**Goal 1: Extraction: Given a resume, we would like to extract name, email, programming skills, college & education.**

**Step 1:**

Resumes are a form of unstructured data and are found in different formats including pdf, doc or docx format. To work with resumes we need to convert them into text format for further procedure. So we can import pdfminer and doc2text modules. Pdfminer will help in dealing with pdf while doc2text will deal with docs files.

**Step 2:**

Then we can install Spacy, nltk and Regular Expression libraries to extract the relevant information from resume data. We will create separate functions for each section. For example, we can use spacy.matcher to extract names and re.compile and re.findall to extract phone numbers and emails from resumes.

**Step 3:**

For extraction of skills, we can create a skill dataset against which we can compare the skills in a particular resume. After loading that skill dataset, we can tokenize our data and compare it with the skill dataset. We can use stop words to further clear out results.

**Step 4:**

Finally, we can use re, nltk and spacy libraries to extract desired education information. We can first create a list of degrees and educational qualifications that resumes generally have and then will create a function that would first extract the relevant information.

**Goal 2 - Matching: Given a resume, we want to match the candidate with the most relevant Job.**

**Step 1:**

First I will scope this project and choose the most popular jobs as the basis for my analysis, most probably 8-10 different job types. I won’t include too many job applications as the model can get cluttered which might not give the best accuracy. So after settling for the fact, I will sparse data from websites like Indeed or Glassdoor using requests and bs4 (Beautifulsoup) library.

**Step 2:**

The next step would be data preprocessing and cleaning the data. The regular operations would be removing punctuation and capitalization, tokenization and stemming or lemmatization. The final task in this step would be to a vectorizer and convert data into an array format.

**Step 3:**

The next step would be to use topic modelling and build a classification algorithm. We can try different algorithms including decision tree classifier, random forest classifier, KNN, Adaboost classifier and so on. After creating a model we can give it a functional purpose that can predict which jobs the resume fit best with. Or, we can also extract the percent match by each job type, for example; it can predict the outcome 50% Data Scientist, 30% Data Analysis and so on.