Report on Resume Ranking Implementation

Approach:

1. The overall idea started by trying out various models and trying out which models and vectorstore embeddings tend towards better results and better accuracy.
2. After carrying out multiple trials FAISS vectorstore yielded better results and hence I went with the idea to FAISS vectorstore.
3. Now every resume was split into multiple chunks and hence one challenge was to decide which heuristic would beb the most appropriate.Right now I have applied mean of all the similarity scores over all chunks to get an overall relevance score for a particular resume.But even though I havent implemented it right now , I investigated a 40% semantic similarity ,30% keyword matchings and 30% to a weighted average of the cosine similarity score of particular projects and experiences can be given a try and would be implementing over time.
4. A important reason for including keyword matchings was sometimes semantic similarity returned a higher score for people with complete different tech stack but similar experience, whereas people with same tech but different experiences and projects were having lesser relevancy score.
5. It was also seen during debugging sessions when same 3 resumes were fed to a OpenAI model , with time it the relevancy scores started converging to a single value with multiple runs.

Implementation :

1. There is a engine module which takes job_role,job_description, and a list of resumes as input and returns a dictionary of recommended_candidates with all the details of every candidate and also the list of non_recommended_candiates with the score and path.
2. The engine first starts by evaluating every resume one by one with help of FAISS vectorstore and for every splitted_doc we receive a score of which avg is taken to get the final relevancy score.
3. Now a similarity threshold has been set to 0.5 above which the resumes will be recommended or else they would be non_recommended.
4. After every recommended_resume is used and with llama_index we get all the details related to resume.
5. We then use cosine similarity to get relevancy for every project of the resume and after all the processing the resumes along with details are returned as response.