International Journal of Innovative Research in Engineering and Management (IJIREM)
ISSN (Online): 2350-0557, Volume 13, Issue 2, April 2026
DOI: https://doi.org/10.55524/ijirem.2026.13.2.6
Article ID IJIRE-1438, Pages 38-46
www.ijirem.org

An Intelligent Web Platform for Job Search and Skill Matching Using Smart Filtering Techniques

Sanabam Bineshwor Singh[1], T. Abhishek[2], T. Manikanta[3], K. Hima Bindhu[4], and Vishal Dev Kumar[5]

[1, 2, 3, 4, 5] Department of Computer Science & Engineering (AI & ML), Malla Reddy University, Hyderabad, India

Correspondence should be addressed to Sanabam Bineshwor Singh; sbsingh@mallareddyuniversity. ac.in

Received: 26 February 2026; Revised: 13 March 2026; Accepted: 27 March 2026

ABSTRACT— Recruitment is now widely conducted using online recruitment platforms. However, most of the online recruitment platforms currently in use only provide simple keyword matching between resumes and job descriptions. This does not capture the true meaning of the skills and experience possessed by the candidates and can often provide inaccurate job recommendations and increased processing for the recruiters. In this paper, an intelligent web-based job search and skill matching system using Natural Language Processing (NLP) techniques for more accurate candidate and job matching is proposed. The proposed system uses the spaCy library for information retrieval from resumes and the Sentence Transformers library for semantic similarity between the candidate and job descriptions. In addition, the DistilBERT model is utilized for the contextual classification of the jobs. The system is built using the Django framework for the backend and React for the frontend. The system implemented a hybrid approach using the TF-IDF model as a fallback option in the event of unavailability of embedding services. The experimental results show that the semantic similarity-based matching provides more accurate job recommendations than the simple keyword matching approach with an accuracy of 87%. The method may find applications in real time job search.

KEYWORDS—Natural Language Processing; Semantic Skill Matching; Resume Parsing; Recruitment Automation; Trans- former Models; Intelligent Job Recommendation System.

I. INTRODUCTION

The entire employment sector has undergone fundamental recruitment transformation because of digital technologies which enable organizations to find talent and job applicants to discover job vacancies. Employers now use online recruitment platforms as their main method to find potential employees. Job portals still use traditional keyword-based matching systems despite the technological advancements which have emerged. The systems use exact keyword matches to evaluate resumes against job descriptions but they fail to consider contextual meaning and semantic relationships between candidate skills and job requirements. The different terminology used by candidates causes qualified candidates to become hidden while recruiters must sift through numerous applications manually which creates operational delays and extends the time needed for recruiting.

The process of manual resume screening requires a lot of time and it suffers from human mistakes. Recruiters often need to review hundreds or even thousands of applications for a single job posting. Job seekers tend to receive job suggestions which do not match their actual work abilities and educational background and professional ambitions. The traditional recruitment process experiences difficulties because employers need to match specific job qualifications with the abilities shown by candidates.

Artificial Intelligence (AI) and Natural Language Processing (NLP) have improved recruitment systems through their recent technological developments Which resulted in major upgrades to modern recruitment systems. Systems can understand the semantic meaning of textual information through transformer- based language models and contextual embedding techniques which eliminate the need for keyword frequency analysis. The system processes unstructured text through Advanced Text Mining Techniques which extract relevant entities and determine how similar resumes are to job descriptions. Recruitment platforms improve their matching results through AI-driven systems which boost their performance and make the process more efficient.

The study presents a web-based job search system which uses intelligent technology to filter job applications and match skills to job positions. The system provides tailored job recommendations through automatic resume parsing and job classification with semantic similarity calculation. spaCy tool extracts and structures resume information to streamline the process of locating skills and education and work experience information. Sentence Transformers computes the semantic similarity between the candidate profiles and job descriptions while DistilBERT handles the job classification for contextual information. The platform uses Django as its backend framework and React as its frontend framework together with relational databases to manage data efficiently. The recommendation engine uses a threshold to rank job postings which match established criteria for presenting relevant results to users.

II. RELATED WORK

Several studies have also been conducted on the application of Natural Language Processing and machine learning methodologies to enhance recruitment systems and job recommendation systems. In general, traditional recruitment systems use keyword-based matching systems. However, these systems do not effectively consider the contextual relationship between the skills of candidates and the requirements of jobs. Gautam at al. [1] developed a resume parsing system that uses Natural Language Processing methodologies to effectively extract significant information from resumes. In addition, Gupta et al. [2] also developed an information extraction system for skill matching to effectively organize information for recruitment purposes.

Chandak et al. [3] proposed a resume parser and job recommendation system using machine learning techniques to improve candidate filtering. In addition, Rezaeipourfarsangi and Milios [4] presented an AI-based document ranking approach for matching resumes with job descriptions using deep learning techniques.

Recent research has also explored semantic matching techniques for recruitment systems. Deshmukh and Raut [5] proposed a resume screening system using Sentence-BERT to improve candidate selection accuracy.

Transformer-based models have also been widely applied for semantic understanding of text. Devlin et al. [6] introduced the BERT model, which significantly improved contextual language understanding. DistilBERT, proposed by Sanh et al. [7], provides a lighter and faster version of BERT while maintaining strong performance.

Reimers and Gurevych [8] introduced Sentence-BERT, which generates semantic embeddings for sentences and documents, enabling efficient similarity comparison between candidate profiles and job descriptions.

The theoretical foundation for information retrieval and semantic analysis is discussed in previous studies such as Manning et al. and Jurafsky and Martin [9]. Natural Language Processing libraries such as spaCy [10] provide efficient tools for resume parsing and information extraction. General concepts of artificial intelligence are also discussed in Russell and Norvig [11].

These studies demonstrate the importance of semantic analysis and deep learning models in improving recruitment systems and job recommendation platforms.

Furthermore, recent advancements in Artificial Intelligence have enabled the integration of intelligent recommendation systems in recruitment platforms. These systems utilize semantic analysis and machine learning models to better understand candidate profiles and job requirements, thereby improving the accuracy and efficiency of job matching.

III. EXISTING SYSTEM

The current recruitment systems mainly use keyword-based matching algorithms to compare the resumes of the candidates with the job description. Such recruitment systems are designed to match the exact keyword-based search, not the contextual implications of the skills and qualifications possessed by the candidates. Moreover, the process of screening the resumes is a time-consuming task for the recruiter. Although some recruitment systems use machine learning algorithms, the algorithms are not efficient as they don't contain the features of semantics and personalization, making the recruitment process inefficient.

IV. PROPOSED SYSTEM

The proposed system introduces an AI-driven intelligent web platform designed to improve job search and skill matching through semantic analysis. The system utilizes Natural Language Processing techniques to analyze resumes and job descriptions and identify contextual relationships between candidate skills and job requirements. Resume data is extracted using spaCy, while semantic similarity between candidate profiles and job descriptions is computed using Sentence Transformers. DistilBERT is applied for contextual job classification, and a threshold-based recommendation engine ranks job postings according to similarity scores. This approach improves matching accuracy, reduces manual screening effort, and provides personalized job recommendations in an efficient and scalable recruitment platform.

V. SYSTEM ARCHITECTURE

The proposed system follows a modular architecture consisting of data input, preprocessing, AI processing, recommendation engine, and user interface layers. Job seekers upload resumes while recruiters post job descriptions through the web interface. Resume data is extracted and structured using spaCy. Semantic embeddings are obtained by using the Sentence Transformers library, and the job classification task has been performed by utilizing the DistilBERT library. The similarity score has also been obtained to match the candidate profile with the job requirements. Django has been used in the backend development, and the frontend development has been performed by utilizing the React framework.

A. Layered Design Principles

The platform is organized into eight distinct layers to separate ingestion, storage, processing, and delivery. This structure follows contemporary AI infrastructure guidance to ensure that high-latency AI tasks, such as embedding generation, do not block user-facing operations. Table 1 outlines the purpose and technologies for each layer.

B. Microservices and Domain Decomposition

The system is decomposed into domain-specific microservices. This modularity allows the platform to use different technologies for different tasks and facilitates easier model rollouts.

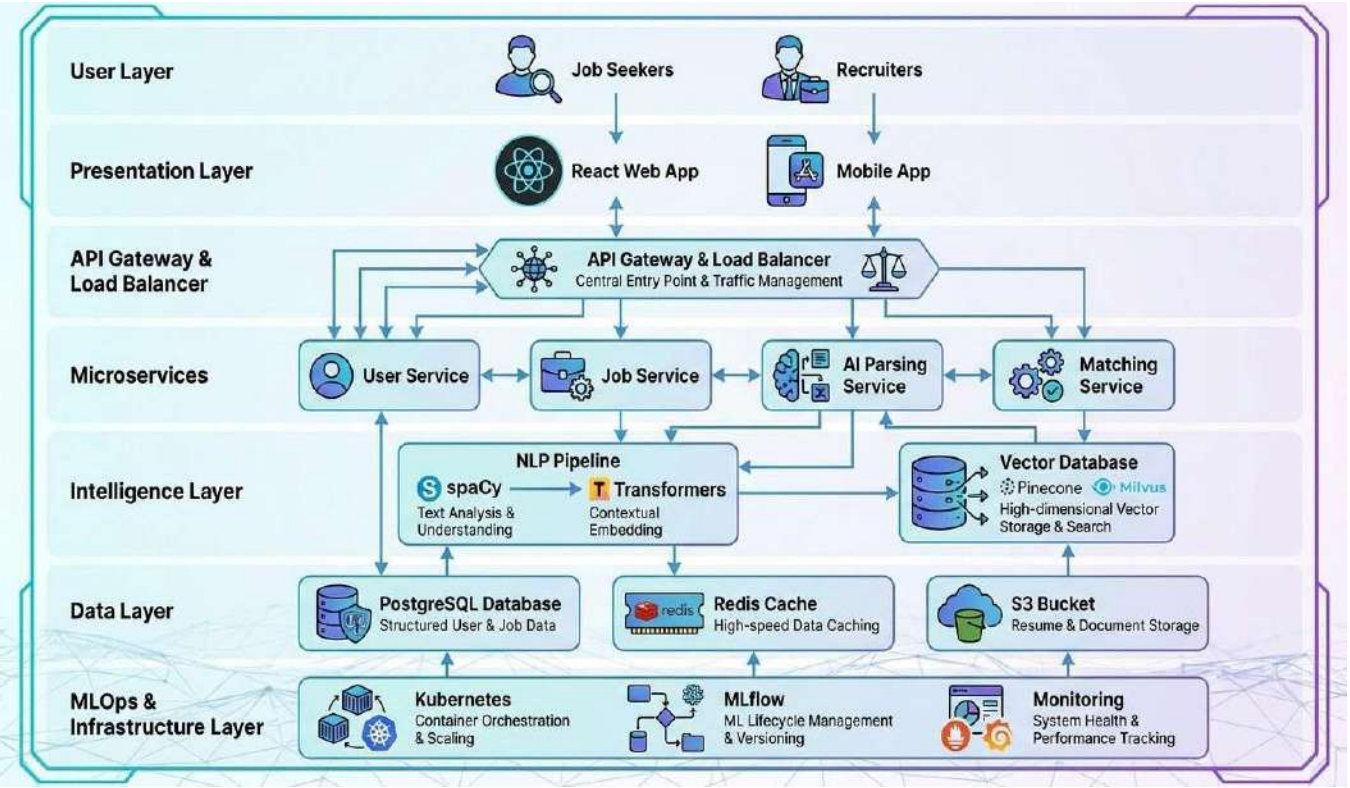

Figure 1: System architecture of the intelligent job search and skill matching platform illustrating the workflow from resume upload to job recommendation generation.

Table 1: Layered Design Principles and Technologies

Layer	Purpose	Examples
Ingestion	Collect/enrich job feeds and resumes.	Kafka, ETL pipelines.
Storage	Long-term storage and metadata.	Cloud store, SQL/NoSQL.
Feature Store	Serve features and embeddings.	Redis, Hopsworks.
Vector Search	High-dimensional ANN retrieval.	Pinecone, Milvus.
Knowledge Graph	Canonicalize skills and support explainability.	Neo4j, GraphDB.
Model Training	Experiment and train models.	MLflow, Kubeflow.
Model Serving	Low-latency inference and API.	Seldon, BentoML.
Orchestration	Coordinate pipelines and monitoring.	Airflow, Prometheus.

- Ingestion Service: Handles the reception of resumes and job postings via a secure API.
- Profile Normalization Service: Uses spaCy to parse resumes and extract structured entities.
- Embedding Generation Service: Invokes Sentence Transformer models to convert text into high-dimensional vectors.
- Retrieval and Ranking Service: Queries the vector database using ANN search.
- API Gateway: Acts as the single-entry point for the React frontend.
- Explainability Service: Interacts with the Knowledge Graph to provide transparency.

C. Vector Stores and Semantic Retrieval

Semantic search is centered on embeddings and an Approximate Nearest Neighbor (ANN) index. The design covers embedding generation, index maintenance, and query serving. The system utilizes production-proven ANN libraries (e.g., FAISS) to enable sub-second retrieval. The index is tuned for an optimal tradeoff between latency and recall.

D. MLOps and Productionization

Robust MLOps is required for continuous training, safe deployment, and observability. The platform utilizes cloud-native orchestration (Kubernetes) to manage container lifecycle, autoscaling, and health checks. CI/CD for models ensures that each model version is contaierized and undergoes automated tests.

E. Knowledge Graph and Explainability

A dedicated Knowledge Graph layer is integrated to canonicalize skills and roles. This layer maps relationships between related skills and provides the foundation for the system's explainability features.

VI. IMPLEMENTATION

Figure 2: User role module showing the two primary roles: job applicants and employers (For Applicants)

Figure 2: Interface that allows users to choose their role in the system as either an applicant or an employer. Based on the selected role, the system redirects the user to the corresponding authentication and system modules.

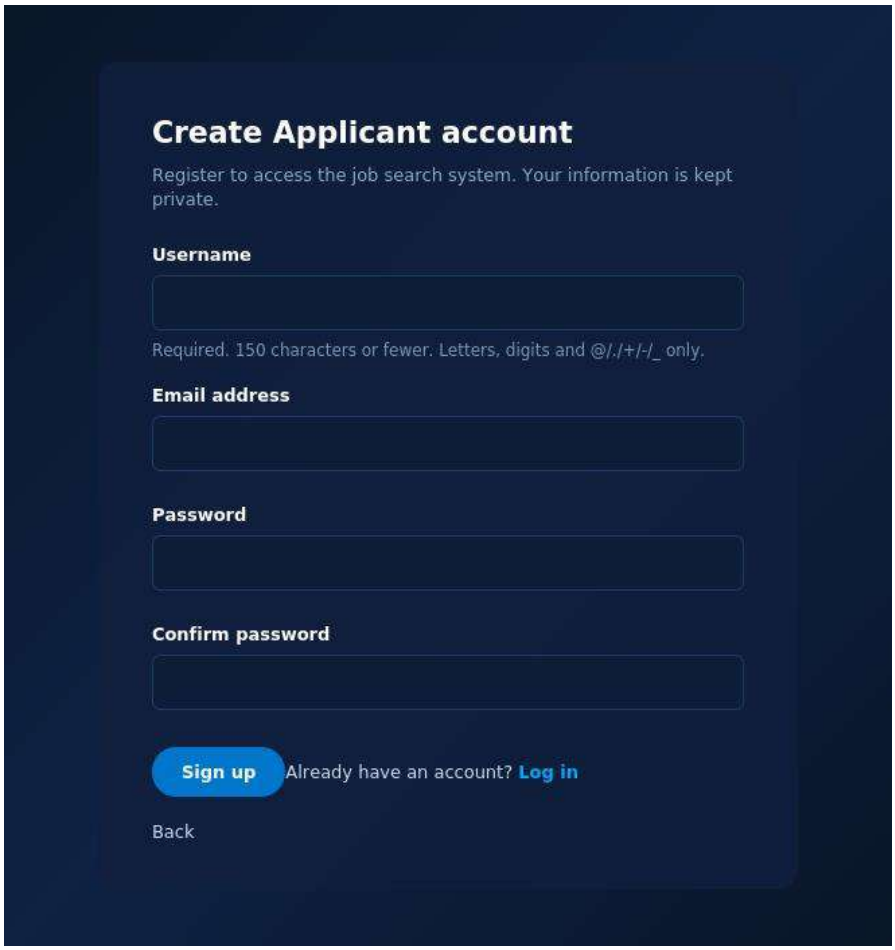

Figure 3: Applicant registration interface used for creating a candidate profile by entering personal and professional information

Figure 3: Applicant registration interface used to create a new account by entering basic details such as username, email address, and password. This process enables new users to access the job search platform.

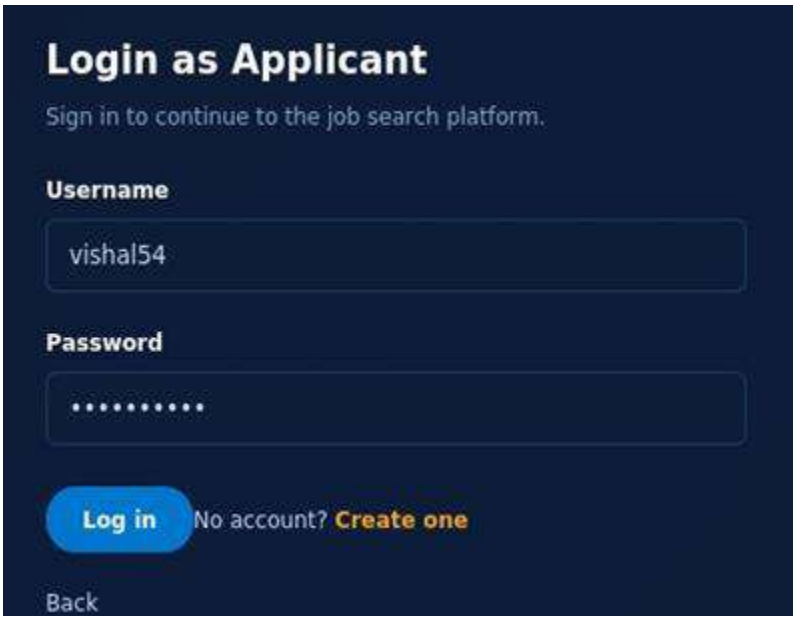

Figure 4: Applicant login interface that allows registered users to securely access the job recommendation platform

Figure 4: Interface that allows registered applicants to securely log in to the system using their username and password. After successful authentication, users can access the job recommendation features of the platform.

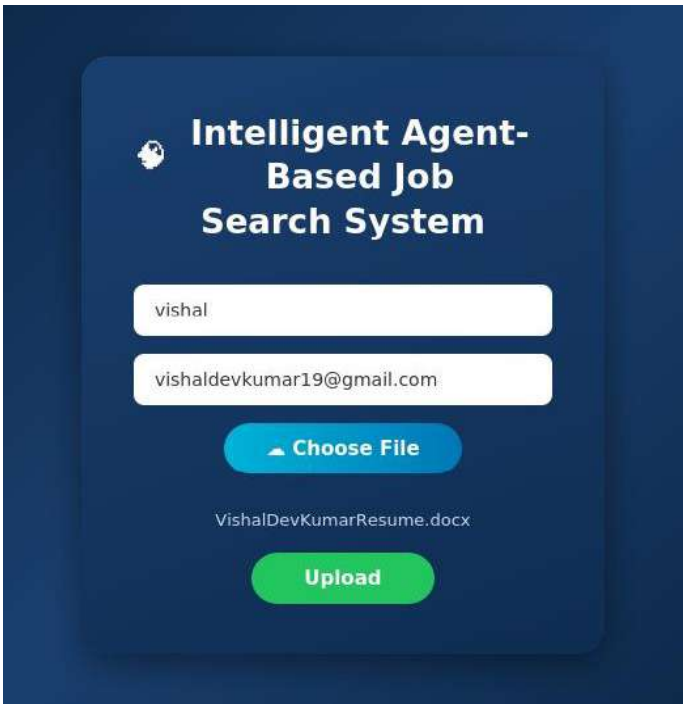

Figure 5: Resume upload module where applicants submit their resumes for automated skill extraction and analysis

Figure 5: Resume upload interface where applicants submit their resumes to the system. The uploaded resume is processed to extract relevant skills and candidate information.

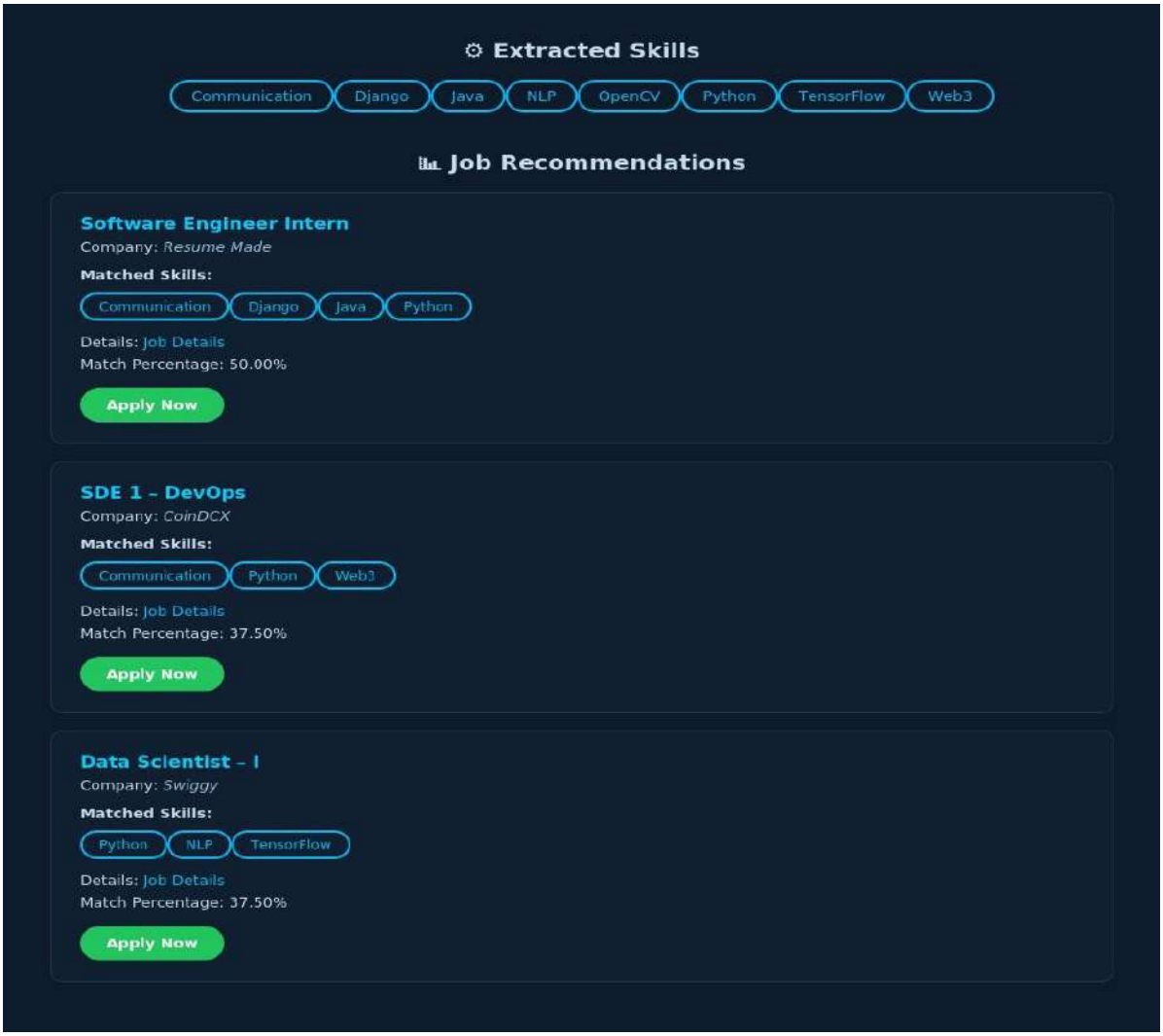

Figure 6: Job recommendation interface displaying extracted skills and matched job suggestions

Figure 6: Interface displaying extracted skills and job recommendations generated by the system. The platform analyzes the applicant's resume and suggests suitable job opportunities with matching skills.

Figure 7: Employer registration interface for creating recruiter accounts to post job openings(For Interviewer)

Figure 7: Employer registration interface that allows recruiters to create an account by providing basic details such as username, email, and password. This enables employers to access recruitment features.

Figure 8: Employer login interface enabling recruiters to access the system and manage job postings

Figure 8: Employer login interface that enables registered recruiters to securely access the platform. After logging in, employers can manage job postings and view candidate matches.

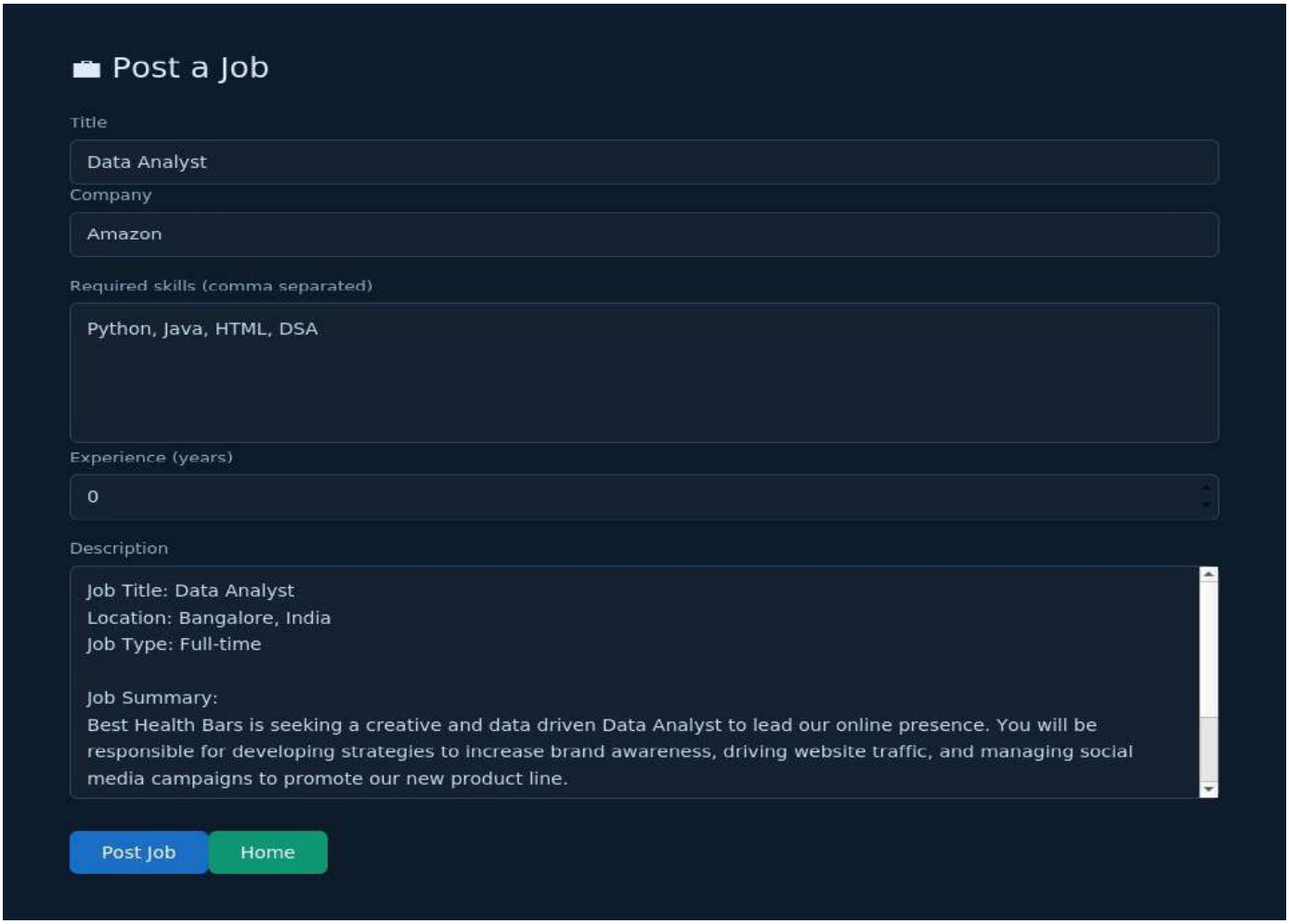

Figure 9: Job posting interface used by employers to create and publish job descriptions on the platform.

Figure 9: Job posting interface where employers can enter job details such as title, required skills, experience, and description. The system uses this information to identify suitable candidates.

Figure 10: Matched applicant results showing ranked candidate profiles based on semantic similarity with job requirements

Figure 10: Interface displaying matched applicants for a posted job based on skill similarity and system analysis. Employers can review candidate profiles and resumes to select suitable applicants.

VII. EVALUATION AND RESULT

A. Dataset and Experimental Setup

A dataset comprising various job descriptions and applicant profiles was collected. The text was preprocessed using standard NLP techniques. For resume parsing, spaCy was utilized to extract relevant entities which were then transformed into embeddings.

Table 2: Performance Comparison of Job Applicant Matching Methods

Method	Precision	Recall	Accuracy
TF-IDF	68%	64%	66%
BERT	82%	80%	81%
Proposed Model	88%	86%	87%

Here's how the different methods stack up-
So, transformer-based semantic matching really bumps up the relevance of job recommendations compared to old-school keyword matching. The hybrid approach keeps things robust and makes sure the recommendations stay accurate.

B. Performance Metrics

To measure how well the system works, we used text similarity scoring. When we didn't have embeddings available, we built a TF-IDF document–term matrix using TF-IDF Vectorizer. In other words, we took all the job descriptions and applicant text and turned them into vectors. Then we calculated the similarity between the applicant and the job post using cosine similarity. These are the vectors that represent the job descriptions. The remaining vectors represent the applicants.
Once we were able to get this score, we were then able to acquire it into percentages, thus allowing us to easily compare each candidate based on the similarity of the job requirements and the applicant text.
This method is significant to us since it provides us a quick way of determining how similar a candidate's profile is to a certain job description.

C. System Robustness and Fallback Mechanism

The evaluation compared traditional TF-IDF matching with the proposed transformer-based semantic matching. The results, summarized in Table 2, indicate that the proposed approach provides a significant improvement.

VIII. CONCLUSION

This paper presented an intelligent web platform for job search and skill matching using Natural Language Processing and transformer-based techniques. The proposed system addresses the limitations of traditional keyword-based recruitment platforms by introducing semantic analysis for better understanding of candidate profiles and job descriptions. The system integrates spaCy for automated resume parsing, Sentence Transformers for semantic similarity computation, and DistilBERT for contextual job classification.
By leveraging these technologies, the platform is able to analyze unstructured textual data from resumes and job postings and generate more relevant job recommendations. The exper-

imental evaluation demonstrates that semantic similarity techniques provide more accurate matching compared to traditional keyword-based approaches. Additionally, the hybrid matching strategy incorporating TF-IDF as a fallback mechanism improves system reliability and ensures continuous operation even when embedding services are unavailable.

The proposed system can offer a viable solution for modern recruitment systems, reducing human screening and increasing the relevance of recommended jobs. Overall, it is evident that the system has the potential for applying ai-driven technologies to improve efficiency and accuracy in digital recruitment systems.

IX. FUTURE SCOPE

- Real-time labor market analytics: ensure that the skills and trends in the recommendations remain up-to-date.
- Advanced generative AI for resume/JD improvement: automatically enhance resumes and job descriptions for better matches.
- Explainable AI (XAI) for recommendations: provide explanations for the matches in order to increase the level of trust with employers and candidates.
- Continuously retrain the models with new data: update the recommendations according to the changing trends in the market.

CONFLICT OF INTEREST

The authors declared that they have no conflict of interest.

REFERENCES

[1] Gautam, D. Sharma, and V. Chaturvedi, "Automating Talent Acquisition Assisted by Resume Parsing System for Enhanced Job Matching," *IEEE*, 2024. Available from: https://ieeexplore.ieee.org/document/10932337

[2] Gupta, P. Thapliyal, S. Aggarwal, S. Pokhariyal, and A. Das, "Information Extraction from Resume for Skill Matching," *IEEE Conference Publication*, 2024. Available from: https://ieeexplore.ieee.org/document/10911176

[3] A. V. Chandak, H. Pandey, G. Rushiya, and H. Sharma, "Resume Parser and Job Recommendation System using Machine Learning," *IEEE Conference Publication*, 2024. Available from: https://ieeexplore.ieee.org/document/10481635

[4] S. Rezaeipourfarsangi and E. E. Milios, "AI-Powered Resume-Job Matching: A Document Ranking Approach Using Deep Neural Networks," in *Proc. ACM Symposium on Document Engineering (DocEng '23)*, 2023. Available from: https://doi.org/10.1145/3573128.3609347

[5] A. Deshmukh and A. Raut, "Enhanced Resume Screening for Smart Hiring Using S-BERT," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 8, 2024. Available from: https://doi.org/10.14569/ijacsa.2024.0150825

[6] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proc. NAACL-HLT*, 2019. Available from: https://aclanthology.org/N19-1423/

[7] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT: A Distilled Version of BERT," *arXiv preprint arXiv:1910.01108*, 2019. Available from: https://doi.org/10.48550/arXiv.1910.01108

[8] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in *Proc. EMNLP*, 2019. Available from: https://aclanthology.org/D19-1410/

[9] D. Jurafsky and J. H. Martin, "Speech and Language Processing," Pearson, 3rd ed., 2023. Available from: https://ieeexplore.ieee.org/document/10391814

[10] M. Honnibal and I. Montani, "spaCy 2: Natural Language Understanding with Bloom Embeddings," Explosion AI, 2017. Available from: https://cir.nii.ac.jp/crid/1370021390573874949

[11] S. Russell and P. Norvig, "Artificial Intelligence: A Modern Approach," Pearson, 4th ed., 2021.