

CSCE 5290.003 Natural Language Processing

Group 3 Project Proposal

Job Description Based Resume Matcher

1. Project Title and Team Members

Job Description Based Resume Matcher

Suhas Siddarajgari Tellatakula	11626111
Vamshi Telukuntla	11618397
Srikanth Mamillapalli	11551359
Sairohithvarma Kantem	11606743

GitHub link: <https://github.com/Vams98>

2. Goals and Objectives

- **Motivation:**

The number of candidates planning to apply for jobs online every year is increasing rapidly. But unfortunately, only few of them manage to secure their dream job in dream company/firm. This is because most of the candidates' resumes will not get picked in the first place. Students frequently apply to numerous jobs with single resume irrespective of job description. There is need to be aware of creating job description tailored resumes. So, we are trying to build a solution which can analyse and understand the job descriptions and selectively picks the resumes with rich and relevant keywords.

- **Significance:**

As companies receive plethora of applications for one job opening, it will be very difficult for the companies to go through each resume and filter the candidates. Hence, they employ artificial intelligence-based resume pickers which look for specific keywords in the resumes. Creating job description based tailored resumes is one of the crucial steps in applying for a job. A recent survey indicates that 34% of people do not know how to prepare unique resumes based on job description. Hence, this tool can be extremely helpful for

candidates to check their resume relevancy to the applied job and increase their chances of landing dream job.

- **Objectives:**

With this project, we are aiming to solve above problem by taking advantage of the Natural Language Processing methodologies employing *text summarization, label classification and keyword extraction* techniques. The objectives of the project are as follows:

1. To summarize multiple paragraph job/role descriptions into a simple yet fully understandable output by executing different text summarization techniques like frequency-based method or Machine learning based summarizer libraries like TextRankSummarizer, LexRankSummarizer, LSSummarizer (Latent Semantic Analyzer), Luhn Summarizer.
2. To classify the jobs/roles under few tags/groups based on the above generated short summary employing concept/tags classification mechanisms.
3. To identify and filter out the important keywords from a document based on different keyword extraction methods using python libraries like spacy, yake, rake-nltk, genism.
4. To provide the relevancy or matching score between job descriptions and resumes based on a machine learning model built on the generated summary and keywords extracted from resumes.

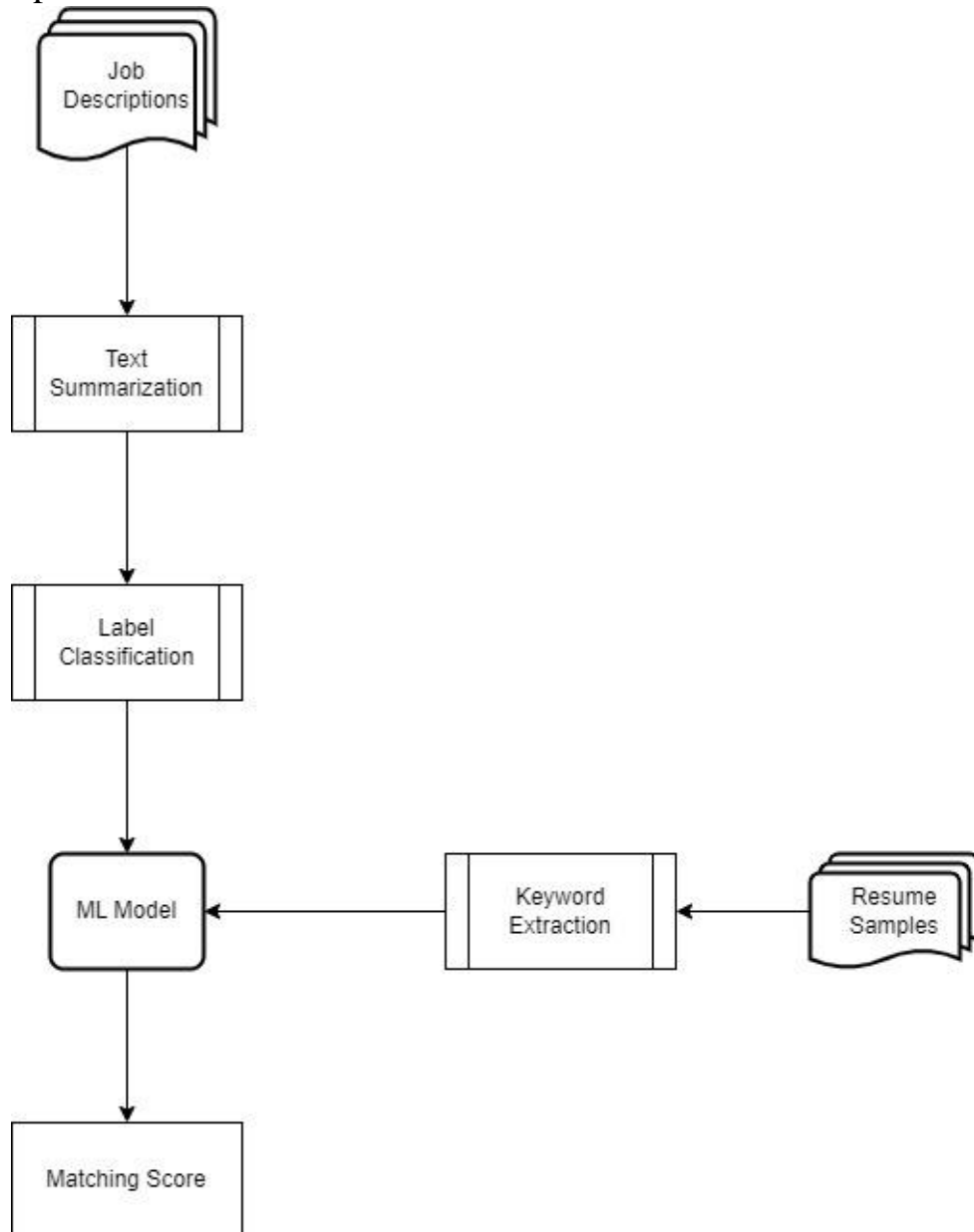
- **Features:**

Once the project is fully developed based on the above objectives, the solution is expected to have following features:

1. Tool provides a brief summary from a posted job description/role requirement.
2. Tool can extract important relevant keywords from a resume document.
3. Tool suggests best fit candidates' resumes to a given job description or generates matching score between job description and resume.

3. **Workflow**

As shown in the fig 1, we first perform text summarization on the job description datasets. This process is depicted in detailed in fig 2, where the text data is first split into sentences which are then converted into vectors which go through similarity matrix and produce sentence rankings and finally a summary. Once we have the generated summary, we run them through the label classification techniques and label classified job description summaries.



Resume - Job Description Matcher Workflow

Fig 1. Project workflow

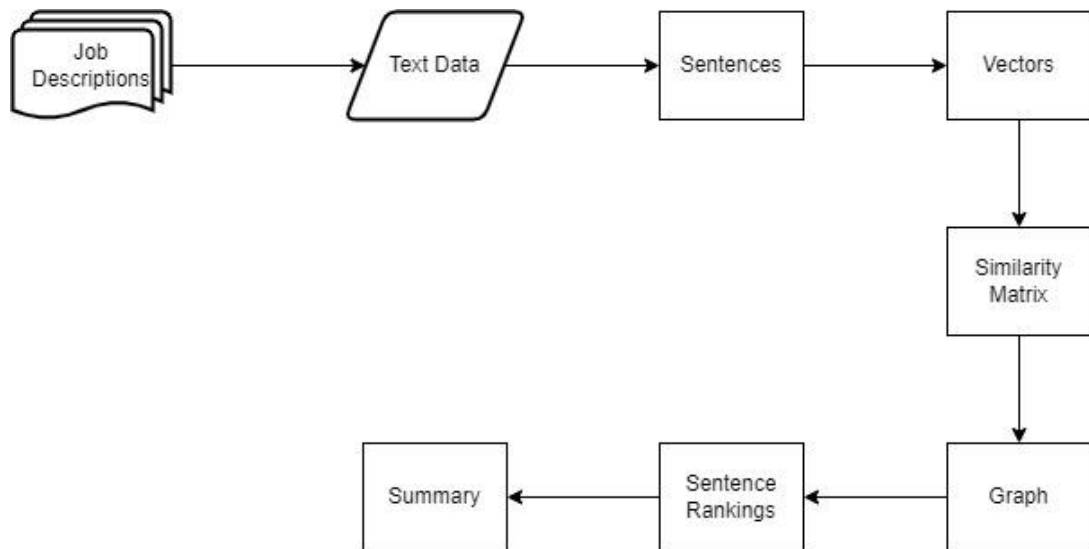


Fig 2. Text Summarization Process

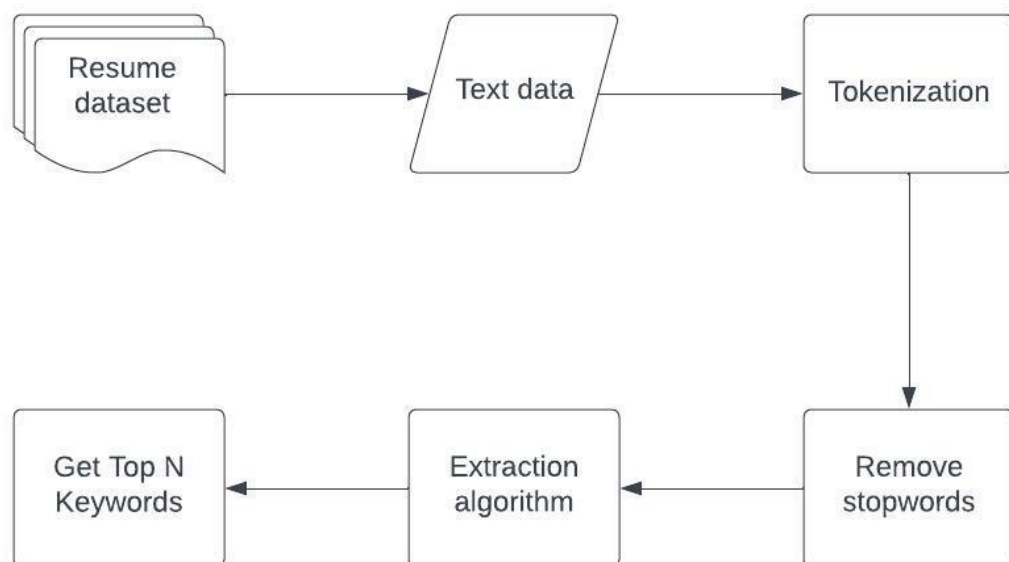


Fig 3. Keyword extraction Process

On the other side, we try out different keyword extraction techniques on resume dataset and extract the keywords from each resume. This process is depicted in fig 3. Finally, a regression or classification Machine Learning model takes these as input and finds out the matching score between them in terms of percentages.

4. **References**

- a) <https://www.kaggle.com/datasets/snehaanbhawal/resume-dataset>
- b) <https://www.kaggle.com/datasets/yusufolonade/data-science-job-postings-indeed-usa>
- c) <https://www.kaggle.com/datasets/shivamb/real-or-fake-fake-jobposting-prediction>
- d) <https://towardsdatascience.com/keyword-extraction-process-in-python-with-natural-language-processing-nlp-d769a9069d5c>
- e) <https://www.topcoder.com/thrive/articles/text-summarization-in-nlp>
- f) <https://www.analyticsvidhya.com/blog/2020/11/words-that-matter-a-simple-guide-to-keyword-extraction-in-python/>

5. **Contributions**

Sl.No.	Name	Contribution
1.	Suhas Siddarajgari Tellatakula	Documentation, Workflow, GitHub
2.	Vamshi Telukuntla	Documentation, Workflow, GitHub
3.	Srikanth Mamillapalli	Documentation, Goals and objectives
4.	Sairohithvarma Kantem	Documentation, Goals and objectives