22AIE201 FUNDAMENTALS OF AI
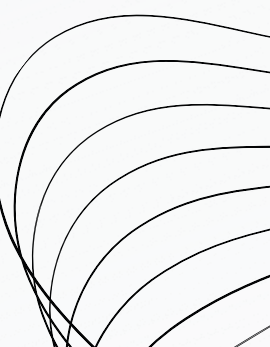
JOB RECOMMENDATION SYSTEM USING ML

DEPARTMENT OF ARTIFICIAL INTELLIGENCE
AMRITA VISHWA VIDYAPEETHAM, COIMBATORE

BATCH:- C-12

TEAM MEMBERS

Avvaru Bharadwaj eswar-Cb.sc.u4aie23210

Divagar.s   -Cb.sc.u4aie23223

Harish s.s   -Cb.sc.u4aie23230

G Harshavardhan   -Cb.sc.u4aie23226

INTRODUCTION

The "Job Recommendation System using ML" aims to assist job seekers in finding relevant job opportunities and recruiters in identifying suitable candidates. By leveraging machine learning techniques, such as natural language processing (NLP) and text similarity models, the system analyzes job descriptions and CVs to recommend the best matches. It uses machine learning algorithms to extract key skills from resumes, compute similarity scores, and provide real-time job recommendations from both static datasets and external job APIs, ensuring a seamless and personalized user experience.

OVERVIEW

The Job Recommendation System aims to streamline the job search process for candidates by utilizing advanced Natural Language Processing (NLP) techniques. The system extracts relevant skills from uploaded resumes and compares them to job descriptions in a database. By employing methods such as TF-IDF vectorization and cosine similarity, it identifies the best job matches for users. This automated approach enhances the efficiency of job searches, ensuring candidates receive personalized recommendations that align with their qualifications and career aspirations.

RECOMMENDATION SYSTEM

1.**User Interaction**: Job seekers upload CVs, recruiters input job descriptions to find matching candidates.

2.**Text Extraction and Preprocessing**: CV text is extracted and preprocessed (removal of stopwords, lemmatization).

3.**Skill Extraction**: Key skills are extracted from CVs using NLP models.

4.**Vectorization:** Job descriptions and CVs are converted into numerical vectors using TF-IDF.

5. **Similarity Computation:** Cosine similarity is used to measure the match between job descriptions and CVs.

6.**Job Recommendations:** Static job listings and real-time job data are recommended based on skill matching.

7.**CV Recommendations for Recruiters:** Top CVs are ranked for recruiters based on similarity to the job description.

8.**Interactive Outputs:** Job seekers get job recommendations, and recruiters receive candidate suggestions with similarity scores.

TF-IDF (TERM FREQUENCY-INVERSE DOCUMENT FREQUENCY)

TF-IDF (Term Frequency-Inverse Document Frequency) is a numerical statistic used to evaluate the importance of a word in a document relative to a collection of documents.

1. **Term Frequency (TF):** Measures how frequently a term appears in a document.
2. **Inverse Document Frequency (IDF):** Reduces the weight of common terms by assigning lower importance to words that appear in many documents.

TF-IDF is commonly used in information retrieval and text mining to represent documents in a way that highlights the most relevant terms while downplaying common ones.

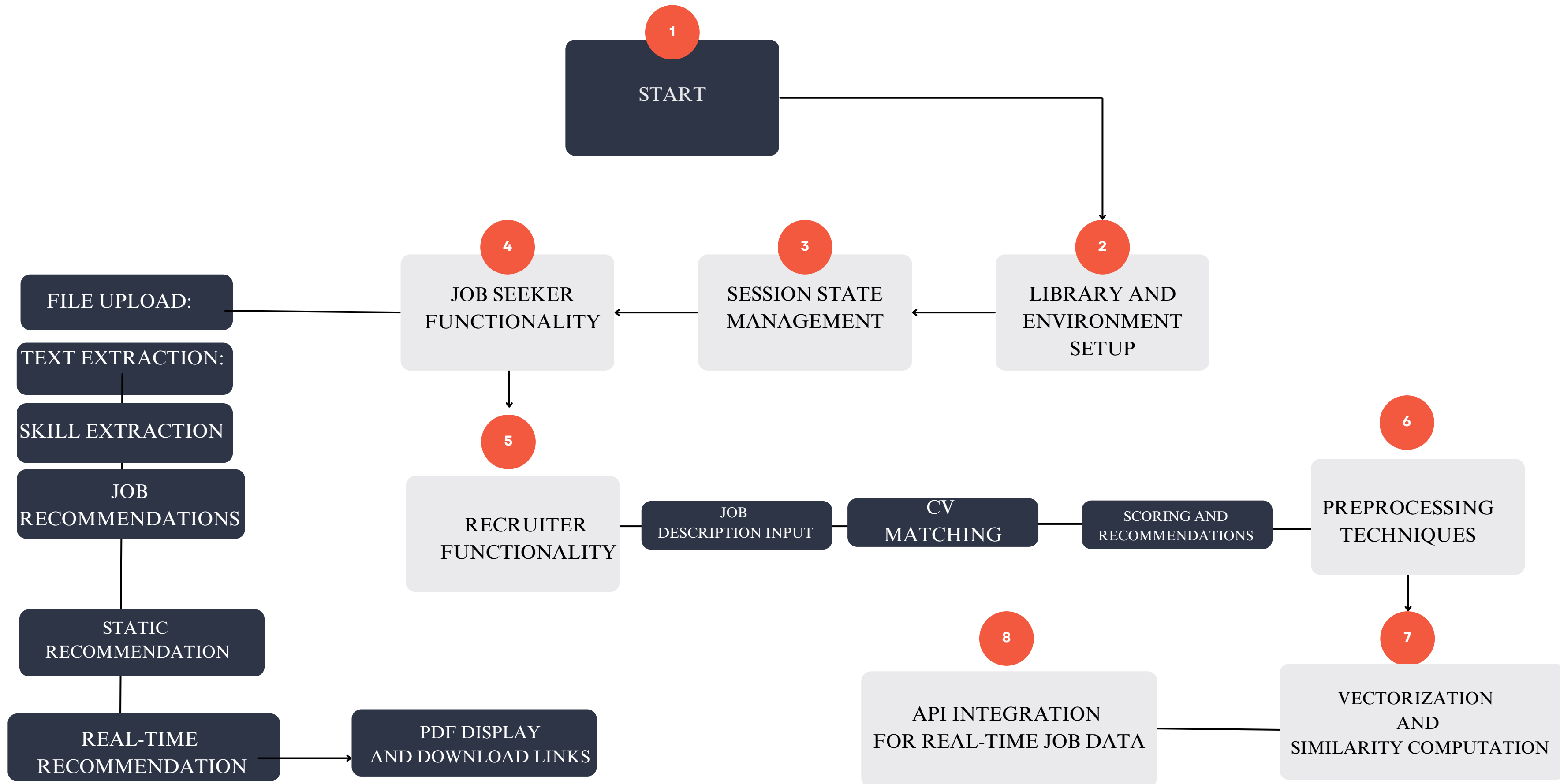| sentence 1	earth is the third planet from the sun			
sentence 2	Jupiter is the largest planet			
Word	TF (Sentence 1)	TF (Sentence 2)	IDF	TF*IDF (sentence 1)	TF*IDF (Sentence 2)
---	---	---	---	---	---
earth	1/8	0	log(2/1)=0	0.0375	0
is	1/8	1/5	log(2/2)=0	0	0
the	2/8	1/5	log(2/2)=0	0	0
third	1/8	0	log(2/1)=0.3	0.0375	0
planet	1/8	1/5	log(2/2)=0	0	0
from	0	0	log(2/1)=0.3	0	0
sun	1/8	0	log(2/1)=0.3	0.0375	0
largest	0	1/5	log(2/1)=0.3	0	0.06
Jupiter	0	1/5	log(2/1)=0.3	0	0.06

COSINE SIMILARITY

It is used to compare the TF-IDF vectors of the resume skills and job descriptions. This metric measures how similar the user's resume is to the job descriptions based on their skill sets.

The jobs with the highest similarity scores are recommended to the user. Those are displayed, showing details such as the job title, company name, and est. salary.
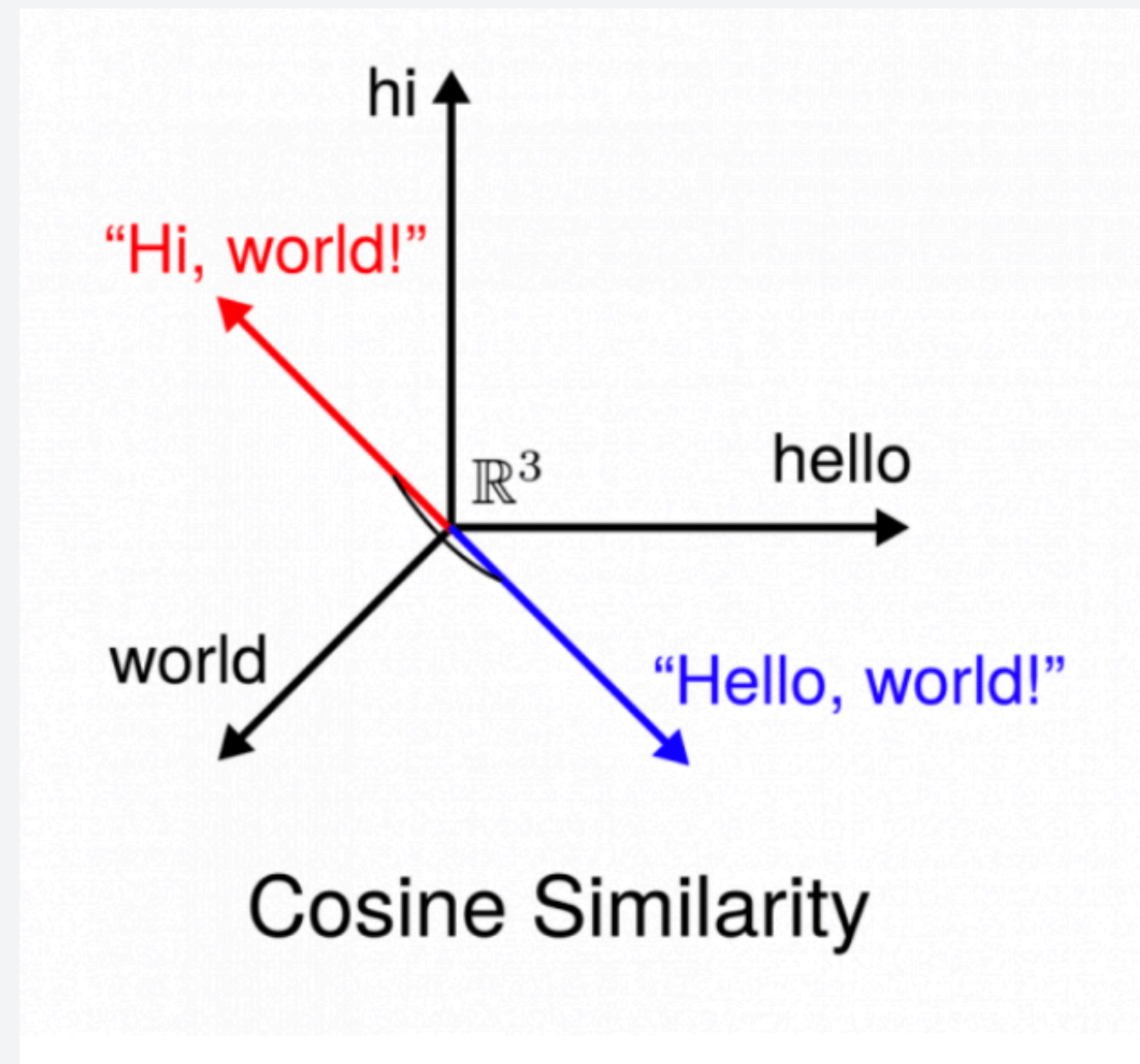
LIBRARIES USED

- streamlit: For building the web application interface.
- pandas: For data handling and CSV operations.
- requests: To make HTTP requests (e.g., Jooble API).
- scikit-learn: Specifically `TfidfVectorizer` and `cosine_similarity` for text matching.
- fitz (PyMuPDF): For extracting text from PDF files.
- spacy: For NLP tasks, like extracting skills.
- nltk: Used for text preprocessing, including:
- stopwords
- word_tokenize
- WordNetLemmatizer
- base64: For encoding files for download.
- io: To manage file input and output.

METHODOLOGY

1 START

2 LIBRARY AND ENVIRONMENT SETUP

3 SESSION STATE MANAGEMENT

4 JOB SEEKER FUNCTIONALITY

FILE UPLOAD:

TEXT EXTRACTION:

SKILL EXTRACTION

JOB RECOMMENDATIONS

STATIC RECOMMENDATION

REAL-TIME RECOMMENDATION

PDF DISPLAY AND DOWNLOAD LINKS

5 RECRUITER FUNCTIONALITY

JOB DESCRIPTION INPUT

CV MATCHING

SCORING AND RECOMMENDATIONS

6 PREPROCESSING TECHNIQUES

7 VECTORIZATION AND SIMILARITY COMPUTATION

8 API INTEGRATION FOR REAL-TIME JOB DATA

JOB SEEKER SECTION

• The Job Seeker section is designed to make it easy for users to upload their CVs, extract relevant skills automatically, and receive job recommendations tailored to their qualifications.

• The interface offers two types of job recommendations: dataset-based and real-time listings

STREAMLIT INTERFACE FOR JOB SEEKERS:

- **CV Upload Section:** Displays how users can upload multiple PDF files, with each file processed to show extracted skills.

- **Dataset-Based Recommendations**: Shows the list of job recommendations based on the dataset, with job details such as title, company, and salary.

- **Real-Time Recommendations from Jooble API:** Captures real-time job listings fetched via the Jooble API, highlighting features like job title, company, location, and a direct application link.

RECRUITER SECTION

•The Recruiter section of the application is to efficiently manage and evaluate candidate CVs by matching job descriptions with uploaded CVs, scoring candidates based on relevance, and recommandations them for a more streamlined hiring process.

STREAMLIT INTERFACE FOR RECRUITERS

- **Job Description Input Area:** Captures the job description entry field and the ability to trigger the recommendation process.
- **Recommended CV List:** Shows the display of top-matched CVs with similarity scores, extracted skills, and an option to download CV files.
- **Function :** create_download_link is used to convert CV data to a string for a download link, making it accessible for recruiters.

DATA COLLECTION

Data Collection:
1. CVs (Resumes):
   - Users (Job Seekers) upload their CVs in PDF format.
   - The CVs contain personal information, job experiences, and most importantly, skills.
   - The CV text is extracted using PyMuPDF (fitz), which converts the PDF content into text.
2. Job Descriptions:
   - Job Seekers and Recruiters provide job descriptions.
   - Job Listings data is also fetched from a CSV file and Jooble API for real-time job recommendations.

PREPROSSESING:

1. CV Preprocessing:

Text Extraction: PDF content is extracted to plain text.

Skill Extraction: Using spaCy, the extracted text is processed to identify potential skills (nouns and proper nouns) from the CV.

2. Job Description Preprocessing:

Text Tokenization: The job description is converted to lowercase and tokenized into words.

Stopwords Removal: Unnecessary words (like "the", "and") are removed using NLTK stopwords.

Lemmatization: Words are reduced to their root form (e.g., "running" becomes "run") using WordNet Lemmatizer from NLTK.

3. TF-IDF Vectorization:

TF-IDF (Term Frequency-Inverse Document Frequency) is applied to convert both CV and job description text into numerical vectors for   comparison.

Cosine Similarity is used to calculate how similar the job description is to each CV based on these vectors.

FUTURE IMPLEMENTATION

- Incorporating Deep Learning for Personalized Recommendations
- Incorporation of Candidate Feedback and Interaction Data
- Skills Gap Analysis and Training Recommendations
- Predictive Analytics for Career Path Recommendations

CONCLUSION

In an evolving digital and competitive job market, using our model for job matching presents a useful solution to streamline recruitment and improve outcomes for both job seekers ,employers and recruiters. The job recommendation system developed in this project represents a fundamental yet impactful application of machine learning and natural language processing techniques. By implementing TFIDF vectorization and cosine similarity, the system can analyze and match resumes to job descriptions with notable accuracy, significantly reducing the time and effort required for initial candidate screening.