International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 11 Issue: 04 | Apr 2024 www.irjet.net p-ISSN: 2395-0072

Machine learning based Resume Shortlisting and Classification

Einesh Naik[1], Dr. Nilesh B. Fal Dessai[2]

[1]Student, Department of Information Technology and Engineering, Goa College of Engineering, Farmagudi, Goa, India
[2]Head of Department, Department of Information Technology and Engineering, Goa College of Engineering, Farmagudi, Goa, India

---

Abstract - *The recruitment of candidates tailored to specific job profiles is a pivotal task for many companies. With the increasing prevalence of online recruitment, traditional hiring methods are proving to be less efficient. These conventional techniques typically involve a labor-intensive process of manually sifting through submitted applications, reviewing resumes, and creating a shortlist of potential candidates for interviews. In the current technological era, job searching has evolved to be more intelligent and accessible. Companies receive a vast number of resumes/CVs, which are not always well-organized. While significant progress has been made in optimizing the job search process, the selection of candidates based on their resume remains a largely manual task. This research presents a survey of methods that could be used for resume shortlisting and classification according to the company's job description.*

Key Words: **Candidate Classifying, Candidate Shortlisting, KNN, Machine Learning, Cosine Similarity; Random Forest; SVM**

1.INTRODUCTION

The conventional job trend, wherein individuals typically settled for one or two positions throughout their entire professional journey, was once prevalent. Employers took pride in having employees committed to their organization for extended periods, often spanning two or three decades.

However, in the contemporary era, marked by swift technological changes, such scenarios are no longer applicable for most employees and employers alike.

In the digital age, where job opportunities abound and applicants are plentiful, the process of resume shortlisting and classification poses a significant challenge for organizations. Manual screening of resumes is not only labor-intensive but also prone to human biases and inconsistencies. However, with the advent of machine learning (ML) techniques, there's a newfound opportunity to revolutionize this aspect of talent acquisition.

Machine learning offers the promise of automating and optimizing the initial stages of the recruitment process by leveraging algorithms and models to analyze resumes and categorize them based on predefined criteria. By harnessing the power of data-driven insights, organizations can streamline their recruitment efforts, identify top candidates more efficiently, and ultimately improve hiring outcomes.

The traditional approach to resume screening involves manual review by human recruiters, a process that is not only time consuming but also susceptible to bias. Human biases, whether conscious or unconscious, can influence decisions and inadvertently exclude qualified candidates based on factors such as gender, ethnicity, or educational background.

Machine learning presents a transformative solution to these challenges. By training algorithms on historical data of successful hires, organizations can develop models that learn to identify patterns and characteristics indicative of a good fit for a given role. These models can then be deployed to automate the initial screening process, reducing the burden on human recruiters and mitigating bias.

1.1 Objectives

Shortlisting and Ranking of Resume: Evaluate and rank resumes based on their relevance to the job description provided by the company.

Segmentation of Resumes: Categorizing resume into different segments or classes based on their suitability for the given job profile. Clustering algorithms or classification models are implemented to achieve this segmentation, helping recruiters quickly identify potential candidates.

Filtering of Resumes: Filtering of Resumes according to location and percentage or any other criteria.

2. Related Work

2Q-Learning scheme for Resume [1], In this research 2Q-learning framework is proposed. 2Q-learning framework selects the resumes based on the given set of skills specified in the job description. The 2Q-Learning approach involves the selection or rejection of resumes by comparing the resume with the skills or requirements mentioned in the job description by recruiter. In this research NLP model is used for extracting the features from resume. Then extracted skillset from the resume is given to the 2Q-learning agent that matches the skillset and job description and takes the action. The dataset required for training was obtained from

Kaggle it contained 2400+ resumes with 14 categories. AI-Powered Resume job matching: A document ranking approach using deep neural networks [2], LSTM component processes the embeddings of each input sequence, capturing contextual information and sequential dependencies. CNN component applies convolutional filters to the embeddings, extracting local patterns and features. Then the output of LSTM and CNN was concatenated to combine their respective representation. Next, multi-head attention mechanism was applied to the concatenated representation. This layer, allowed the model to focus on relevant information. Finally, a similarity score is generated using the cosine measure, which serves as an indicator of the degree of similarity or dissimilarity between the Resume and the job description. An Efficient algorithm for Ranking candidates in e-recruitment system [3], This research proposed an algorithm Slashrank for ranking candidates based on job posts. To rank the candidate score is generated by comparing there feature value to JD. Priority to feature is given so that candidates can be evaluated based on some important features like education. Automated Resume Evaluation system using NLP [4], The proposed architecture in this research first Converts the unstructured resumes in structured data using NLP. Extraction phase where relevant information is extracted from the resume and giving them identifier values. candidate's information like personal, education, experience, technical skills, internship(if any), hobbies, etc. The next module is the filtration module, which refines the lists by removing the insignificant terms that do not contribute to the matching process. It takes the list of skills from both the resume and job posts to construct the semantic network. It takes the semantic network as input and produces the measures of closeness between them as output. Based on the values assigned, the resumes are ranked accordingly in the final segment. To make the filtration process more efficient, a score is given to each resume to rank the applicant.

3. Methods used for Resume Shortlisting and Classification of Resume

3.1 Manual Shortlisting and Classification of Resumes

Manual shortlisting and classification of resumes have long been fundamental practices employed by organizations to identify the most suitable candidates for job openings. This process, although traditional, remains a cornerstone of recruitment strategies, offering insights into candidate qualifications, skills, and experiences that are crucial for making informed hiring decisions.

Manual shortlisting and classification involve human recruiters or hiring managers reviewing and evaluating resumes submitted by job applicants. This process typically follows a set of predefined criteria aligned with the requirements of the job role, such as educational qualifications, relevant experience, skills, certifications, and achievements. Recruiters meticulously assess each resume, often employing a combination of intuition, expertise, and organizational guidelines to determine the suitability of candidates.

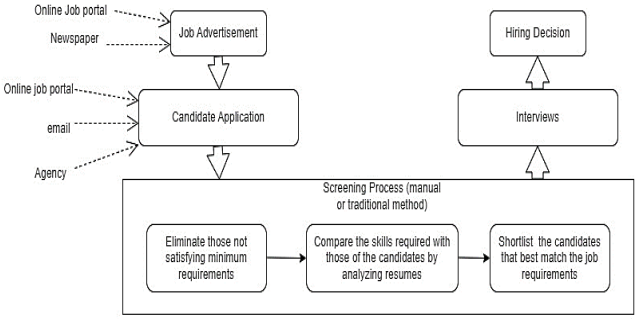

Fig -1: Shortlisting of Resumes using Manual or Traditional method

3.2 Challenges involved in Manual or Traditional Method

While manual shortlisting and classification offer valuable insights into candidate suitability, they are not without challenges:

Time-intensive process: Manual resume screening is labor-intensive and time-consuming, especially for organizations receiving a high volume of applications. Recruiters must dedicate significant resources to review each resume thoroughly, leading to prolonged time-to-hire and potential delays in candidate engagement.

Subjectivity and Bias: Human judgment introduces subjectivity and potential bias into the screening process. Recruiters may inadvertently favor certain candidates based on personal preferences, stereotypes, or unconscious biases, leading to disparities in candidate selection and representation.

Scalability and Consistency: As organizational needs evolve and recruitment volumes fluctuate, manual processes may struggle to scale effectively. Maintaining consistency and standardization across multiple recruiters and job openings can be challenging, resulting in variability in screening criteria and outcomes.

However, manual screening is not without its limitations, including time-intensive nature, susceptibility to bias, and challenges in scalability and consistency. As organizations strive to optimize their recruitment processes, there is a growing interest in complementing manual practices with technology-driven solutions such as machine learning algorithms for automated resume screening. By integrating theoretical insights with practical considerations, organizations can enhance their recruitment strategies and

improve hiring outcomes while mitigating the inherent challenges of manual resume screening.

3.3 Machine Learning based Shortlisting and Classification of Resumes

Use of Machine Learning in Resume shortlisting and classification can be broken down into 2 components. First component Shortlisting of Resumes, it can be done using machine learning techniques such as similarity matrix example Cosine Similarity. Similarity Matrix finds the similarity between resume and job description, this similarity is measured in values. Higher the value, more similar is the resume to job description. Similarity value of resumes can be arranged in descending order to shortlist resumes according to number of available positions. Second component that is Classification of resumes can be achieved through various classification models like KNN, SVM, Random Forest.

1.Steps involved in Pre-Processing

Dropping of unwanted Features: Removal of less important features such as emails, phone, etc. Machine learning libraries such as pandas could be used to remove unwanted entries.

Handling Null entries: Handling Null entries is important, so that model can be trained efficiently. Default values could be used to handle null entries.

Lemmatization: Lemmatization is a natural language processing (NLP) technique used to reduce words to their base or root form, known as the lemma. The lemma represents the canonical, dictionary form of a word, which allows for better analysis and understanding of text data.

Unlike stemming, which simply chops off prefixes or suffixes from words to derive their root form (sometimes resulting in non-real words), lemmatization considers the morphological analysis of words and aims to return a real word that belongs to the language.

Working of Lemmatization:

Tokenization: The text is first divided into individual words or tokens.

Part-of-Speech Tagging (POS): Each word is tagged with its part of speech (e.g., noun, verb, adjective).

Lemmatization is Based on the part-of-speech information, each word is mapped to its lemma using a vocabulary and linguistic rules. For example: "Running" (verb) → "run", "Cats" (noun) → "cat", "Better" (adjective) → "good"

Lemmatization is beneficial in NLP tasks where the semantic meaning of words is important, such as text classification, sentiment analysis, and machine translation. It helps reduce the dimensionality of the feature space and ensures that different inflected forms of words are treated as the same token, thereby improving the accuracy and interpretability of NLP models.

Popular NLP libraries such as NLTK (Natural Language Toolkit), spaCy, and CoreNLP provide lemmatization functionality, making it easy to incorporate into NLP pipelines for various applications.

Vectorization: Vectorization is the process of converting textual data into numerical vectors or arrays that can be understood and processed by machine learning algorithms. This transformation allows NLP models to work with text data effectively, as most machine learning algorithms require numerical inputs. There are several techniques for vectorizing text data:

Bag-of-Words (BoW): BoW represents each document as a vector where each dimension corresponds to a unique word in the vocabulary. The value of each dimension indicates the frequency of the corresponding word in the document. BoW disregards the order of words and only considers their presence or absence in the document.

Term Frequency-Inverse Document Frequency (TF-IDF): TF-IDF is similar to BoW but also takes into account the importance of words in the context of the entire corpus. It computes a weight for each word based on its frequency in the document (TF) and rarity across all documents in the corpus (IDF).

Term Frequency (TF):

$$\mathrm{TF}(t, \text{text}) = \frac{\text{Number of times term } t \text{ appears in text}}{\text{Total number of terms in the text}}$$

Inverse Document Frequency (IDF):

$$\mathrm{IDF}(t, \text{corpus}) = \log\left(\frac{\text{Total number of texts in corpus } |\text{corpus}|}{\text{Number of texts containing term } t}\right)$$

TF-IDF:

$$\text{TF-IDF}(t, \text{text}, \text{corpus}) = \mathrm{TF}(t, \text{text}) \times \mathrm{IDF}(t, \text{corpus})$$

2. First Component Shortlisting of Resumes

This can be done using Similarity Matrix. A similarity matrix is a square matrix that quantifies the similarity between pairs of objects in a dataset. Each element of the matrix represents the degree of similarity between two corresponding objects, typically ranging from 0 (no similarity) to 1 (perfect similarity). Similarity matrices are commonly used in various fields, including natural language processing, image analysis, and clustering.

In the context of Candidate shortlisting, a similarity matrix can be constructed to measure the similarity between Resume and Job Description. Different similarity metrics can

be used depending on the specific task and the characteristics of the data. Some commonly used similarity measures in NLP include:

Cosine Similarity: Measures the cosine of the angle between two vectors in a vector space. It is often used to compare the similarity between document vectors or word embeddings.

$$\cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \sqrt{\sum_{i=1}^{n} B_i^2}}$$

Jaccard Similarity: Calculates the intersection over the union of two sets. It is frequently used to compare the similarity between sets of words or documents.

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Euclidean Distance: Measures the straight-line distance between two points in a multidimensional space. It is often used in vector-based representations to measure dissimilarity.

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2}$$

3. Second Component Classification of Resume using Machine Learning Models

There are various machine learning models for classification. In this research we have implemented three classification models which are KNN, SVM, Random Forest.

KNN - The K Nearest Neighbor (KNN) algorithm is a well-established method in AI that serves for both classification and regression tasks [14]. Unlike many other algorithms, KNN doesn't build an explicit model from the training data; instead, it memorizes the training instances to make predictions on new data points. KNN operates by identifying the k nearest data points to a query point based on a distance metric such as Euclidean or Manhattan distance. It then assigns the label or value of the majority of those k nearest neighbors to the query point. Being a non-parametric algorithm, KNN makes no assumptions about the distribution of the data and can perform effectively even on small datasets. However, its performance may be limited by the curse of dimensionality, where the distance between data points becomes less meaningful as the number of dimensions increases.

Support Vector Machines (SVM) are powerful supervised learning models used for classification and regression tasks. SVMs are particularly effective for classification problems in which the data can be separated into distinct classes by finding the hyperplane that maximizes the margin between them. In SVM, the margin is the distance between the hyperplane and the nearest data points from each class. The goal is to find the hyperplane that maximizes this margin, as it provides the greatest separation between the classes and is less likely to overfit the data. Support vectors are the data points closest to the hyperplane and are crucial for defining its position. These points determine the decision boundary and are used to maximize the margin.

Random Forest is an ensemble learning method that constructs a multitude of decision trees during training and outputs the mode (for classification tasks) or mean (for regression tasks) prediction of the individual trees. Each decision tree is built using a random subset of the training data, with replacement, and a random subset of features at each split. This randomness introduces diversity among the trees, reducing overfitting and improving generalization. During prediction, each tree in the ensemble independently classifies or predicts the target variable for a given input, and the final output is determined by aggregating the individual tree predictions through majority voting or averaging. Random Forest is known for its robustness, scalability, and ability to handle high-dimensional data, making it a popular choice for a wide range of classification and regression tasks.

The dataset used in this research can be obtained from kaggle. The Dataset contains resumes which are relevant to IT industry. The Resume Dataset contains collection of resumes, it has 963 resumes. It has 25 categories which are, Java Developer, Testing, DevOps Engineer, Python Developer, Web Designing, HR, Hadoop, Blockchain, ETL Developer, Operations Manager, Data Science, Sales, Mechanical Engineer, Arts, Database, Electrical Engineering, Health and fitness, PMO, Business Analyst, DotNet Developer, Automation Testing, Network Security Engineer, SAP Developer, Civil Engineer, Advocate.

TABLE -1: DIFFERENT CATEGORIES AND COUNT OF RESUMES

Category	No. of Resumes
Java Developer	84
Testing	70
DevOps Engineer	55
Python Developer	48
Web Designing	45
HR	44
Hadoop	42
Blockchain	40
ETL Developer	40
Operations Manager	40
Data Science	40
Sales	40
Mechanical Engineer	40
Arts	36
Database	33
Electrical Engineering	30
Health and fitness	30
PMO	30
Business Analyst	28
DotNet Developer	28
Automation Testing	26
Network Security Engineer	25
SAP Developer	24
Civil Engineer	24
Advocate	20

TABLE -2: COMPARISON OF DIFFERENT MODELS

Model	Accuracy
KNN	98.44
SVM	99.48
Random Forest	99.42

4. Shortlisting of Resumes using Existing Machine Learning Algorithms

The 2Q Learning Framework [1], involves the evaluation of resumes against a predetermined set of skills outlined in the job description. This approach facilitates the selection or rejection of resumes by comparing their contents with the specified skills or requirements provided by the recruiter. Initially, the resume analyzer retrieves a resume from the database and extracts the relevant skill set from it. Subsequently, the 2Q Learning agent makes a determination regarding the suitability of the resume based on the skills outlined in the job description. Resumes that meet the criteria are advanced to the interview stage, while those that do not are subjected to further review by the resume selector, which proceeds to evaluate the next resume provided by the resume analyzer.

AI-Powered Resume job matching [2], LSTM component captures contextual information and sequential dependencies. CNN component applies convolutional filters to the embeddings, extracting local patterns and features. Then the output of LSTM and CNN was concatenated to combine their respective representation. Next, multi-head attention mechanism was applied to the concatenated representation. This layer, allowed the model to focus on relevant information. Finally, a similarity score is generated using the cosine measure, which finds the degree of similarity or dissimilarity between the Resume and the job description.

Slashrank algorithm [3], It is an efficient algorithm which helps to evaluate resumes based on important features. First we assign priority to features. Then we set Threshold for feature priority and candidates feature score. If feature priority is greater than Threshold of feature priority, then for each candidate we calculate candidate score, else next feature.

TABLE -3: COMPARISON OF EXISTING ALGORITHMS

Algorithm/ Method	Dataset	Accuracy
AI-Powered Resume job matching (LSTM+CNN+sBERT)	Canadian company dataset	84%
Slashrank	1.Real world dataset(96 records), 2.DBLP dataset, 3.Synthetic dataset(10000 records)	87 % average accuracy, tested on real world dataset (96 records)
2Q-learning	Dataset obtained from Kaggle(2400 records)	95.80%

4. Results

We computed accuracy of KNN, SVM, Random Forest as shown in Table II. Among these three models, the best results were obtained using SVM. As evident in Table II, SVM model outperforms the other three machine learning models in terms of performance.

Also 10 Resumes were manually created to test ranking of resumes using Cosine similarity. Cosine similarity found the similarity between Resume and job description and assigned a similarity value. Then resumes were sorted in descending order of similarity value. Cosine Similarity effectively ranked the resumes, which help in shortling of resumes.

5. Conclusion

Shortlisting, Ranking and classifying a resume in Human Resource department is usually a critical problem. In this paper, different methods based on machine learning techniques are implemented and compared. Manual or Traditional methods are acceptable, but they are time consuming, susceptible to human bias. Machine learning based Shortlisting, Ranking, and classification of resume algorithms have overcome such issues and achieved significant performances in both accuracy and speed.

For future work, Deep Learning based models for Shortlisting of Resumes could be used, such as Semantic Similarity model. Semantic Similarity is the task of determining how similar two sentences are, in terms of what they mean. Use of semantic similarity models such as BERT. Semantic models can be used for finding similarity between Resume and Job Description. Some widely used Semantic models are all-mpnet-base-v2, Praphrase-MiniLM-L6-v2, all-roberta-large-v1.

all-mpnet-base-v2:This is a sentence transformers model: It maps sentences & paragraphs to a 768 dimensional dense vector space and can be used for tasks like clustering or semantic search.

Praphrase-MiniLM-L6-v2: sentence transformers model: It maps sentences & paragraphs to a 384 dimensional dense vector space and can be used for tasks like clustering or semantic search.

all-roberta-large-v1: It maps sentences & paragraphs to a 1024 dimensional dense vector space and can be used for tasks like clustering or semantic search.

REFERENCES

[1] Bhoomika SP, Likhitha S, Chandana H S, Kavya S A, Bhargavi K, 2Q-Learning scheme for Resume, 4th International Conference for Emerging Technology (INCET) IEEE 2023.

[2] Sima Rezaeipourfarsangi, Evangelos E. Milios, AI-Powered Resume job matching: A document ranking approach using deep neural networks, ACM Symposium on Document Engineering 2023.

[3] Abdul Hanan Minhas, Mohammad Daniyal Shaiq, Saad Ali Qureshi, Musa Dildar Ahmed Cheema, Shujaat Hussain, Kifayat Ullah Khan, An Efficient algorithm for Ranking candidates in e-recruitment system, 16th International Conference on Ubiquitous Information Management and Communication IEEE 2022.

[4] Rohini Nimbekar, Yogesh Patil, Rahul Prabhu, Shainila Mulla, Automated Resume Evaluation system using NLP, International Conference on Advance in Computing, Communication and Control IEEE 2019.

[5] Saswat Mohanty, Anshuman Behera, Sushruta Mishra, Ahmed Alkhayyat, Deepak Gupta, Resumate: A Prototype to Enhance Recruitment Process with NLP based Resume Parsing, 4th International Conference on Intelligent Engineering and Management (ICIEM) IEEE 2023.

[6] M.F. Mridha, Rabeya Basri, Rabeya Basri, Md. Abdul Hamid, A Machine Learning Approach for Screening Individual's Job Profile Using Convolutional Neural Network, International Conference on Science & Contemporary Technologies (ICSCT) IEEE 2021.

[7] Tumula Mani Harsha, Gangaraju Sai Moukthika, Dudipalli Siva Sai, Mannuru Naga Rajeswari Pravallika, Satish Anamalamudi, MuraliKrishna Enduri, Automated Resume Screener using Natural Language Processing(NLP), 6th International Conference on Trends in Electronics and Informatics (ICOEI) IEEE 2022.

[8] Vishnu S. Pendyala, Nishtha Atrey, Tina Aggarwal, Saumya Goyal, Enhanced Algorithmic Job Matching based on a Comprehensive Candidate Profile using NLP and Machine Learning, Eighth International Conference on Big Data Computing Service and Applications (BigDataService) IEEE 2022.

[9] Muntaha Mehboob, M.Saad Ali, Saif ul Islam, Saif ul Islam, Evaluating Automatic CV Shortlisting Tool For Job Recruitment Based On Machine Learning Techniques, Mohammad Ali Jinnah University International Conference on Computing (MAJICC) IEEE 2022.

[10] Chamila Maddumage, Dulanjaya Senevirathne, Isuru Gayashan, Tharusha Shehan, Sagara Sumathipala, Intelligent Recruitment System, 5th International Conference for Convergence in Technology (I2CT) IEEE 2019.

[11] Thapanee Boonchob, Nuengwong Tuaycharoen, Santisook Limpeeticharoenchot, Narongthat Thanyawet, Job-Candidate Classifying and Ranking System Based Machine Learning Method, 26th International Computer Science and Engineering Conference (ICSEC) IEEE 2022.

[12] Pratibha Swami, Vibha Pratap, Resume Classifier and Summarizer, International Conference on Machine Learning, Big Data, Cloud and Parallel Computing (COM-IT-CON) IEEE 2022.

[13] S Bharadwaj, Rudra Varun, Potukuchi Sreeram Aditya, Macherla Nikhil, G.Charles Babu, Resume Screening using NLP and LSTM, International Conference on Inventive Computation Technologies (ICICT) IEEE 2022.

[14] M.Alamelu, D.Sathish Kumar, R.Sanjana, J.Subha Sree, A.Sangeerani Devi, D.Kavitha, Resume Validation and Filtration using Natural Language Processing, 10th International Conference on Internet of Everything, Microwave Engineering, Communication and Networks (IEMECON) IEEE 2021.

[15] Rishabh Bathija, Vanshika Bajaj, Chandni Megnani, Jasmine Sawara, Prof. Sanjay Mirchandani, Revolutionizing Recruitment: A Comparative Study Of KNN, Weighted KNN, and SVM - KNN for Resume Screening, 8th International Conference on Communication and Electronics Systems (ICCES) IEEE 2023.

[16] Rasika Ransing, Akshaya Mohan, Nikita Bhrugumaharshi Emberi, Kailas Mahavarkar, Screening and Ranking Resumes using Stacked Model, 5th International Conference on Electrical, Electronics, Communication, Computer Technologies and Optimization Techniques (ICEECCOT) IEEE 2021.

[17] Ashif Mohamed, Wickram Bagawathinathan, Usama Iqbal, Shahik Shamrath, Anuradha Jayakody, Smart Talents Recruiter – Resume Ranking and Recommendation System, IEEE 2018.

[18] Sujit Amin, Nikita Jayakar, Sonia Sunny, Pheba Babu, M.Kiruthika, Ambarish Gurjar, Web Application for Screening Resume, International Conference on Nascent Technologies in Engineering (ICNTE 2019) IEEE 2019.