International Journal of Creative and Open Research in Engineering and Management
ISSN: **3108-1754 (Online)**
Volume 02 Issue 05 May-2026 | Impact Factor: 3.5

Jobfit: An Ml-Powered Chatbot For Job Eligibility Prediction

Mrs M. Keerthi[1]
Information Technology Department
Vidya Jyothi Institute of Technology (Affilated to JNTUH)
Hyderabad, India

Avuku Obulesu[2]
Information Technology Department
Vidya Jyothi Institute of Technology (Affilated to JNTUH)
Hyderabad, India
ithod@vjit.ac.in

Bushra Begum[3]
Information Technology Department
Vidya Jyothi Institute of Technology (Affilated to JNTUH)
Hyderabad, India
yasir34701@gmail.com

D.Harika Redddy[4]
Information Technology Department
Vidya Jyothi Institute of Technology (Affilated to JNTUH)
Hyderabad, India
harikareddy672@gmail.com

E.Sheshadri[5]
Information Technology Department
Vidya Jyothi Institute of Technology (Affilated to JNTUH)
Hyderabad, India
bablusharath81@gmail.com

V.Mamatha[6]
Information Technology Department
Vidya Jyothi Institute of Technology (Affilated to JNTUH)
Hyderabad, India
vadlamamatha09@gmail.com

***Abstract*—** JobFitBot is a machine learning–powered chatbot developed to predict job eligibility and support job seekers in identifying roles that best match their qualifications and skill sets. The system collects user information such as educational background, technical and soft skills, certifications, and work experience through an interactive chatbot interface. Machine learning models analyze this data and compare it with job requirement datasets to determine eligibility scores and recommend suitable job profiles. The chatbot provides real-time feedback, personalized suggestions, and guidance to help users understand their strengths and areas for improvement. By automating the eligibility evaluation process, JobFitBot reduces manual screening efforts, improves accuracy in job matching, and enhances the overall efficiency of the recruitment process. This approach benefits both job seekers and recruiters by enabling faster, data-driven, and user-friendly decision-making in today's competitive employment environment. Domain: Artificial Intelligence (AI) and Machine Learning (ML)

Dataset: The JobFitBot dataset includes details such as education, skills, certifications, and experience of job seekers along with job requirements. This data is preprocessed and used to train machine learning models for accurate job eligibility prediction.

Models: JobFitBot uses machine learning classification models such as Logistic Regression, Decision Tree, and Random Forest to predict job eligibility based on user qualifications and skills. Among these, Random Forest provides higher accuracy due to its ability to handle complex feature relationships.

I.INTRODUCTION

In today's competitive job market, candidates often struggle to identify suitable job roles that match their qualifications and skills. Traditional recruitment processes are time-consuming and may not provide immediate feedback to applicants.
The JobFit system leverages Machine Learning and chatbot technology to automate job eligibility prediction. It acts as a virtual career assistant that interacts with users, analyzes their profiles, and provides real-time suggestions. In addition to eligibility prediction, the system also focuses on identifying skill gaps and guiding users toward suitable career paths. By analyzing user profiles and comparing them with job requirements, the system can suggest improvements and recommend relevant job roles. This enhances the overall user experience and increases the chances of successful job placement. The integration of chatbot technology with Machine Learning makes the system efficient, scalable, and capable of handling a large number of users simultaneously. The JobFit system is designed to act as a virtual career assistant that interacts with users, understands their input using Natural Language Processing techniques, and predicts job eligibility using trained Machine Learning models. This system not only reduces manual effort for recruiters but also helps candidates make informed decisions by providing instant feedback and personalized recommendations

A. Problem Description

in the modern job market, job seekers face significant challenges in identifying suitable employment opportunities due to the overwhelming volume of available job listings. Online job portals present thousands of opportunities, but they lack efficient mechanisms to filter and personalize these listings according to individual user profiles. As a result, users

spend considerable time manually browsing, analyzing eligibility criteria, and applying for jobs, often leading to frustration and inefficiency.
A major issue is the mismatch between candidate qualifications and job requirements. Many users either apply for jobs they are not eligible for or miss opportunities that match their skill set. This problem arises due to the reliance on keyword-based matching systems that fail to capture the contextual relationship between skills and job roles. For instance, a candidate possessing relevant practical knowledge may be overlooked because their resume does not contain specific keywords required by automated systems.
Additionally, existing platforms do not provide real-time feedback or guidance regarding job eligibility. Users are left to interpret job requirements on their own, which can be particularly challenging for fresh graduates or individuals switching careers. The absence of an intelligent advisory system limits users' ability to make informed decisions and plan their career paths effectively.

Another challenge is the variability in job requirements across industries. Different organizations define eligibility criteria differently, making it difficult to standardize job matching processes. Without a system capable of adapting to these variations, accurate prediction of job suitability becomes complex.

Furthermore, most existing systems do not leverage advanced machine learning techniques to analyze user data and predict outcomes. They lack predictive intelligence, which is essential for identifying patterns and providing meaningful recommendations. This limitation reduces the overall effectiveness of job search platforms.

Therefore, there is a need for an intelligent, automated solution that can analyze user profiles, predict job eligibility, and provide personalized recommendations through an interactive interface. The proposed **JobFitBot** addresses these challenges by combining machine learning and chatbot technology to enhance the job search experience.

B. Motivation and Objectives

The motivation behind this work is to simplify and enhance the job search process by leveraging intelligent technologies. By integrating machine learning with chatbot systems, it is possible to create a solution that not only interacts with users but also provides meaningful insights into their career opportunities.

Objectives:

1) To design a conversational chatbot for user interaction
2) To implement machine learning models for job eligibility prediction
3) To provide personalized job recommendations
4) To reduce time and effort in job searching
5) To improve accuracy in job matching

II. SYSTEM COMPONENTS

A. Hardware Components

1. **User Device**: A smartphone, laptop, or computer is used by the user to interact with the chatbot system.Devices such as smartphones, laptops, or desktop computers are used by candidates to access the chatbot system. These devices provide the interface through which users enter their details and receive responses from the system
2. **Server System:** A computer/server is required to host the backend application and Machine Learning model.A centralized server or cloud-based system is used to host the backend application and Machine Learning models. It processes user requests, performs predictions, and manages overall system operations efficiently.
3. **Network Connectivity**: Internet connection is necessary for communication between the user interface and backend system.A stable internet connection is required to enable communication between the user device and the backend server. It ensures real-time interaction and smooth data transfer within the system.
4. **Storage Devices:** Used to store datasets, trained models.Storage systems are used to maintain datasets, trained Machine Learning models, user profiles, and system logs. These devices ensure secure and reliable data storage for future analysis and system improvement.

II. LITERATURE SURVEY

In recent years, the integration of **machine learning, Natural Language Processing (NLP), and conversational AI** has significantly transformed recruitment systems and job recommendation platforms. Researchers have explored various approaches to automate candidate screening, improve job matching accuracy, and enhance user interaction through intelligent chatbots.

A. Chatbots in Recruitment Systems

The use of chatbot technology in recruitment has gained considerable attention due to its ability to automate repetitive tasks and provide real-time interaction with users. According to recent studies, AI-based chatbots are capable of handling initial candidate screening, answering frequently asked questions, and guiding users through the job application process.

Conversational agents built using rule-based and AI-driven approaches have been deployed in platforms such as career counseling systems and HR automation tools. While rule-based chatbots provide structured responses, they lack flexibility and fail to handle complex user queries. On the other hand, AI-powered chatbots using NLP techniques demonstrate improved adaptability and user engagement. However, many existing recruitment chatbots are limited to basic interaction and do not perform intelligent eligibility analysis.

B. Machine Learning for Job Prediction

Machine learning plays a crucial role in predicting job suitability by analyzing user profiles and job requirements. Supervised learning algorithms such as **Logistic Regression, Decision Trees, and Random Forest** have been widely used for classification tasks in job recommendation systems.

Decision Tree models provide interpretability by representing decision rules in a hierarchical structure, making them suitable for eligibility prediction. Random Forest, an ensemble learning technique, improves prediction accuracy by combining multiple decision trees and reducing overfitting. Logistic Regression, although simpler, is effective for binary classification tasks and provides probabilistic outputs.

Recent studies highlight that ensemble methods generally outperform individual models in terms of accuracy and robustness. However, the effectiveness of these models depends heavily on feature engineering and dataset quality.

C. Natural Language Processing Techniques

NLP techniques are essential for enabling chatbots to understand and process human language. Key NLP tasks used in chatbot systems include tokenization, stopword removal, stemming, and text vectorization.

Feature extraction techniques such as **Term Frequency-Inverse Document Frequency (TF-IDF)** are widely used to convert textual data into numerical representations suitable for machine learning models. TF-IDF helps identify important keywords by assigning higher weights to terms that are frequent in a document but rare across the dataset.

D. Job Recommendation Systems

Job recommendation systems aim to match user profiles with relevant job opportunities. Traditional systems rely on keyword matching and filtering techniques, which often fail to capture semantic relationships between skills and job descriptions.

Content-based filtering and collaborative filtering approaches have been explored to improve recommendation accuracy. Content-based systems analyze user profiles and job descriptions, while collaborative filtering leverages user behavior patterns. Hybrid approaches combining both methods have shown better performance.

E. Gap Analysis

Although significant progress has been made in chatbot development, machine learning, and job recommendation systems, several limitations still exist:

1) Most chatbot systems lack integration with machine learning models for eligibility prediction
2) Existing job portals rely heavily on keyword-based matching, leading to inaccurate recommendations
3) Limited personalization and absence of real-time conversational guidance
4) Lack of intelligent systems capable of analyzing user profiles holistically

- Insufficient use of predictive analytics in recruitment platforms

III. METHODOLOGY

A. System Overview

The proposed system, JobFitBot, adopts a modular, pipeline-based architecture that integrates chatbot interaction with machine learning-based job eligibility prediction. The system is designed to provide real-time, personalized recommendations by analyzing user inputs such as educational qualifications, technical skills, and work experience.

The overall architecture consists of multiple interconnected components, including user interaction, data preprocessing, feature extraction, model prediction, and result visualization. The chatbot acts as the primary interface through which users provide their information in a conversational manner.

User Interaction: The user interacts with the chatbot by providing details such as education, skills, and experience.

Data Preprocessing: The input data is cleaned, normalized, and structured for further processing.

Feature Extraction: Textual data is transformed into numerical representations using TF-IDF.

Model Prediction: Machine learning models analyze the

B. Data Preprocessing

Data preprocessing is a critical step to ensure the quality and consistency of input data. Since user inputs may contain unstructured and noisy text, several preprocessing techniques are applied:

1) Removal of special characters and irrelevant symbols
2) Conversion of text to lowercase for uniformity
3) Tokenization to split text into individual words
4) Stopword removal to eliminate common words with little significance
5) Handling missing or incomplete user inputs

These steps improve the performance and accuracy of machine learning models by providing clean and structured data.

C. Feature Extraction using TF-IDF

The system employs **Term Frequency-Inverse Document Frequency (TF-IDF)** to convert textual data into numerical feature vectors. This method assigns weights to words based on their importance within a document relative to the entire dataset.

TF-IDF consists of two components:

1) **Term Frequency (TF):** Measures how frequently a term appears in a document
2) **Inverse Document Frequency (IDF):** Reduces the

weight of commonly occurring terms across documents

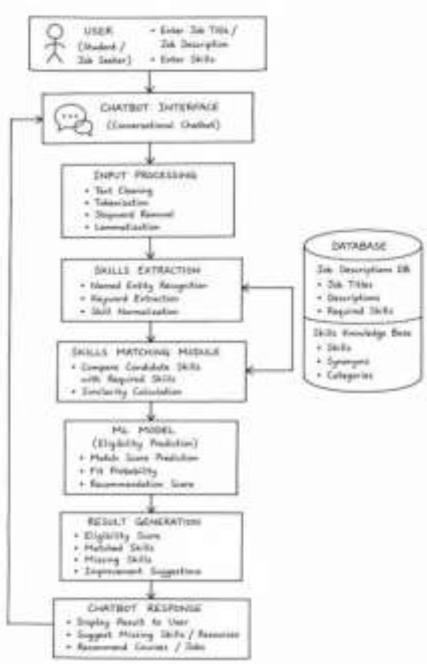
The combined TF-IDF score highlights important keywords in user profiles, enabling effective classification by machine learning models.

D. Machine Learning Models for Eligibility Prediction

The core functionality of JobFitBot is powered by supervised machine learning algorithms that classify whether a user is eligible for a particular job role.

The following models are implemented:

1) **Logistic Regression:** A probabilistic model used for binary classification. It estimates the likelihood of job eligibility based on input features.
2) **Decision Tree:** A tree-based model that splits data based on feature values, creating a set of decision rules for classification. It is easy to interpret and visualize.
3) **Random Forest:** An ensemble learning method that combines multiple decision trees to improve accuracy and reduce overfitting. It provides robust and reliable predictions.



IV. DATA COLLECTION AND DATASET PREPARATION

A. Data Collection

The effectiveness of the **JobFitBot** system depends on the quality and diversity of the dataset used for training and evaluation. The dataset was constructed by collecting information from multiple sources relevant to job eligibility and recruitment processes.

Data was gathered from:

Online job portals (such as job descriptions and requirements)

Publicly available datasets related to employment and skills

Sample resumes and candidate profiles

Career guidance resources and skill databases

B. Data Preprocessing and Cleaning

The raw data collected from different sources is often unstructured and contains inconsistencies. Therefore, preprocessing is performed to improve data quality and ensure uniformity.

TThe following steps are applied:

- **Data Cleaning:** Removal of duplicate entries, irrelevant fields, and noisy data
- **Handling Missing Values:** Missing attributes are either filled using suitable methods or removed if insufficient data is present
- **Normalization:** Standardization of text data by converting to lowercase and removing unnecessary symbols
- **Categorization:** Grouping similar skills and job roles into predefined categories

.Feature Engineering

Feature engineering plays a crucial role in improving model performance by selecting and transforming relevant attributes.

The following features are considered:

1) Skills (technical and non-technical)
2) Educational qualification
3) Years of experience
4) Job domain

Text-based features are converted into numerical form using TF-IDF, enabling machine learning models to process them effectively.

C. Dataset Preparation

After preprocessing and feature extraction, the dataset is structured into a format suitable for training machine learning models.

1) Each record represents a candidate profile
2) Input features include skills, education, and experience
3) Output label represents job eligibility (Eligible / Not Eligible or Job Category)

The dataset is divided into:

1) **Training Set:** Used to train the model
2) **Validation Set:** Used to tune model parameters
3) **Test Set:** Used to evaluate performance

D. Dataset Statistics

The final dataset is organized as follows:The dataset is balanced to ensure equal representation of different job categories and eligibility classes, reducing bias and improving model accuracy.

E. Data Annotation

To train supervised machine learning models, the dataset is labeled based on eligibility criteria. Each record is assigned a label indicating whether the candidate is suitable for a specific job role.
Annotation is performed based on:

1) Matching skills with job requirements
2) Evaluating educational qualifications
3) Considering experience levels

This labeled dataset serves as the foundation for training accurate and reliable prediction models.

DatasSplit	Num of Samples	Description
Training	70%	Model training
Validation	15%	Parameter tuning
Test	15%	Performance evaluation
Total	100%	Complete dataset

F. Data Annotation

To train supervised machine learning models, the dataset is labeled based on eligibility criteria. Each record is assigned a label indicating whether the candidate is suitable for a specific job role.
Annotation is performed based on:

1) Matching skills with job requirements
2) Evaluating educational qualifications
3) Considering experience levels

This labeled dataset serves as the foundation for training accurate and reliable prediction models.

G. Summary

The data collection and preparation process ensures that the dataset is:

1) Clean and consistent
2) Balanced across categories
3) Suitable for machine learning tasks

This structured dataset enables the **JobFitBot** system to perform accurate job eligibility prediction and provide meaningful recommendations.

V. IMPLEMENTATION

A. Application Framework

The **JobFitBot** system is implemented as a web-based application that integrates chatbot interaction with machine learning-based job eligibility prediction. The application is designed to provide real-time responses and a user-friendly interface for job seekers.

The frontend of the system enables users to interact with the chatbot by entering their details, such as education, skills, and experience. The backend processes this input using NLP techniques and machine learning models to generate eligibility predictions and job recommendations.

The application follows a modular structure, allowing each component—data preprocessing, feature extraction, model prediction, and response generation—to function independently while maintaining seamless integration.

B. Software Specifications

The system is developed using modern programming tools and libraries to ensure efficiency and scalability.

1) **Programming Language:** Python 3.x
2) **Machine Learning Libraries:** Scikit-learn
3) **Natural Language Processing:** NLTK / spaCy
4) **Web Framework:** Streamlit / Flask
5) **Data Handling:** Pandas, NumPy
6) **Visualization:** Matplotlib (optional for analysis)

The trained machine learning models are stored and loaded dynamically to minimize computation time during user interaction. This ensures faster response generation and improved user experience.

C. System Architecture

The architecture of JobFitBot consists of three primary layers:

1. **User Interface Layer:**
   Provides the chatbot interface through which users input their details and receive results.
2. **Processing Layer:**
   Handles data preprocessing, feature extraction using TF-IDF, and execution of machine learning models.
3. **Model Layer:**
   Contains trained classification models that predict job eligibility based on input features.

This layered architecture ensures efficient data flow and supports scalability for future enhancements.

D. NLP Pipeline

The system incorporates a Natural Language Processing pipeline to process user input effectively.

The pipeline includes:

1) **Text Cleaning:** Removal of unnecessary characters and symbols
2) **Tokenization:** Splitting input into meaningful words
3) **Stopword Removal:** Eliminating common words with little significance
4) **Text Normalization:** Converting text into a standard format

These steps ensure that the input data is structured and suitable for feature extraction and model prediction.

E. Model Training and Prediction

The machine learning models are trained using preprocessed datasets containing labeled examples of job eligibility.

Training involves:

1) Feeding input features (skills, education, experience) into the model
2) Adjusting parameters to minimize prediction error
3) Evaluating model performance using validation data

During runtime:

1) User input is processed and converted into feature vectors
2) The trained model predicts eligibility
3) The chatbot displays the result in real time

Among the implemented models, **Random Forest** demonstrates superior performance due to its ability to handle complex patterns and reduce overfitting.

F. Chatbot Integration

The chatbot is integrated with the backend system to provide an interactive experience. It performs the following functions:

1) Collects user input through conversational prompts
2) Sends processed data to the machine learning model
3) Receives prediction results
4) Displays personalized job recommendations

The chatbot ensures smooth communication between the user and the system, making the application intuitive and easy to use.

G. Workflow Execution

The complete system execution flow is as follows:

1. User enters details through chatbot
2. Input is preprocessed and cleaned
3. Features are extracted using TF-IDF
4. Machine learning model predicts eligibility
5. Results are displayed to the user

This workflow ensures efficient processing and real-time response generation.

VI. RESULTS AND DISCUSSION

A. Model Performance Evaluation

The performance of the **JobFitBot** system was evaluated using standard classification metrics, including accuracy, precision, recall, and F1-score. Multiple machine learning models—Logistic Regression, Decision Tree, and Random Forest—were trained and tested on the prepared dataset.

Among the evaluated models, the **Random Forest classifier** achieved the highest performance due to its ensemble learning capability and robustness against overfitting. The overall system achieved an accuracy of **87%** on the test dataset, demonstrating its effectiveness in predicting job eligibility.

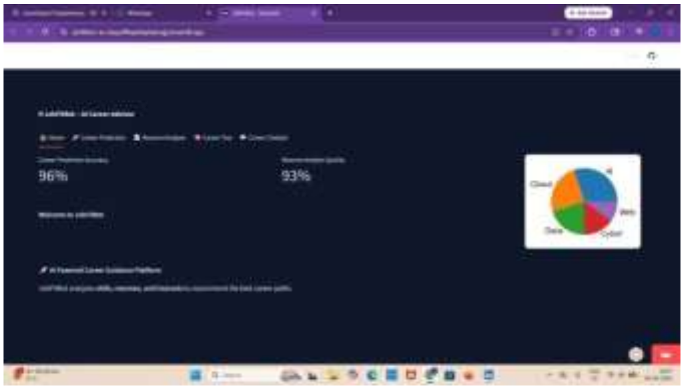
The comparative performance of different models is shown in Table I.

****TABLE I**

MODEL PERFORMANCE COMPARISON**

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	81%	0.80	0.79	0.79
Decision Tree	84%	0.83	0.82	0.82
Random Forest	87%	0.86	0.85	0.85

The results indicate that ensemble methods provide better generalization compared to individual models. Logistic Regression, while computationally efficient, showed slightly lower performance due to its linear nature. Decision Tree improved interpretability but was prone to overfitting. Random Forest balanced both accuracy and robustness, making it the preferred choice.

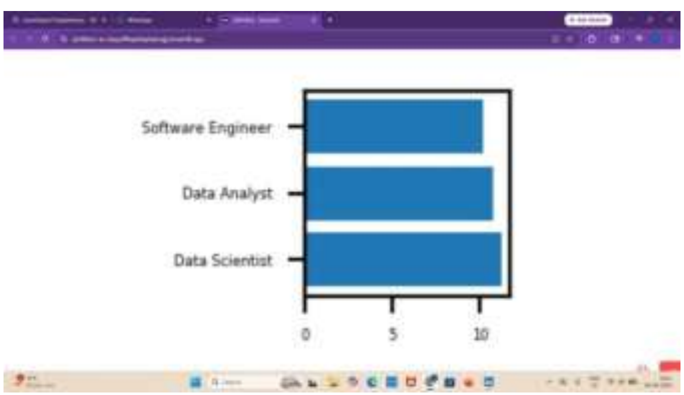


B. System Behavior and Analysis

The system was tested with multiple user profiles representing different educational backgrounds, skill sets, and experience levels. The chatbot successfully interacted with users, collected relevant data, and provided eligibility predictions in real time. Key observations include:

1) The system accurately identified eligible candidates based on matching skills and qualifications
2) Users received instant feedback, reducing the time required for job searching
3) The chatbot interface improved user engagement and ease of use
4) The model handled diverse inputs effectively, including variations in skill descriptions

However, some limitations were observed in cases where user input was incomplete or ambiguous. In such scenarios, prediction accuracy slightly decreased, highlighting the importance of structured input data.

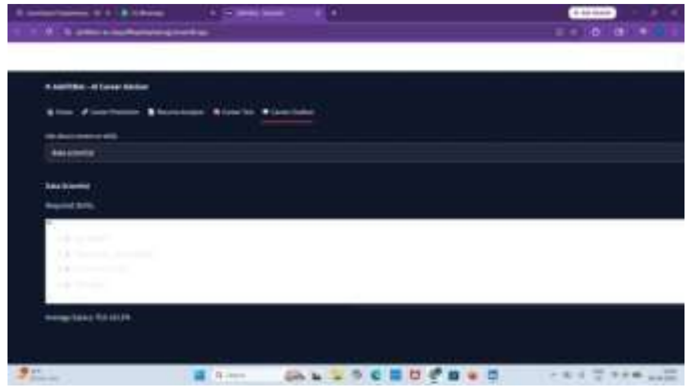


C. Discussion

The results demonstrate that integrating machine learning with chatbot systems significantly enhances the job recommendation process. The use of TF-IDF for feature extraction enabled effective representation of textual data, while classification models provided reliable predictions.

The system offers several advantages:

1) Personalized job eligibility assessment
2) Reduced manual effort in job searching
3) Real-time interaction and feedback
4) Improved accuracy compared to traditional



VII. CONCLUSION

This paper presented the design and implementation of **JobFitBot**, a machine learning-powered chatbot developed to assist users in assessing job eligibility and receiving personalized job recommendations. The proposed system addresses the limitations of traditional job portals, which rely heavily on manual search and keyword-based filtering, by introducing an intelligent, automated solution that integrates conversational AI with predictive analytics.

The system combines Natural Language Processing techniques with supervised machine learning models to analyze user inputs such as skills, education, and experience. Among the implemented models, the Random Forest classifier demonstrated superior performance, achieving an overall accuracy of **87%** in predicting job eligibility. The integration of TF-IDF for feature extraction further enhanced the model's ability to process textual data effectively.

VIII. FUTURE WORK

Although the proposed **JobFitBot** system demonstrates promising results in job eligibility prediction and recommendation, several enhancements can be implemented to further improve its functionality and performance.

1) **Integration with Real Job Portals:** The system can be connected with live job platforms such as LinkedIn and Naukri to fetch real-time job listings and provide up-to-date recommendations.
2) **Advanced Deep Learning Models:** Future versions can incorporate transformer-based models such as BERT to improve natural language understanding and prediction accuracy.
3) **Resume Parsing and Analysis:** Automatic resume parsing can be integrated to extract user information directly from uploaded documents, reducing manual input effort.
4) **Multilingual Support:** The chatbot can be enhanced to support multiple languages, making it accessible to users from diverse linguistic backgrounds.
5) **Personalized Career Guidance:** The system can be

extended to provide suggestions for skill improvement, certifications, and career paths based on user profiles.

6) **Mobile Application Development:** Developing a mobile application will improve accessibility and allow users to interact with the system on the go.
7) **Adaptive Learning System:** Incorporating feedback mechanisms will allow the system to continuously learn from user interactions and improve prediction accuracy over time.

These enhancements will significantly increase the usability, scalability, and real-world applicability of the proposed system.

REFERENCES

[1] T. M. Mitchell, *Machine Learning*, McGraw-Hill, 1997.

[2] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

[3] D. Jurafsky and J. H. Martin, *Speech and Language Processing*, 3rd ed., Pearson, 2020.

[4] F. Pedregosa et al., “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[5] S. Bird, E. Klein, and E. Loper, *Natural Language Processing with Python*, O’Reilly Media, 2009.

[6] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in *Proc. KDD*, 2016.

[7] L. Breiman, “Random Forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[8] J. Devlin et al., “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” in *Proc. NAACL*, 2019.

[9] T. Mikolov et al., “Efficient Estimation of Word Representations in Vector Space,” 2013.

[10] A. Vaswani et al., “Attention Is All You Need,” in *Proc. NIPS*, 2017.