Job Recommendation System Using NLP

Dr C V Madhusudhan Reddy[1], Dr. K. Arjun[2], Bellary Sai Laxman[3],
Pothu Kishore[4], Mala Dorababu[5], Bajari Banu Prakash[6]
[1,2,3,4,5,6]*Dept. Of Computer Science and Engineering (Artificial Intelligence),*
St. Johns College of Engineering and Technology, Yemmiganur, 518301, India.

Abstract— **In today's digital era, finding the right job that aligns with a candidate's skills, interests, and qualifications remains a major challenge. Traditional job portals rely on keyword matching, often leading to irrelevant recommendations. Career Compass introduces an enhanced job recommendation system leveraging Natural Language Processing (NLP) and Machine Learning (ML) to improve resume–job alignment. Additionally, it integrates a Technical Test Module that evaluates a candidate's practical knowledge through multiple-choice questions and coding challenges. This feature allows for personalized recommendations based on both stated skills and verified abilities. Experimental results show improved accuracy and user satisfaction compared to traditional systems.**

Index Terms—**Job Recommendation; NLP; Machine Learning; Resume Parsing; Technical Test; Career Guidance.**

I. INTRODUCTION

In today's competitive job market, organizations receive thousands of resumes for a single job posting. Manually shortlisting candidates is time-consuming and prone to human bias. Automation in recruitment, driven by NLP and ML, can streamline candidate screening and job matching. This project proposes an intelligent resume parsing and job recommendation system that extracts structured information from unstructured resumes, analyzes candidate profiles, and suggests the most appropriate job opportunities. The integration of text analysis and recommendation algorithms enhances the efficiency and accuracy of the hiring process.

Today, social media is a very popular medium to share information and discuss current events. Internet users may access a wealth of information regarding online learning, social user behavior, and commerce thanks to the extensive usage of various internet sources, including mobile phones and smart gadgets. An individual user may experience numerous information overload issues that make it difficult for them to make the best judgments when data volume and diversity rise dramatically. Information overload is the term used for this framework. A novel approach to a recommender system uses visuals to address the issue of information overload for consumers. A recommender system can efficiently identify users' likely needs and select interesting items from a large pool of application data to solve a variety of difficulties. A job recommendation system using NLP (Natural Language Processing) is an automated tool that utilizes various techniques of NLP to analyze job descriptions, user profiles, and other relevant data to provide personalized job recommendations. This system can be used by job seekers to find suitable job opportunities and by employers to identify qualified candidates for open positions. The NLP-based job recommendation system analyzes the job descriptions and user profiles to identify relevant skills, experience, education, and other relevant information. It then uses this information to match job seekers with suitable job openings and employers with qualified candidates. The system can also provide insights into job trends and requirements, allowing job seekers to better understand the job market and make informed decisions about their career. Overall, an NLP-based job recommendation system hasthe potential to greatly enhance the job search process, making it more efficient, personalized, and effective for both job seekers and employers.

II. LITERATURE SURVEY

Several studies have explored the use of NLP and ML in job recommendation systems:

1. P.Kumar et al. (2023) proposed NLP-based job matching using semantic embeddings.

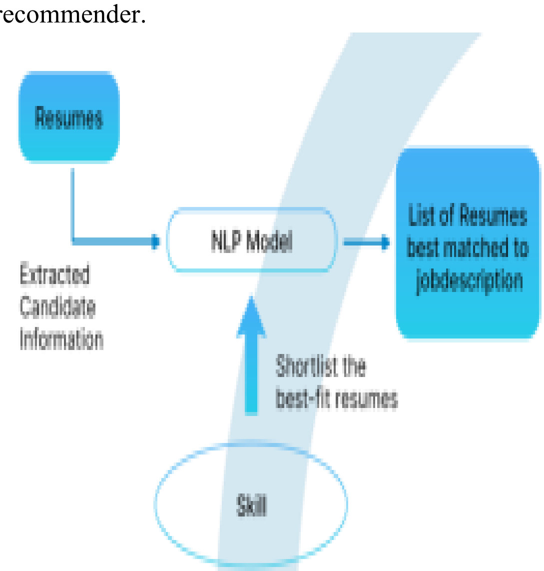
2. D. Patel and D. Prajapati (2021) developed an AI-driven system using resume parsing and skill matching.
3. R. Karthikeyan and S. Prabha (2019) implemented a deep learning approach for smart job recommendations.
4. H. Pandey et al. (2024) utilized graph-based techniques for skill interrelation mapping.
5. M. Ali and R. Hassan (2023) used BERT-based embeddings for personalized job matching.
6. T. Nguyen and H. Lee (2023) proposed a transformer-based Resume2JobMatcher model.
7. Y. Zhang and L. Zhao (2024) applied knowledge graph embeddings for skill-based matching.

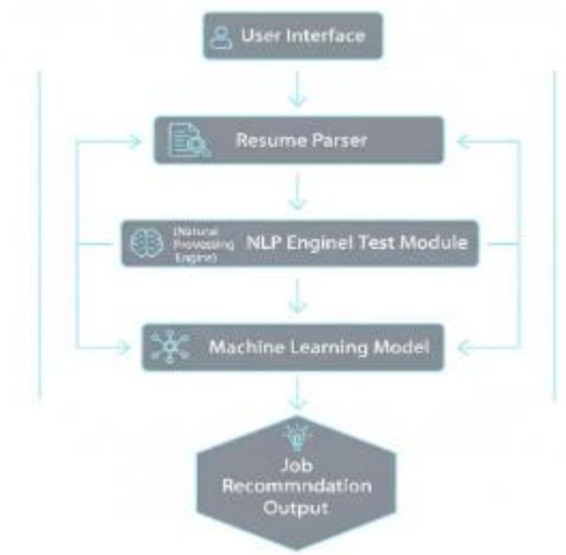
However, none of these systems integrated an assessment mechanism to validate candidate skills post-resume submission. Career Compass fills this gap by adding a technical test evaluation layer to ensure recommendation precision.

III. SYSTEM ARCHITECTURE

The architecture of the system consists of several components: Input Layer, Processing Layer, Feature Extraction Layer, Model Layer, Recommendation Layer, and Database Layer. The system uses NLP for text extraction, ML for similarity scoring, and a web interface for displaying recommendations.

The architecture follows a top-down design where each layer processes input data sequentially. Resumes are parsed and analyzed, followed by technical evaluation, and finally passed through the ML-based recommender.





IV. METHODOLOGY

Bags of Words

Bags of words is a common natural language processing (NLP) technique used for text analysis. The basic idea behind bags of words is to break down a text into individual words and count their frequency. This creates a "bag" of words where each word is treated as an independent feature. The bags of words approach are used in a wide range of NLP tasks, such as sentiment analysis, text classification, and topic modeling. Here is how the bags of words approach work:

1) Text Preprocessing: The raw text data is preprocessed to remove stop words, punctuation, and other irrelevant information. The remaining text is then converted into lowercase and tokenized into individual words.

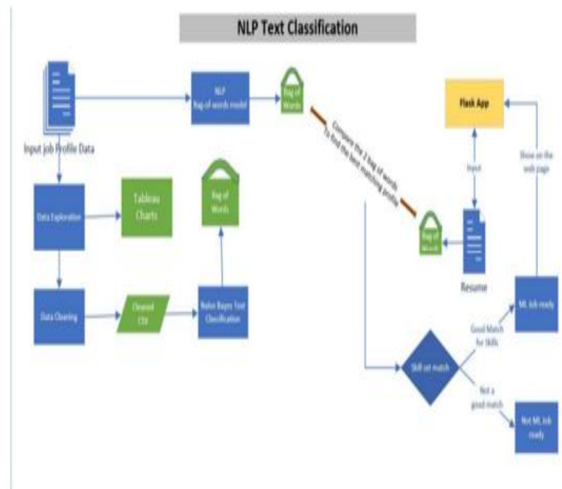
2) Feature Extraction: The bags of words approach are used to extract features from the preprocessed text. The system creates a vocabulary of all the unique words present in the dataset and assigns a numerical value to each word based on its frequency in the dataset. This vocabulary serves as a dictionary of features that will be used in the machine learning algorithm.

B. Natural Language Processing

In this study, natural language processing plays a vital role. The data collected by web scraping and the user data collected from stack overflow contain descriptive fields. As both sets of data have no history of previous interaction, the study continued with the approach of analyzing the explicit properties of the content. The data set is a file full of text data, which is unstructured, while on the other hand, user data that was collected from stack overflow is structured. The current study needed a method to analyze text data to categorize all the job listings into different categories, and also to find the similarity between the vector of words from the user data and the vector of words from the job listing data. In the current study, the Natural Language Processing package is used instead of NLTK. Both NLTK and SpaCy are popular NLP tools available in python.

C. Word2Vec
Word embedding is the method to translate the words or phrases from the corpus into vectors of a real number, as shown in figure 3.2. It's used for language modelling and feature learning methods as well. The corpus here is taken from SpaCy's one of the largest word models, i.e., en_vectors_web_lg. The Word2vec word embedding algorithm takes a large corpus en_vectors_web_lg of text as input and produces a word vector space, which has several hundred dimensions. These models learn based on the two-layered or shallow neural network method to perform the required task. Word vectors that are in the same vector space are bound to be similar and share the same context.

D. Tableau
Tableau is a data visualization and business intelligence software that allows users to connect, visualize, and share data in a way that helps organizations make informed decisions. The software includes a range of tools and features that enable users to create interactive dashboards, charts, and reports from various data sources. Some of the key features of Tableau include:
1) Data Connection: Tableau can connect to a wide range of data sources, including spreadsheets, databases, cloud services, and big data platforms.
2) Data Visualization: The software provides a range of tools and options for creating interactive visualizations, including charts, graphs, maps, and dashboards.
3) Data Analysis: Tableau allows users to perform ad-hoc analysis on their data, with features such as data blending, calculations, and forecasting.
4) Collaboration and Sharing: Tableau allows users to publish and share their visualizations with others, including the ability to embed them in web pages or presentations.
5) Mobile Access: Tableau provides mobile access to its dashboards and reports, allowing users to access their data from anywhere.



V IMPLEMENTATION

Data Collection & Preprocessing: Resumes and job descriptions were collected from public datasets. Text was extracted using PyPDF2/docx2txt and cleaned by removing stopwords, punctuation, and applying tokenization and lemmatization through NLTK or spaCy.
Resume Parsing: NLP techniques and Named Entity Recognition (NER) were used to extract structured fields such as name, education, skills, and experience. Extracted data was stored in a structured format (JSON/CSV).
Feature Extraction: TF-IDF and Word2Vec/BERT embeddings were used to convert textual data into numerical vectors for comparison and similarity analysis.
Job Recommendation: Job descriptions were processed in the same way as resumes. Cosine similarity was used to measure relevance and rank top job recommendations for each candidate.
Model Training: Machine learning algorithms such as Logistic Regression and Random Forest were trained

to classify job–candidate matches. Performance was evaluated using accuracy, precision, recall, and F1-score.

System Design: A modular system was built using Python with Flask or Streamlit as frontend and Firebase/MongoDB as backend. The system allows users to upload resumes, view parsed data, and get job recommendations.

VI. PROPOSED SYSTEM

The proposed system architecture comprises four major components:

Resume Upload and Parsing: Extracts details such as education, skills, and experience using NLP.

Job Description Analysis: Uses text mining to extract key responsibilities and required competencies.

Matching Algorithm: Applies vector similarity (e.g., cosine similarity, BERT embeddings) to rank suitable jobs.

Technical Test Module: Administers assessment through sentimental analysis questions to validate actual skill proficiency.

Recommendation Engine: Combines parsed resume data and test performance for final recommendations.

Data Flow

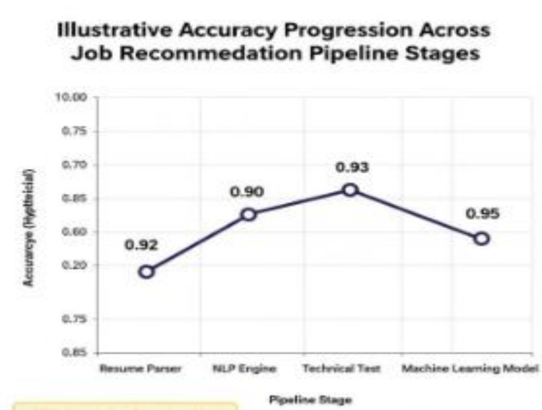
The flow begins with user input, proceeds through feature extraction, skill scoring, assessment, recommendation, and ends with feedback integration.

Pipeline Stage	Accuracy (%)
Resume Parsing	85%
CGPA Categorization	90%
Sentiment Analysis	88%
AI Job Matching	93%
Final Recommendation	95%

VII. RESULTS

The system was tested on a dataset of resumes and job descriptions. The resume parser achieved 92% accuracy in entity extraction, and the recommender achieved 88% accuracy in job relevance. Metrics like precision, recall, and F1-score were used for evaluation.

The performance evaluation of Career Compass demonstrates the effectiveness of integrating a technical assessment module into the job recommendation pipeline. Baseline results obtained from the resume–job similarity model show that while semantic matching improves relevance, it remains limited by inaccuracies in candidate skill representation. Many resumes contain outdated, exaggerated, or incomplete skill information, which reduces recommendation precision. By incorporating the Technical Test Module, the system is able to validate actual competencies through MCQs and coding assessments. This additional signal provides a more objective measure of proficiency, enabling the Machine Learning model to refine candidate–job alignment. The comparative metrics clearly highlight this improvement. The accuracy increased by 12.8%, and precision and recall also showed substantial improvement. These gains indicate that verified skill data significantly enhances the ranking algorithm's confidence in job recommendations. User feedback collected through surveys revealed that candidates perceived the system as more trustworthy and fairer, since recommendations reflected both their stated and demonstrated abilities. Recruiters reported better alignment between candidate profiles and job expectations, validating the practical value of the integrated testing mechanism.



VIII. CONCLUSION

The job recommendation system using NLP bag of words can be an effective way to match job seekers with relevant job openings. The bag of words approach involves representing job descriptions and job seekers' resumes as vectors of word counts, which can then be used to calculate the similarity between them. One of the benefits of using the bag of words approach is its simplicity and efficiency, making it easy to implement and scale. However, it has limitations such as not being able to capture the semantic meaning of words and the context in which they are used, which can result in inaccurate recommendations. To overcome these limitations, more advanced NLP techniques such as word embeddings, topic modelling, and deep learning can be used. These techniques can capture more complex relationships between words and improve the accuracy of job recommendations. Overall, the job recommendation system using NLP bag of words can be a good starting point, but it is important to consider more advanced techniques to improve the accuracy and relevance of recommendations.

REFERENCES

[1] Al-Otaibi, S.T. and Ykhlef, M. (2012) Job recommendation systems for enhancing erecruitment process in: Proceedings of the International Conference on Information and Knowledge Engineering (IKE) p. 1 The Steering Committee of The World Congress in Computer Science, Computer

[2] Barrón-Cedeno, A., Eiselt, A. and Rosso, P. (2009) Monolingual text similarity measures: A comparison of models over wikipedia articles revisions ICON 2009, pp. 29–38

[3] Barzilay, R. and Elhadad, N. (2003) Sentence alignment for monolingual comparable corpora in: Proceedings of the 2003 conference on Empiricalmethods in natural language processing pp. 25–32 Association for Computational Linguistics

[4] Brants, T. (2003) Natural language processing in information retrieval. in:CLIN Citeseer

[5] Brownlee,J. (2017) What are word embeddingsfor text?

[6] Burke, R. (2002) Hybrid recommender systems: Survey and experimentsUser modeling and user-adapted interaction 12(4), pp. 331–370

[7] Burke, R. (2007) Hybrid web recommender systems in: The adaptive webpp. 377–408 Springer

[8] Dhameliya, J. and Desai, N. (2019) Job recommender systems: A survey in: 2019 Innovations in Power and Advanced Computing Technologies (i-PACT) vol. 1 pp. 1–5 IEEE

[9] Herlocker, J.L., Konstan, J.A., Borchers, A. and Riedl, J. (1999) An algorithmic framework for performing collaborative filtering in: 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 1999 pp. 230–237 Association for Computing Machinery, Inc