Name: Alexis Collier

Email: colliera75@gmail.com

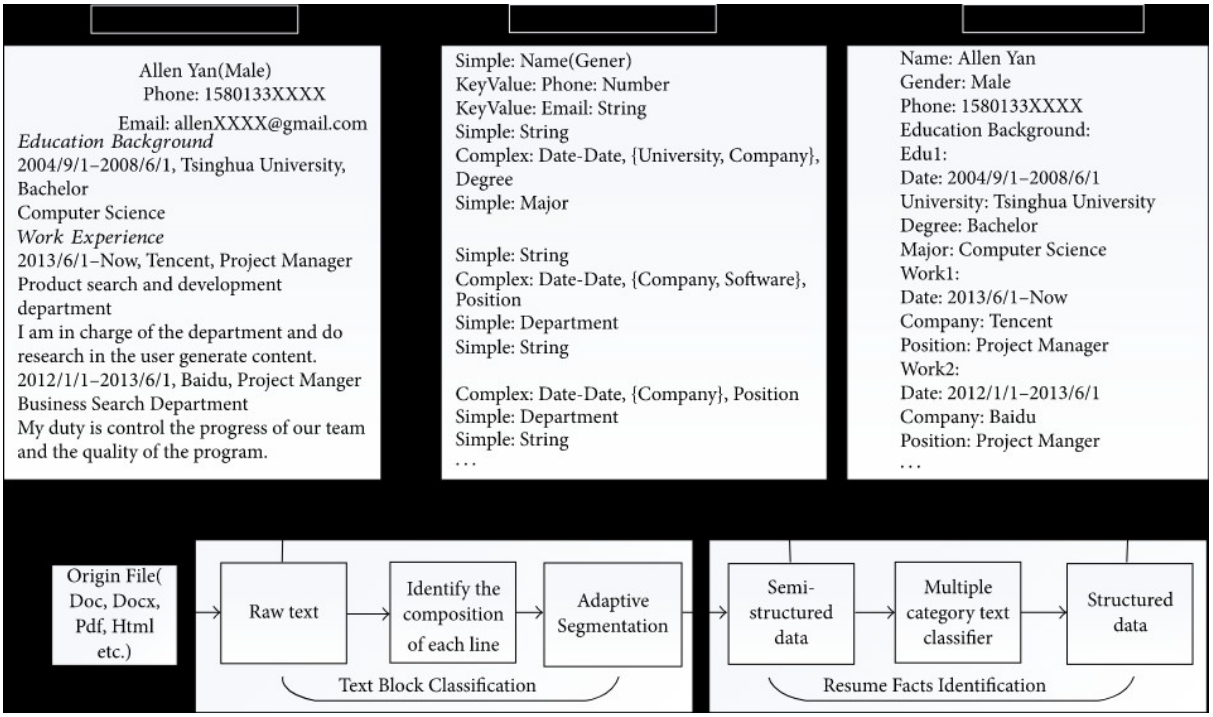
Country: United States

College: Fullstack Academy

Specialization: NLP

Problem description:

Resumes contain excess information irrelevant to the HR/authority, and they must manually process the resumes to shortlist the promising candidates. And thus, making the shortlisting task a herculean task for HR. Using the NER (Named Entity Recognition) model of NLP, this problem can be solved by finding and classifying the entities present in each resume into predefined classes such as person name, college name, academic information, relevant experiences, skill set, etc.



EDA:

To provide meaningful insights by analyzing the resume extraction dataset, I created a data frame containing the different labels of each unordered resume.

0	Companies worked at	oracle
1	Companies worked at	oracle
2	Companies worked at	oracle
3	Skills	languages: core java, go lang, data structures...
4	Companies worked at	oracle
...	...	...
3203	Degree	b- tech
3204	Designation	security analyst
3205	Companies worked at	infosys - career contour
3206	Designation	security analyst
3207	Name	pradeep kumar

3208 rows × 2 columns

I will be performing statistical analysis on each one of these elements except for the name, email, and designation:

```
print(df[0].unique())
```

```
['Companies worked at' 'Skills' 'Graduation Year' 'College Name' 'Degree'
 'Designation' 'Email Address' 'Location' 'Name' 'Years of Experience']
```

So, I split the data according to each label:

	0	text
6	Graduation Year	2012
9	Graduation Year	2012
56	Graduation Year	2016
59	Graduation Year	2018
70	Graduation Year	2009
...	...	...
3127	Graduation Year	2005
3130	Graduation Year	2013
3135	Graduation Year	2013
3136	Graduation Year	2013
3170	Graduation Year	2002

222 rows × 2 columns

0	Companies worked at	oracle
1	Companies worked at	oracle
2	Companies worked at	oracle
4	Companies worked at	oracle
10	Companies worked at	oracle
...	...	...
3186	Companies worked at	infosys bpo ltd
3189	Companies worked at	infosys bpo ltd
3192	Companies worked at	infosys bpo ltd
3194	Companies worked at	infosys bpo ltd
3205	Companies worked at	infosys - career contour

676 rows × 2 columns

	0	text
22	Location	bengaluru
35	Location	hyderabad
38	Location	hyderabad
39	Location	hyderabad
40	Location	hyderabad
...	...	...
3183	Location	bengaluru
3185	Location	bengaluru
3188	Location	bengaluru
3191	Location	bengaluru
3197	Location	bengaluru

381 rows × 2 columns

	0	text
3	Skills	languages: core java, go lang, data structures...
5	Skills	apex. (less than 1 year), data structures (3 y...
28	Skills	functional testing, blue prism, qtp
53	Skills	languages & technologies: python, r, sql, nos...
55	Skills	python (2 years), sql. (1 year), nosql (1 year...
...	...	...
3124	Skills	sap hana (4 years), sap ui5/fiori (4 years), a...
3155	Skills	data backup (1 year), exchange (1 year), lan (...
3163	Skills	auditing (less than 1 year), cfa (less than 1 ...
3169	Skills	excel (10+ years), operations (7 years), proje...
3201	Skills	splunk, network security, arc sight (2 years),...

417 rows × 2 columns

I also prepared my data frame to make it easier to use with the matplotlib and the seaborn libraries.

Transforming all the text into lowercase, removing unnecessary spaces, changing dates into numeric variables, and removing unnecessary words.

```
0                          oracle
1                          oracle
2                          oracle
4                          oracle
10                         oracle
                    ...
3186              infosys bpo ltd
3189              infosys bpo ltd
3192              infosys bpo ltd
3194              infosys bpo ltd
3205    infosys - career contour
Name: text, Length: 676, dtype: object
```

```
[ ] stopwords = ['what', 'who', 'is', 'a', 'at', 'is', 'he','of','university','college','public','private','school','institute','academy']
    L4=CollegeN["text"]
    L=list(L4)
    H=[]
    L3=[]
    for i in range(291):
      H=L[i].split()
      L1 = [word for word in H if word.lower() not in stopwords]
      L2= ' '.join(L1)
      L3.append(L2)

    print(L3)
```

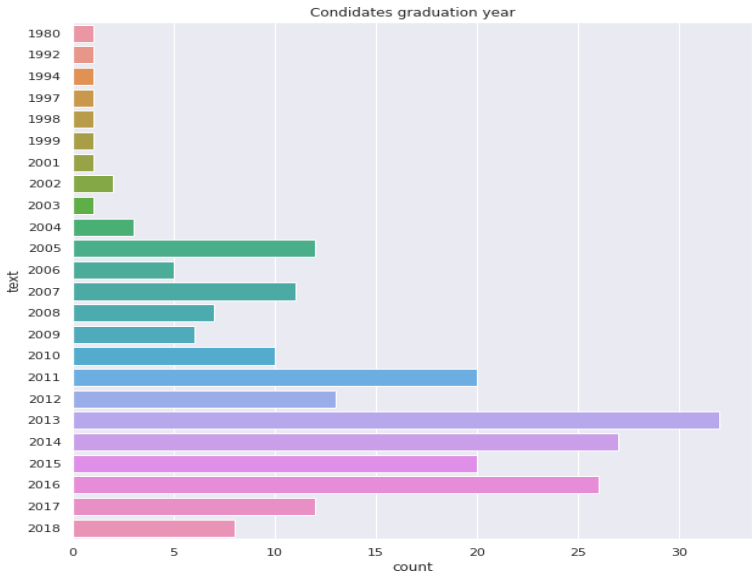
```
['adithya', 'osmania', 'osmania', 'manipal', 'manipal', '', 'birla', 'rashtriya military bangalore', 'rashtriya military bangalore', 'army', 'acharya chembur',
```

```
Gradyear["text"]=pd. to_numeric(Gradyear["text"])
type(Gradyear['text'][6])
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py
A value is trying to be set on a copy of a slice from a Data
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.
  """Entry point for launching an IPython kernel.
numpy.int64
```

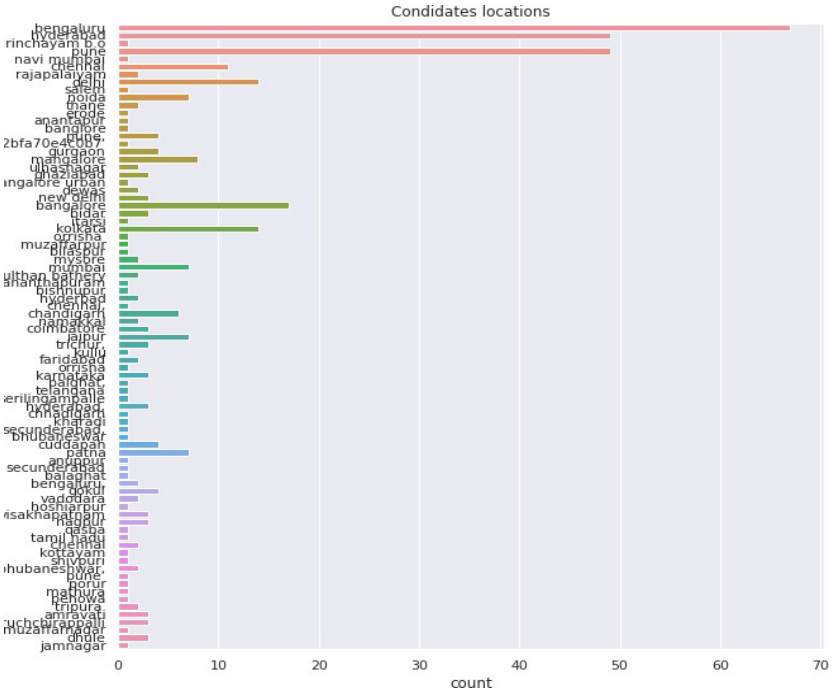
Starting with the graduation year:



We can see that most of the candidates graduated after 2005.

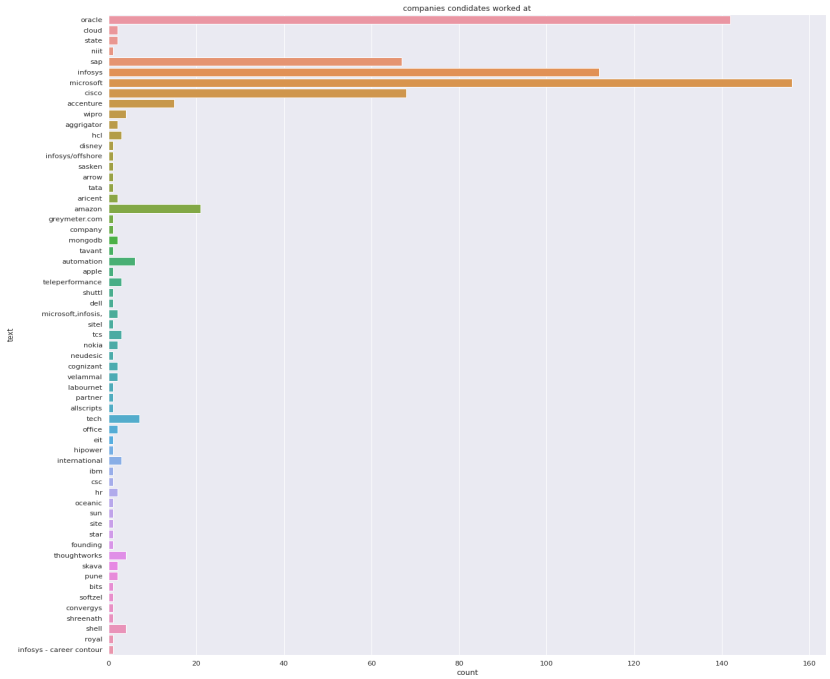
16% of the candidates graduated in 2013 only, and 50% graduated between 2016 and 2013. No candidates graduated after 2018.

Next, I have candidates' locations:

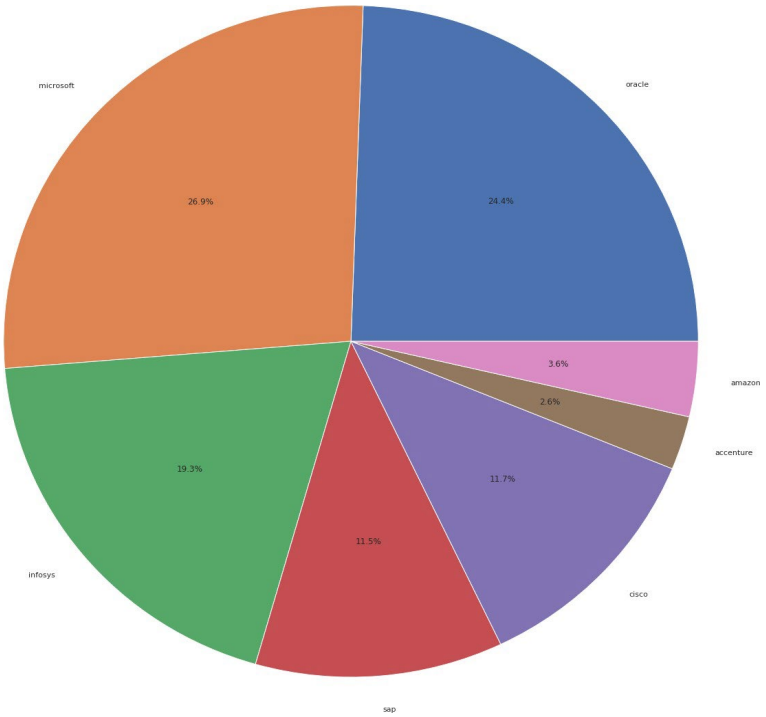


Most candidates are in Bengaluru (67 out of 200), Hyderabad and Pune (49 in each).

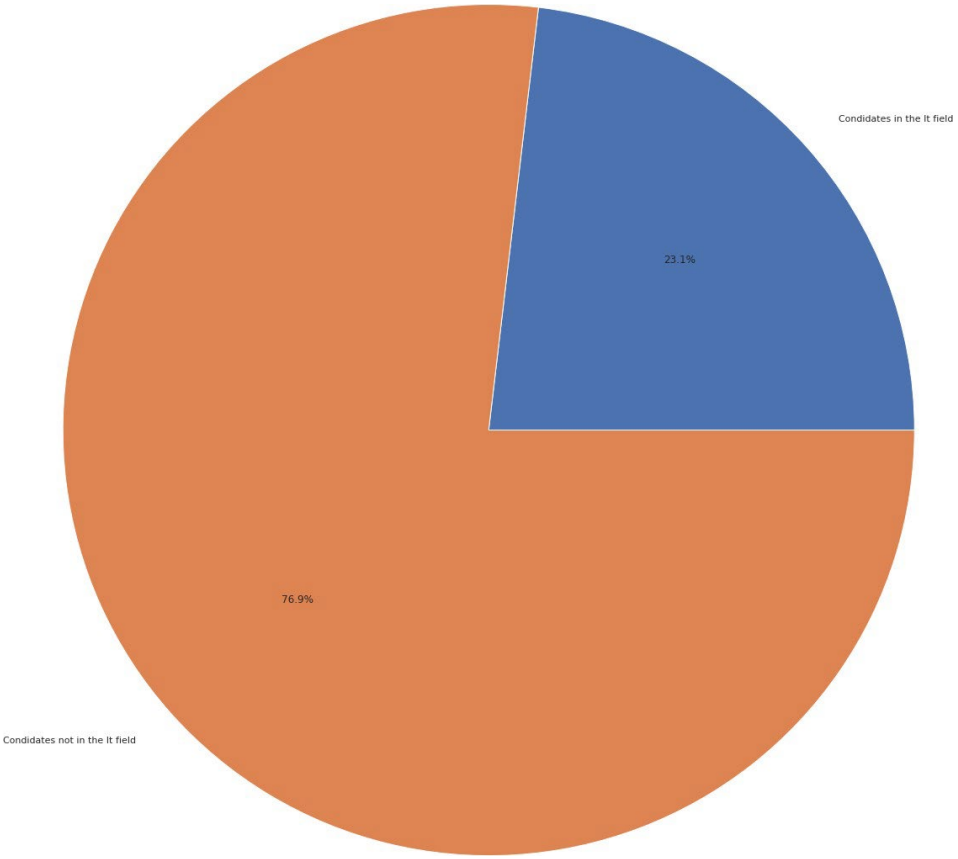
As for the 'Companies worked at' data frame, the following chart shows that Oracle, Microsoft, SAP, Cisco, NIIT, and Infosys have the most significant numbers of candidates having previously worked at them.



At least 155 out of 200 (26.9%, as seen through this chart below) candidates worked at Microsoft before, and 142 out of 200(24.4%) worked at Oracle.



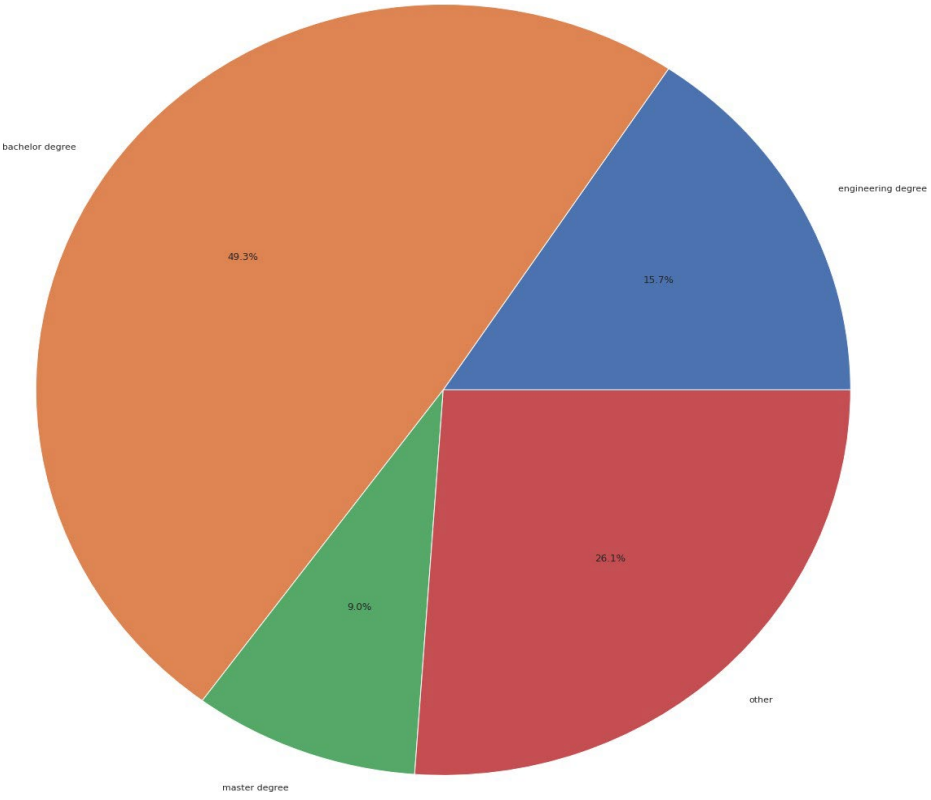
Analyzing the "Degree "data frame showed me that only 23.1% of the candidates for this job are in IT, while the rest have different fields of study, such as business chemistry, electronics, etc.



49.3% of the candidates have a bachelor's degree.

15.7% of them are engineers.

9% of candidates have a master's degree.



Analyzing the universities, the candidates studied at, I figured out that almost everyone went to a different college.

```
CollegeN["text"]=L3
len(CollegeN["text"].unique())
```

```
/usr/local/lib/python3.7/dist-packages
A value is trying to be set on a copy
Try using .loc[row_indexer,col_indexer

See the caveats in the documentation:
  """Entry point for launching an IPyt
238
```

Depending on these insights, the client's HR department can request the elimination of some candidates' categories depending on the job profile needed (for example, the client needs candidates with engineering degrees)

Also, this EDA has shown that those who applied for this job are entirely different, especially when talking about the fields of study; I recommend that the HR department takes more care of the job description and the requirements provided to have a more accurate candidates resumes.

GitHub Repo link: https://github.com/colla00/NLP-Resume-Extraction-Project