LLM PROJECT

Garv Makkar : 2021530

Shubham Pal : 2021564

Sachin Sharma : 2021560

Rishav Raj : 2021556

Parth Kaushal : 2021548

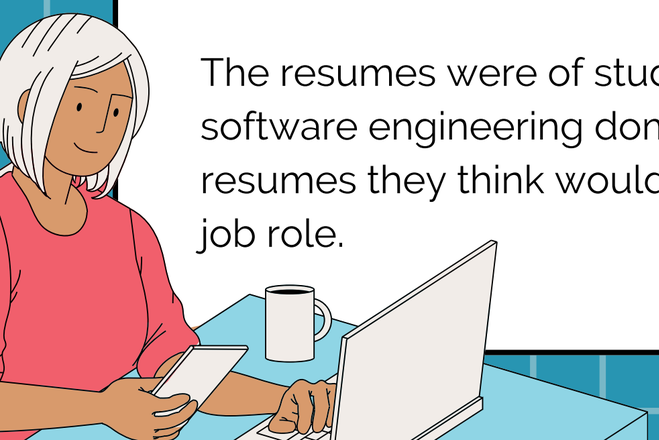
ABOUT DATASET

Dataset Summary

The dataset consists of various resumes and job description from a variety of field but mainly consisting of software engineer roles.

The job description is a document posted by the hiring company for the necessary requirements it seeks to see in a potential candidate. It consists of various skills, qualifications and soft skills which it wants to see in a candidate. These skills are mainly what are required for the resume of a candidate to get a good score on the ATS.

The resumes were of students who were related to software engineering domain and had created the resumes they think would be the best for a particular job role.



Feature Description

Role, skills and eligibility were the main three divisions we could see in every job description and thus these sections were extracted from the job description.

The main sections of the resume included person_details, education,skills,work_ex,projects,extras

PROBLEM STATEMENT & IMPORTANCE

Problem Statement

1. Objective: Develop an AI-powered tool to assist users in creating and enhancing resumes/CVs for specific job roles and overall career development.
2. Features:
    - Analyze user profiles, resumes, and job descriptions to assess strengths, weaknesses, and improvement areas.
    - Provide tailored recommendations based on industry-specific insights.
3. Outcome: Help users increase their chances of securing desired job opportunities.
4. Future Vision:
    - Expand the tool to offer comprehensive career guidance.
    - Analyze aspirations and qualifications to identify gaps and provide actionable recommendations for achieving career goals.

DATA PREPROCESSING

Key Preprocessing Steps

1. Text Extraction

- PDF to text conversion using PyPDF2
- Handles multiple pages
- Raw text extraction for further processing

2. Text Embedding

- Uses SentenceTransformer model
- Converts text to numerical vectors
- Enables semantic similarity matching

3. Information Extraction

- Structured data extraction using LLM
- Entity recognition for personal details
- Pattern matching for skills and qualifications

4. Data Structuring

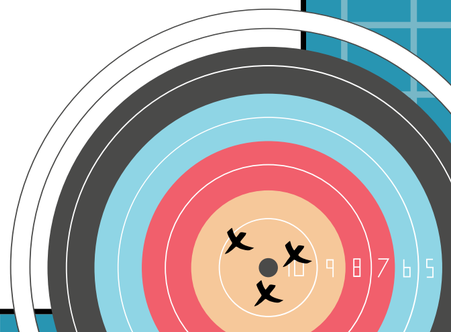
- JSON format conversion
- Hierarchical data organization
- Standardized output format

Resume Analysis

Person Details
Education History
Skills Inventory
Work Experience
Projects
Additional Information

Job Description Analysis

Role Classification
Required Skills
Eligibility Criteria
Similarity Scores



DATA PREPROCESSING

Resume Input and Tokenization

- Resumes in PDF format have been converted to JSON. Each JSON file includes detailed sections like personal information, education, skills, work experience, projects, and additional details.
- Converting PDFs to JSON ensures a structured, consistent schema, which is more effective than using plain text (.txt).
- The Llama 3.1 70B model was employed for this task, leveraging its superior Named Entity Recognition (NER) capabilities, structured data comprehension, adaptability, and high accuracy in parsing complex content.

Token Cleaning and Preprocessing

- Standard cleaning techniques were used to remove stop words and irrelevant tokens, preparing the data for efficient analysis.

Section Identification and Tagging

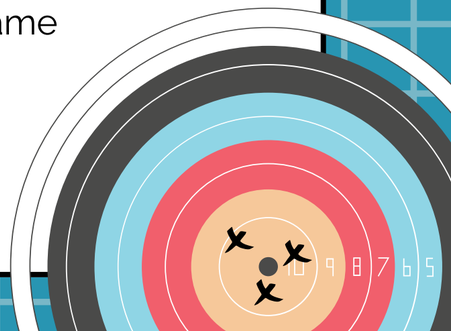
- Key resume sections were identified and labeled with tags such as <Work Experience>, <Achievements>, and <Skills>.
- This structured format aids in more effective downstream analysis and model tuning.

Skill Extraction and Cleaning

- Relevant Skill Extraction: Used keyword matching to identify skills from resumes.
- Skill Cleaning and Normalization: Ensured consistency by removing stop words and normalizing variations.

Information Extraction from Job Descriptions (JD)

- JSON Conversion: Job descriptions were converted to JSON using the Llama 3.1 70B model.
- Extracted Sections:
- Role: Identified using cosine similarity with a list of roles.
- Required Skills: Extracted with the Llama model.
- Eligibility Criterion: Extracted using the same model.



APPROACH 1 : ZERO SHOT PROMPTING (BASELINE)

Approach

Different sections of resume were separately parsed onto LLMs at different stages along with the context of JD to get inferences of enhancement separately and sequentially.

Zero-Shot Prompting and Skill Tuning

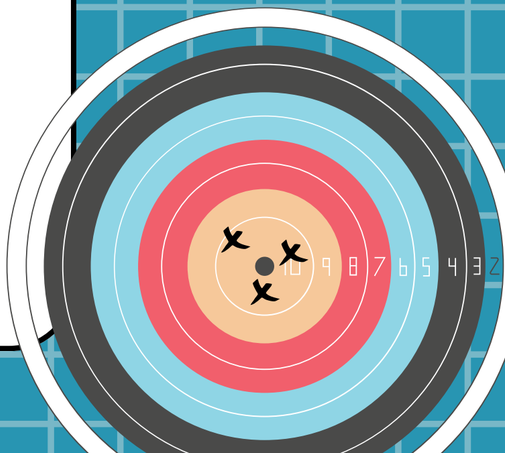
Model Selection: Multiple language models, including Llama3 8b, Gemma2 9b, and Gemini, were employed for zero-shot prompting.

Prompt Engineering

Prompts were carefully crafted to guide the models in refining the extracted skills based on the provided job descriptions. Prompt engineering is the main driving factor here and therefore a good prompt was made.

Skill Tuning

The models generated improved versions of the extracted skills by leveraging their understanding of the job requirements and the con text provided in the prompts



APPROACH 2 : RAG

RAG System

- A Retrieval-Augmented Generation system is employed to extract relevant skills from job descriptions and suggest improvements to resumes.

Implementation

- Model: Google Generative AI Gemini-1.5-pro-latest for content generation.
- Libraries: LangChain for document processing and FAISS for efficient data retrieval.
- Knowledge Base: An extensive collection of skills in various domains, represented as PDFs.
- Vector Store: Text chunks are converted into embeddings using Hugging Face's sentence-transformers model and stored in FAISS for quick retrieval.

Process

1. Skill Extraction: Relevant skills are extracted from job descriptions.
2. Resume Analysis: Candidate's resume is analyzed to identify existing skills.
3. Skill Matching: Skills from the job description are compared to those in the resume.
4. Resume Enhancement: Suggestions are provided to enhance the resume, such as adding relevant keywords or highlighting specific experiences.

APPROACH 2 : RAG

Retrieval Mechanism

- Retrieves the top 3 most relevant text chunks from the vector store.
- Ensures high relevance in the content passed to the Gemini model.

System Workflow

1. Input: Takes resume and job description JSON files.
2. RAG Query: Queries the RAG system for relevant industry knowledge.
    - Resume Enhancement:Enhances sections like skills, experience, projects, and extras.
    - Incorporates job requirements and industry best practices.
3. Output: Saves the enhanced resume with source references.
    -

APPROACH 3 : DPO

We sampled a set of 30 resumes for the role of Software Development Engineer (SDE) and labeled them on a scale of 0 to 5.

After labeling, we fine-tuned the model using the Direct Preference Optimization (DPO) technique. However, due to memory constraints, we were unable to fine-tune the model or run the code. Therefore we were unable to obtain any significant results with it.

Despite these limitations, we decided to adopt this method since the general concept of DPO is good and it will for sure help in the optimization of resumes and improve the ATS score in real life.

RESULT CALCULATION

1. Skill Matching:
    - Parses and normalizes skills.
    - Uses fuzzy matching for flexibility.
    - Calculates a score based on skill overlap.
2. Grammar:
    - Evaluates writing quality using LanguageTool.
3. Resume-Job Description Alignment:
    - Keyword analysis for relevant terms.
    - Text similarity using TF-IDF.
    - Semantic matching using BERT embeddings.

Final Score Calculation

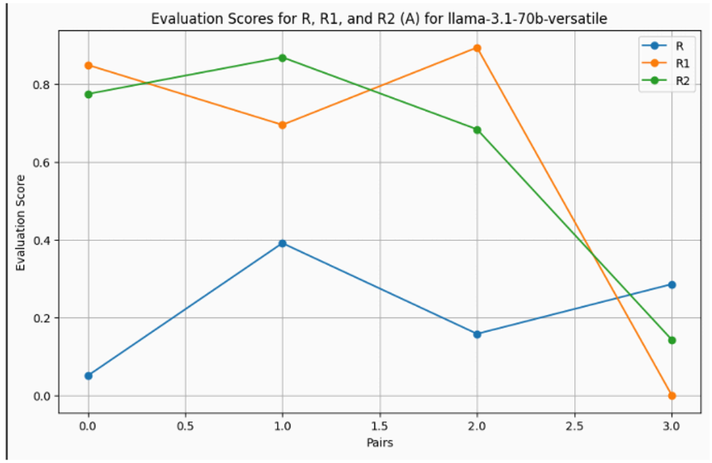
- Semantic Understanding: 40%
- Keyword Matching: 30%
- Content Similarity: 30%

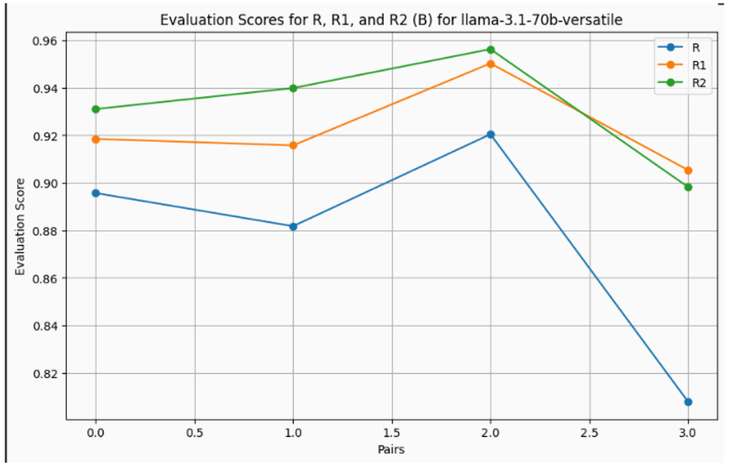
RESULTS

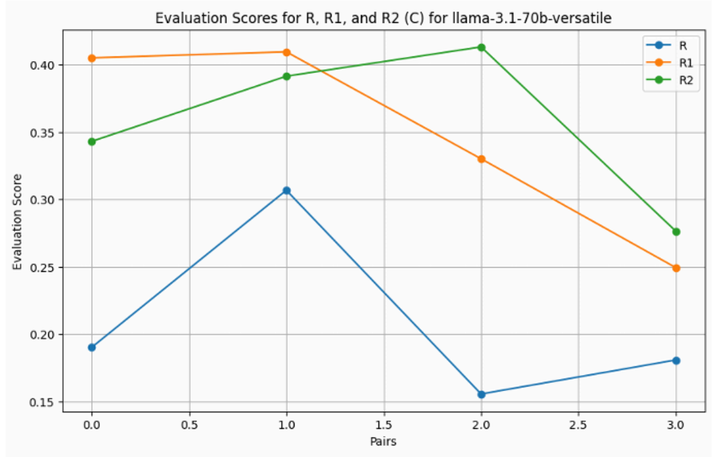
Three types of evaluation scores (A, B, C) have been calculated for each version

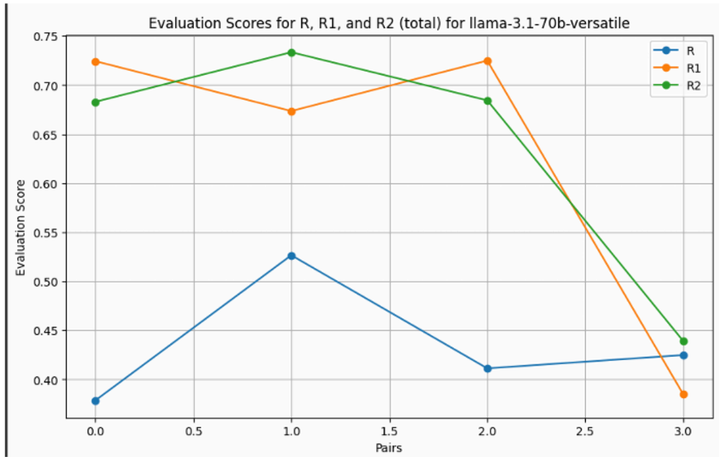
Skills matching (A)
Grammar quality (B)
Overall matching (C)

and their respective graphs have been plotted along with a
Combined total score





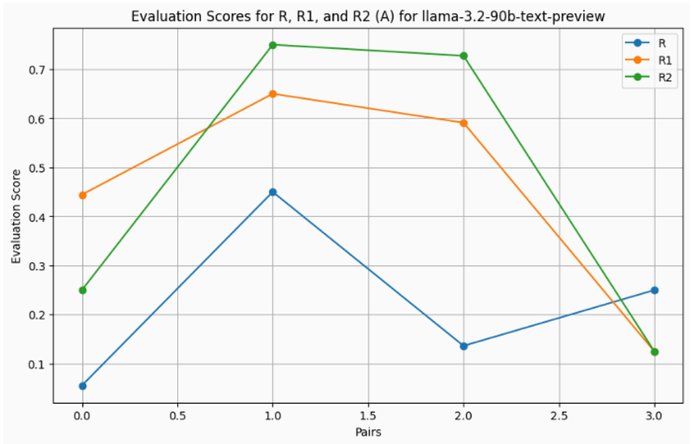


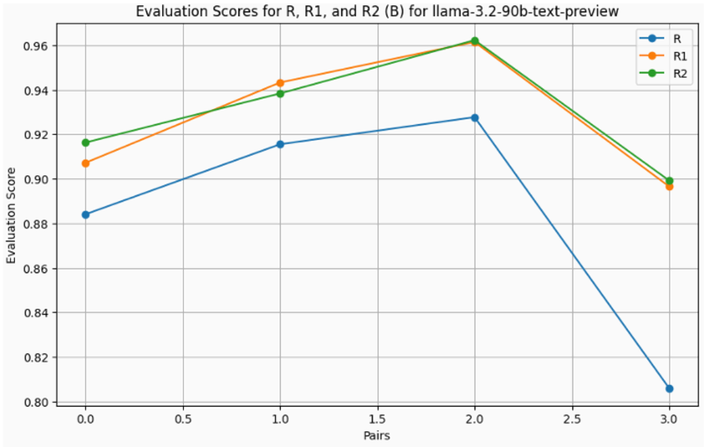


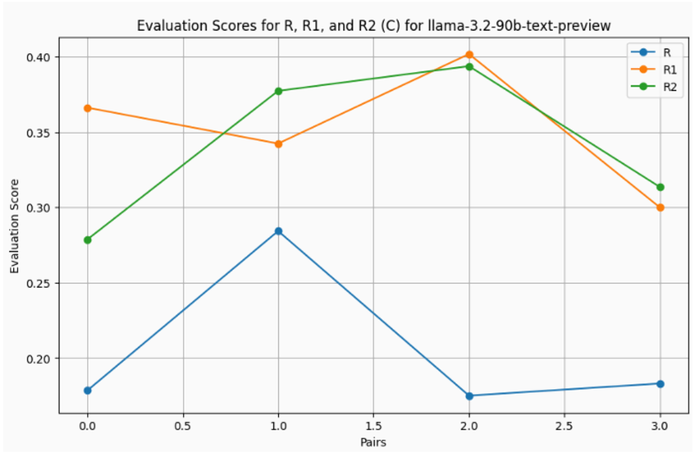
Results for llama-3.1-70b-versatile

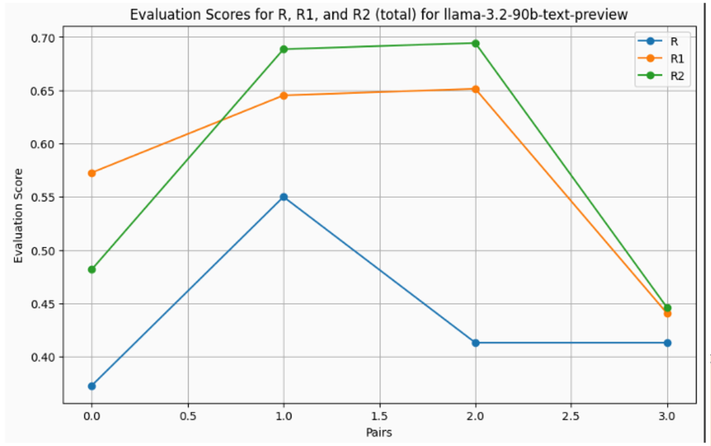
Average Evaluation Score for R: 0.4353264725093886
Average Evaluation Score for R1: 0.6270193125967944
Average Evaluation Score for R2: 0.6350886409211411

RESULTS









Results for llama-3.2-90b-text-preview

Average Evaluation Score for R: 0.4371544996145654

Average Evaluation Score for R1: 0.5774336770940104

Average Evaluation Score for R2: 0.5776433689482731

THANK YOU!