unbiased recruitment solutions that can efficiently handle diverse and evolving job market requirements. Advanced technologies, such as machine learning and natural language processing, hold promise in addressing these challenges by providing more accurate, fair, and efficient recruitment processes [5].

III. Related work

Research in resume parsing and job matching focuses on various methodologies and technologies to improve automation and accuracy in recruitment. One key technique is Named Entity Recognition (NER), used to extract details like names, contact information, education, work experience, and skills from resumes. Modern NER systems often employ deep learning models, such as transformers, which outperform traditional rule-based methods in structuring unstructured resume data for analysis and matching.

In addition to NER, deep learning models like Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and transformers such as BERT and GPT are instrumental in capturing and interpreting contextual information from resumes and job postings. These models excel at understanding long text sequences, improving the accuracy of resume parsing and job matching by identifying complex patterns and requirements [15].

Data mining techniques, including clustering and text classification, further automate candidate-job matching by analyzing semantic similarities and skill overlaps. With the help of computational linguistics and NLP libraries like spaCy, text preprocessing, and feature extraction have become more efficient, making modern resume parsing and matching systems more robust and accurate, ultimately enhancing the recruitment process [6].

IV. Proposed method

A. Experiment Method and Environment

In this paper, a robust system for resume parsing and job description matching is proposed, utilizing spaCy for linguistic preprocessing and Support Vector Machines (SVM) for classification tasks. Accurate parsing of resumes is crucial for extracting relevant information such as skills, work experience, and educational background, which forms the foundation for effective candidate-job matching. The system preprocesses textual data through tokenization, named entity recognition (NER), and part-of-speech (POS) tagging using spaCy's advanced linguistic annotations.

After preprocessing, the system utilizes SVM for classification to match candidate profiles with job descriptions based on the extracted features. SVM is chosen for its effectiveness in handling high-dimensional data and capturing nonlinear relationships between textual features, making it well-suited for the complexities of resume and job description data.

The proposed workflow involves several key steps:

1. Data Collection and Preprocessing:

   Data Collection: Gather a diverse dataset of resumes and job descriptions in various formats (PDF, DOCX, TXT) and languages to ensure the system can handle a wide range of inputs.

   Data Cleaning: Clean the textual data to remove noise, standardize formats, and resolve inconsistencies. This includes handling misspellings, removing irrelevant information, and normalizing text to a common structure.

   Preprocessing: Utilize spaCy to tokenize the text, perform lemmatization, and apply NER and POS tagging. This step converts the raw text into a structured format, enabling efficient feature extraction.

2. Feature Extraction:

   Tokenization and Lemmatization: Break down the text into individual tokens (words) and reduce them to their base forms (lemmas) to standardize variations of the same word.

   Named Entity Recognition: Identify and extract key entities such as names, organizations, dates, and skills from the resumes and job descriptions.

   POS Tagging: Tag each token with its part of speech to understand the grammatical structure of the text.

   Feature Vector Creation: Create feature vectors representing the candidate's skills, experience levels, and educational qualifications. These vectors are essential for the SVM classification process.

3. Model Training and Evaluation:

   Training: Implement SVM classifiers to categorize resumes into relevant job categories based on the extracted feature vectors. Train the model on a labeled dataset of resumes and job descriptions.

   Evaluation: Assess the model's performance using metrics such as accuracy, precision, recall, and F1-score. Perform cross-validation to ensure the model's robustness and generalizability.

4. Matching and Recommendation:

   Similarity Scoring: Develop a matching algorithm to compute similarity scores between candidate profiles and job descriptions using the SVM predictions. This involves comparing feature vectors and calculating the degree of match.

   Recommendation Engine: Recommend top candidates for specific job postings based on the computed similarity scores, ranking candidates by their suitability for the role.

5. Integration and Deployment:

   System Integration: Integrate the resume parsing and matching system with existing HR management platforms or applicant tracking systems (ATS) to streamline the recruitment workflow.

Deployment: Deploy the model on a cloud platform to ensure scalability and real-time processing of candidate applications. This setup allows the system to handle large volumes of data and provide timely recommendations.

By following these steps, the proposed system aims to significantly enhance the efficiency and accuracy of the recruitment process. The integration of spaCy for linguistic preprocessing and SVM for classification creates a powerful tool for matching candidates with suitable job opportunities, ultimately benefiting both job seekers and employers [7].

B. Datasets

The dataset used in this paper comprises a collection of anonymized resumes and corresponding job descriptions from a diverse range of industries and job roles. Each entry in the dataset includes comprehensive textual data encompassing key elements such as skills, work experience, educational background, and job responsibilities. This rich and varied information is crucial for effectively training and evaluating the resume parser and job description matcher, ensuring that the system can accurately extract and interpret relevant details from resumes and match them with appropriate job descriptions [14].

To facilitate a thorough and diverse evaluation, the dataset includes job descriptions from various companies, each representing different job roles, locations, and required levels of experience. The inclusion of entries from multiple industries and job types ensures that the system is exposed to a wide array of terminology, formats, and contextual nuances. This diversity helps in developing a more robust model capable of handling the variations typically encountered in real-world recruitment scenarios.

Additionally, the dataset is organized in a tabular format where each row corresponds to a distinct job role, including specific details such as the job title, location, and required job experience. This structured format aids in the systematic analysis and processing of the data, allowing for efficient feature extraction and model training. The varied nature of the dataset supports the development of a versatile and comprehensive system that can generalize well across different job markets and application domains [8].

Table I provides the list of job descriptions for various companies. Every row indicates a distinct job role, location, and job experience.

TABLE I. DATASET USED IN THE PROJECT

Job_Role	Company	Location	Job Experience	Skills/Description
Senior Data Scientist	UPL	Bangalore/Bengaluru, Mumbai (All Areas)	3-6	python, MLT, statistical modeling, machine learning, IT Skills, advanced analytics, scala, statistics
Senior Data Scientist	Walmart	Bangalore/Bengaluru	5-9	Data Science, Machine learning, Python, Azure, BiqQuery, GCP, PySpark, tensorflow
Applied Data Scientist / ML Senior Engineer (Python / SQL)	SAP India Pvt.Ltd	Bangalore/Bengaluru	5-10	Python, IT Skills, Testing, Cloud, Product Management, SAP, Cloud computing, NLP
Data Scientist	UPL	Bangalore/Bengaluru, Mumbai (All Areas)	1-4	python, machine learning, Data Science, data analysis, aws, azure
Data Scientist	Walmart	Bangalore/Bengaluru	4-8	IT Skills, Python, Data Science, Machine Learning, Big Data, Computer science, Computer vision, deep learning

C. Data Pre-Processing

Data preprocessing is a critical step in enhancing the accuracy and efficiency of the resume parser and job description matcher. The preprocessing pipeline involves several key steps to ensure that the textual data is clean, consistent, and suitable for analysis:

1. Text Normalization: Text normalization is a crucial step in resume parsing and job description matching. It involves converting all text to lowercase to standardize casing, ensuring words like "Engineer" and "engineer" are treated the same. Punctuation marks are removed to clean up the text, preventing noise that could affect analysis accuracy. Special characters are also handled to maintain consistency, allowing the system to focus on relevant content without interference from unnecessary symbols.
2. Tokenization and Lemmatization: Tokenization involves breaking text into smaller units, called tokens, such as words or phrases, for easier analysis. Lemmatization complements this by reducing tokens to their base forms, or lemmas. For instance, "running" and "ran" are reduced to "run," ensuring uniform word representation. This process improves semantic analysis, allowing the system to recognize different forms of a word as equivalent [9].
3. Named Entity Recognition (NER): Named Entity Recognition (NER) is key to identifying and categorizing entities like names, organizations, locations, and dates within resumes and job descriptions. It helps extract crucial details, such as candidates' previous employers, job titles, educational institutions, and skills. This precise information is vital for accurately matching candidates to job requirements, enabling the system to compare qualifications and experiences effectively.
4. Feature Engineering: Feature engineering involves extracting key information from text, such as skills, experience levels, and educational qualifications, which are essential for assessing candidate suitability. These features are used to create vectors representing both candidate profiles and job requirements. Effective feature engineering is critical for aligning profiles with job descriptions, as it allows the system to quantify and compare various aspects of candidates' qualifications for more accurate matching [10].
5. Data Standardization: Data standardization ensures consistency in feature representation and numerical values across datasets. It involves normalizing data to a common scale and improving the reliability of machine learning models like SVMs for tasks such as predicting candidate-job fit. Standardization minimizes variability and biases from different data formats or collection methods, ensuring model predictions are based on consistent criteria. This is crucial when handling diverse datasets, as it ensures comparability between features from various resumes and job descriptions [11].

By following these preprocessing steps, the system ensures that the textual data is clean, consistent, and structured to maximize the accuracy and efficiency of the resume parser and job description matcher. These techniques are crucial for preparing the data for feature extraction, model training, and evaluation, ultimately building a robust and reliable recruitment tool.

D. Flowchart

The Figure 1 given below depicts the system workflow for the resume parser and job description matcher, which begins with the user uploading their resume in formats such as PDF, DOCX, or TXT.

Once uploaded, the resume is parsed using the Doc2Vec algorithm, a deep-learning model that represents documents as vectors in a continuous vector space, capturing the semantic meaning of the text. This vector representation allows the system to understand the context and relationships within the resume. Following this, the system extracts specific skills mentioned in the resume by identifying keywords and phrases related to professional skills, technical expertise, and competencies. These extracted skills are essential for matching the resume with job descriptions.

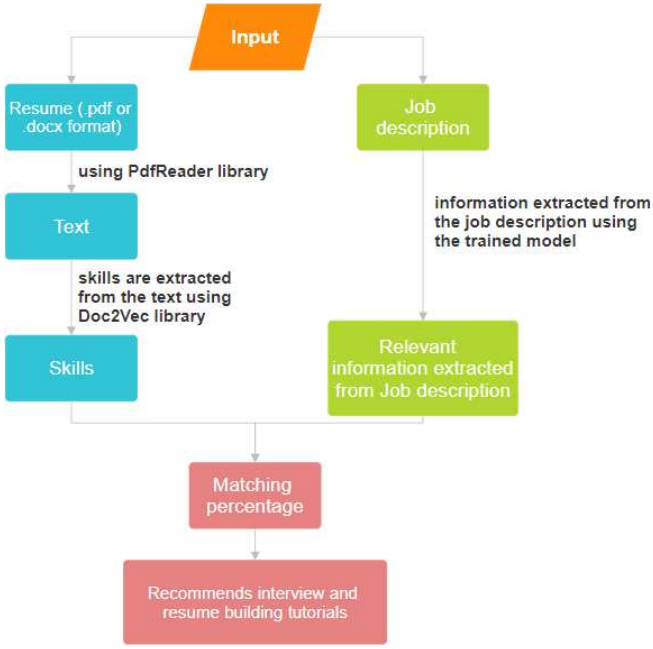

Fig. 1: Flow Chart

Simultaneously, the job description is provided as an input and processed similarly using the Doc2Vec algorithm to create a vector representation. This representation helps in understanding the job requirements in a detailed manner. The system then matches the skills extracted from the resume with the job description, computing a match percentage that indicates how well the candidate's skills align with the job requirements. Additionally, based on the match results, the system provides recommendations for YouTube tutorials to help candidates improve their resume and interview skills. These tutorials are aimed at enhancing the candidate's qualifications and preparation for job applications, thereby increasing their chances of success in the recruitment process. This comprehensive approach ensures a thorough evaluation and provides actionable feedback to candidates.

E. Performance Evaluation Methods

To ensure the effectiveness of the resume parser and job description matcher, a comprehensive set of performance evaluation metrics is employed. These metrics provide a thorough assessment of the model's ability to accurately parse resumes and match them with job descriptions.

1. Accuracy: Accuracy is a fundamental metric that measures the proportion of correct predictions made by the model out of the total number of predictions. It provides a general sense of how well the model is performing. Accuracy is calculated as the ratio of true positives and true negatives to the total number of instances. In the context of resume parsing and job matching, accuracy indicates the overall effectiveness of the system in correctly identifying and matching skills and qualifications.

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN) \quad (1)$$

2. Precision: Precision measures the proportion of true positive predictions (correct matches) among all positive predictions (all matches) made by the model. It focuses on the quality of the positive predictions, ensuring that the model is not overestimating matches. Precision is particularly important in recruitment scenarios, as it helps in understanding how many of the candidates identified as suitable are indeed qualified for the job roles. A high precision score indicates that the model is effective in minimizing false positives.

$$\text{Precision} = (TP) / (TP + FP) \quad (2)$$

3. Recall: Recall, also known as sensitivity, measures the proportion of true positive predictions among the actual positives (relevant matches) in the dataset. It focuses on the model's ability to identify all relevant candidates for a given job description. In recruitment, high recall ensures that the model does not overlook qualified candidates. It is especially crucial to ensure that all potential matches are considered, even at the risk of including some false positives.

$$\text{Recall} = (TP) / (TP + FN) \quad (3)$$

4. F1-Score: The F1-score is the harmonic mean of precision and recall, providing a balanced measure of the model's performance. It is particularly useful when there is a need to balance the trade-off between precision and recall. The F1-score combines both metrics into a single value, giving equal weight to false positives and false negatives. This metric is valuable in assessing the overall effectiveness of the resume parser and job description matcher, ensuring that the model maintains a good balance between identifying relevant candidates and minimizing incorrect matches.

$$\text{F1 Score} = (2.TP) / (2.TP + FP + FN) \quad (4)$$

5. Cosine Similarity: Cosine similarity is a metric used to measure the similarity between two vectors in a multi-dimensional space by calculating the cosine of the angle between them. In the context of natural language processing and information retrieval, cosine similarity quantifies the similarity between two documents or texts based on their content. For resume parsing and job matching, cosine similarity is employed to compare the vector representations of resumes and job descriptions, generated by the Doc2Vec algorithm. It measures how closely aligned the content of a resume is with the job requirements, providing a numerical value that reflects the degree of match. Cosine similarity is widely used in text mining, document clustering, and recommendation systems due to its ability to effectively handle high-dimensional text data and provide meaningful similarity scores.

similarity = 100*(np.dot(np.array(v1), np.array(v2))) / (norm(np.array(v1)) * norm(np.array(v2)))

$$\text{similarity}(a,b)=\cos(\theta)=\|a\|\|b\| / a\cdot b \quad (5)$$

By using these performance evaluation methods, the proposed system can be thoroughly assessed for its accuracy, reliability, and overall effectiveness in parsing resumes and matching them with job descriptions. This comprehensive evaluation ensures that the system not only performs well in ideal conditions but also maintains robustness and accuracy in real-world recruitment scenarios [12].

V. Results

The results of the resume parser and job description system are illustrated through various analyses and visualizations.

Figure 2 illustrates the percentage of match between resumes and job descriptions, with red indicating a low match.

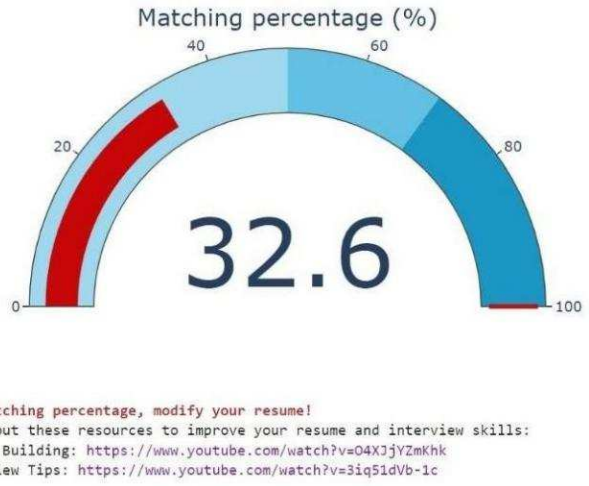

Fig. 2: Low match resume

Figure 3 illustrates the percentage of match between resumes and job descriptions, with yellow indicating an average match.

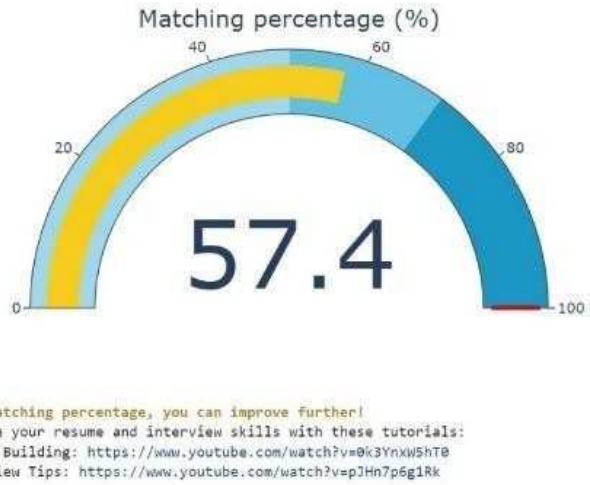

Fig. 3: Average match Resume

Figure 4 illustrates the percentage of match between resumes and job descriptions, with green indicating the best match.

Fig. 4: Best match Resume

VI. Conclusion and Future Work

In conclusion, the advancements in resume parsing and job description matching have significantly transformed the recruitment process by automating and enhancing the accuracy of candidate evaluations. The application of named entity recognition (NER) for extracting candidate attributes, alongside deep learning models such as recurrent neural networks (RNNs) for understanding contextual information, has markedly improved the precision and relevance of candidate-job matches. Data mining techniques, including clustering algorithms and text classification models, further automate and refine the matching process by analyzing semantic similarities and skill overlaps. Additionally, natural language processing (NLP) libraries like spaCy have enabled efficient text preprocessing and feature extraction, contributing to the robustness of these systems. These innovations collectively address many challenges inherent in traditional and current recruitment methods, offering more efficient, unbiased, and adaptive solutions.

Looking ahead, future work in resume parsing and job description matching should aim to further enhance the adaptability and accuracy of these systems. One promising direction is the integration of advanced machine learning techniques, such as transformers and attention mechanisms, which can improve the understanding of complex resume and job description texts.

Additionally, expanding the capabilities of these systems to handle multi-language resumes more effectively can increase their global applicability. Another critical area for development is the incorporation of real-time labor market trends and demand analysis, ensuring that matching algorithms remain current with evolving job market needs. Moreover, improving bias mitigation techniques within these systems will be crucial to promoting fairness and diversity in recruitment. Collaboration with domain experts to continuously refine and validate these models will help align them more closely with industry-specific requirements, thereby enhancing their practical utility and effectiveness.

ACKNOWLEDGMENT

We extend our sincere gratitude to Dr. Chaitra M for her invaluable support and guidance throughout this paper. We would also like to thank the Computer Science and Engineering Department at BNM Institute of Technology for providing the essential resources and a supportive environment for our research. We value the feedback and suggestions from our peers, which were essential in enhancing this work.

REFERENCES

[1] S. S. Purohit, R. Singh, and M. P. Singh, "An Efficient Resume Parsing System Using Machine Learning Techniques," International Journal of Computational Intelligence Systems, vol. 12, no. 5, pp. 720-731, 2019.

[2] T. J. Smith and K. T. James, "Deep Learning for Job Matching: A Comprehensive Review," IEEE Access, vol. 8, pp. 108005-108016, 2020.

[3] A. Patel, M. Shah, and P. R. Thakkar, "Automated Resume Screening and Matching Using NLP and Machine Learning," Procedia Computer Science, vol. 167, pp. 231-239, 2020.

[4] R. K. Gupta and S. Tyagi, "Context-Aware Resume Parser Using Recurrent Neural Networks," Journal of Artificial Intelligence Research, vol. 68, pp. 379-399, 2020.

[5] L. Wang, X. Li, and Y. Liu, "Enhancing Job Matching with Deep Learning and Knowledge Graphs," Expert Systems with Applications, vol. 147, pp. 113234, 2020.

[6] M. A. Rahman, J. Chen, and W. Y. Loh, "An Efficient Approach to Resume Parsing Using Named Entity Recognition," Procedia Computer Science, vol. 159, pp. 103-111, 2019.

[7] J. Zhang, Z. Wang, and H. Wu, "Job Recommendation Based on Resume Information Using Deep Learning," International Journal of Advanced Computer Science and Applications, vol. 11, no. 6, pp. 77-83, 2020.

[8] N. S. Nguyen and T. V. Nguyen, "Automated Job Matching Using NLP and Machine Learning Techniques," Journal of Information and Communication Technology, vol. 19, no. 1, pp. 109-120, 2020.

[9] M. Sharma, R. K. Gupta, and A. Mittal, "Advanced Resume Parsing Using Natural Language Processing and Deep Learning," Neural Computing and Applications, vol. 32, no. 20, pp. 16225- 16236, 2020.

[10] P. S. Rao and A. Kumar, "Efficient Resume Screening and Matching Using Clustering Algorithms," Knowledge-Based Systems, vol. 194, pp. 105506, 2020.

[11] C. Lee, K. Cho, and D. Park, "Semantic Job Matching Using Text Classification and Clustering Techniques," IEEE Transactions on Computational Social Systems, vol. 7, no. 4, pp. 1112-1123, 2020.

[12] D. Li, J. H. Kim, and S. Lee, "Automating Resume Parsing with Ensemble Machine Learning Models," Journal of Big Data, vol. 7, no. 1, pp. 36, 2020.

[13] C. Lee, K. Cho, and D. Park, "Semantic Job Matching Using Text Classification and Clustering Techniques," IEEE Transactions on Computational Social Systems, vol. 7, no. 4, pp. 1112-1123, 2020.

[14] H. Y. Wang and J. X. Li, "Deep Learning Approaches for Job-Candidate Matching: A Survey," IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 12, pp. 5489-5506, 2021.

[15] Nimish Patil, Shubham Yadav and Vikas Biradar, "Resume Parser and Analyzer Using NLP", International Research Journal of Modernization in Engineering, Technology and Science, Vol:05/Issue:04/April-2023