Green University of Bangladesh

Department of Computer Science and Engineering (CSE)
Semester: (Spring, Year: 2025), B.Sc. in CSE (Day)

AI-Powered Next-Gen Job Recommendation System Using NLP and Machine Learning

Lab Report: 03
Course Title: Integrated Design Project II
Course Code: CSE - 406
Section: 213 - D6

<u>Students Details</u>

Name	ID	Role
Irteja Mahmud	213902016	Leader
Pankaj Mahanta	213902002	Member-1
Nazmul Hasan	213902003	Member-2

Submission Date: 01 - 03 - 25
Course Teacher's Name: Babe Sultana

[For teachers use only: Don't write anything inside this box]

Lab Project Status	
Marks:	**Signature:**
Comments:	**Date:**

Chapter 1

Introduction

1.1 Overview

The AI-Powered Next-Gen Job Recommendation System (NGJRS) leverages Natural Language Processing (NLP) and Machine Learning (ML) to enhance job matching. It provides personalized job recommendations by analyzing user profiles, resumes, and job listings. The system offers real-time job search, automated CV parsing, employer-candidate matching, and a user-friendly interface. By integrating external job boards and continuously learning from user interactions, NGJRS ensures accurate and efficient job recommendations, making the hiring process faster and more effective for both job seekers and employers.

1.2 Objectives

The major objectives of this project are:

- **Develop an AI-Based Job Recommendation System:** Implement an intelligent recommendation engine that utilizes NLP and machine learning to enhance job matching accuracy.

- **Improve Candidate-Job Suitability:** Use BERT embeddings and semantic similarity techniques to ensure that job recommendations align with candidate skills and preferences.

- **Optimize the Recruitment Process:** Provide recruiters with a ranked list of potential candidates based on job requirements, reducing manual filtering efforts.

- **Enhance User Experience:** Design an intuitive user interface for job seekers and employers, ensuring ease of use and seamless navigation.

- **Integrate External Job Data Sources:** Aggregate job listings from various external job boards to provide users with up-to-date employment opportunities.

Chapter 2

Module Description

This section provides a detailed description of the three main components of the Next-Gen Job Recommendation System (NGJRS): Job Recommendation, Job Searching, and User Profile Management. Each module is explained with its core functionality, logic, and processing steps.

2.0.1   Job Recommendation Module

The Job Recommendation module is responsible for suggesting the most relevant job opportunities to users by analyzing their profiles, uploaded CVs, job preferences, and historical interactions. It leverages advanced Natural Language Processing (NLP) and Machine Learning (ML) techniques to provide highly personalized recommendations.

Process Flow

- **User Data Collection:**

    - Gathers user profile information, uploaded CV, search history, and applied jobs.

    - Extracts key skills, experience, and preferences for job recommendations.

- **Preprocessing and Feature Extraction:**

    - Text cleaning, tokenization, and vectorization of job descriptions and user profiles.

    - NLP techniques like **TF-IDF, Word2Vec, and BERT embeddings** are used for semantic analysis.

- **Job Matching Algorithm:**

    - **Similarity-based Matching:** Uses **Cosine Similarity, Jaccard Similarity, and Euclidean Distance** to measure job-profile relevance.

    - **Machine Learning-based Classification:** Predicts job fit using **Random Forest, Logistic Regression, and Neural Networks**.

- **Ranking and Recommendation:**

  - Jobs are assigned a **relevance score** based on skill match, experience, and location.
  - Weight factors such as **job application trends, popularity, and employer ranking** are considered.

- **Feedback Loop and Adaptive Learning:**

  - User feedback (clicks, saved jobs, applied jobs) refines recommendation accuracy.
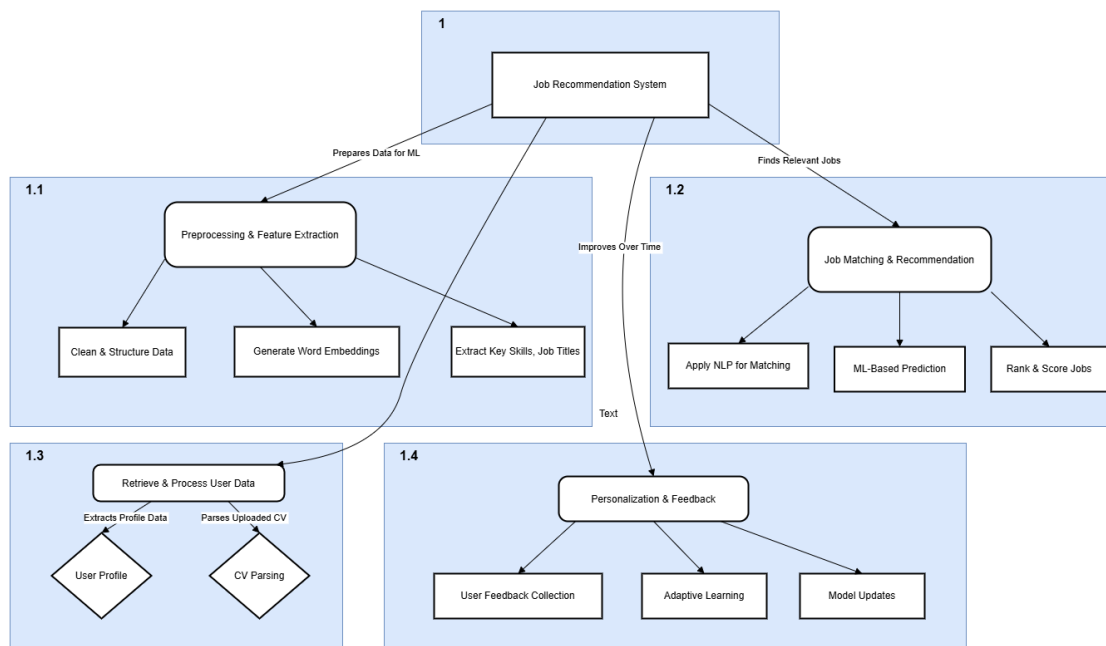 - Reinforcement learning adjusts future recommendations based on user interactions.

Figure 2.1: Job Recommendation System Architecture

2.0.2 Job Searching Module

The Job Searching module enables users to search for relevant job opportunities by processing their queries, applying filters, ranking results, and retrieving job data from various sources. This module ensures accurate and efficient job recommendations by utilizing NLP-based query processing, filtering mechanisms, and machine learning ranking.

Process Flow

- **Query Processing:** The system extracts keywords from the user input, applies semantic analysis for better understanding, and corrects spelling errors to refine the query.

- **Filter Application:** Users can refine search results using various filters, including:

  - **Location** – Filters jobs based on preferred geographical area.
  - **Salary Range** – Displays jobs within a specified salary bracket.
  - **Job Type** – Filters results based on full-time, part-time, or remote jobs.
  - **Experience Level** – Matches job listings based on user experience.

- **Ranking Sorting:** The system assigns relevance scores to job listings using:

  - **Relevance Scoring** – Determines how well a job matches the user query.
  - **Machine Learning Ranking** – Uses AI models to prioritize jobs based on user history and preferences.
  - **Personalized Job Prioritization** – Customizes ranking to display the most suitable jobs first.

- **Data Retrieval Integration:** The system fetches job listings from:

  - **Internal Job Database** – Retrieves job postings stored in the system.
  - **External Job APIs** – Integrates with third-party job portals to aggregate listings dynamically.

- **Result Processing Display:** The processed and ranked job listings are presented to the user with options to:

  - **View Job Details** – Get in-depth information about job requirements.
  - **Save Job for Later** – Bookmark jobs for future reference.
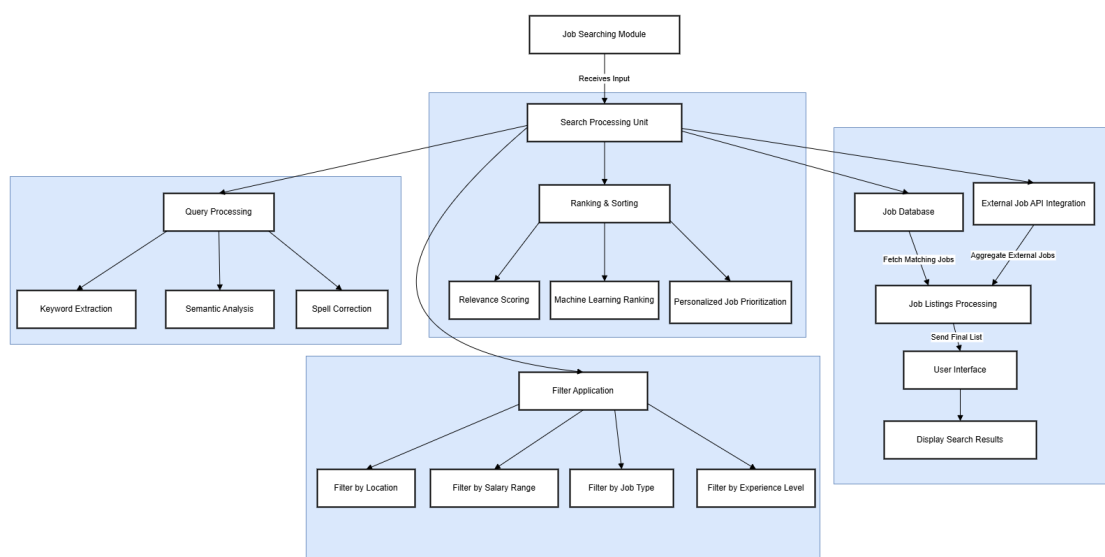 - **Apply for Job** – Redirects users to the application submission page.

Figure 2.2: Job Searching System Architecture

2.0.3 User Profile Management Module

This module is responsible for managing user data, including authentication, profile creation, CV upload, data extraction, and integration with other system components.

Process Flow

- **User Registration and Authentication:** Users register/login, and their credentials are verified through the authentication server.

- **Profile Creation and Management:** Users enter their personal details, set preferences, and update their profile information.

- **CV Upload and Processing:** If a user uploads a CV, the system processes it using an NLP-based parser (e.g., spaCy, NLTK) to extract structured data (skills, experience, education).

- **Data Storage and Integration:** Extracted data and user-entered information are stored securely in the User Profile Database.

- **Profile Update and Preference Management:** Users can edit their profile details and set job preferences, which are dynamically updated in the database.

- **External System Integration:** The stored user data is shared with the Job Recommendation and Job Searching modules to generate personalized job suggestions.
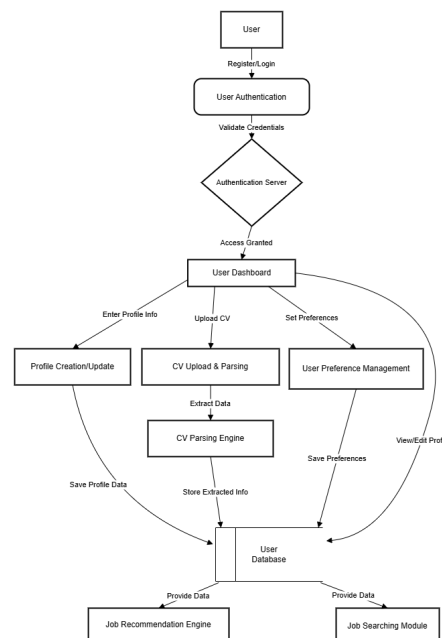

Figure 2.3: User Profile Management System Architecture

Chapter 3

Discussion

The Low-Level Architecture Design of the *Next-Gen Job Recommendation System (NGJRS)* provides a structured, detailed view of the core functionalities and their interconnections. By breaking down the system into modular components such as **Job Recommendation, Job Searching, and User Profile Management,** we ensure a scalable, efficient, and maintainable architecture.

The Job Recommendation module utilizes Natural Language Processing (NLP) and Machine Learning (ML) to analyze user profiles and suggest relevant jobs. It incorporates word embeddings, user preferences, and real-time job updates to provide personalized recommendations. The Job Searching module follows a structured query and ranking approach, allowing users to search and filter job listings based on various parameters. The User Profile Management module enables users to either generate a CV from structured data or extract relevant information from an uploaded resume, enhancing personalization.

This modular approach ensures that each component functions independently yet collaboratively, allowing for future improvements and expansions. The low-level architecture diagrams depict the step-by-step breakdown of each operation, providing clear insights into data flow, interaction between modules, and system processing.

References

[1] GeeksforGeeks. Low-level-design-diagram, 2023. Accessed: 2024-10-17.

[2] IEEE Computer Society. Low level diagram, 2023. Accessed: 2024-10-05.

[3] lucidchart. architectural-diagram, 2023. Accessed: 2024-10-17.

[1] [2] [3]