chlondonvn / **Messy-Group**

Data cleaning group project for the IronHack Data Analytics 10.2020 Cohort, Berlin Campus

0 stars 2 forks

Star Watch

Code Issues Pull requests Actions Projects Wiki Security

main

chlondonvn Delete .DS_Store ... 3 days ago 13

View code

README.md

Messy Group

Group Members:

```
-Andrew Ashdown
-Charlotte Velilla
-Felix Meier
-Fred Hatanian
-Julien Carbonnell
-Nathan Fournillier
```

12.11.2020

[DATA ANALYSIS 10-20 Cohort, Berlin]

Content

Project Description

Working on the 'Data Science Jobs Market' database to clean, analyse and provide interesting and relevant insight into the data and return a final CSV file of cleaned data combined with visualizations. Parts of the requirements of the projects involved:

Questions/ Requirements

- Who gets hired? What kind of talent do employers want when they are hiring a data scientist?
- Which location has the most opportunities?
- What skills, tools, degrees or majors do employers want the most for data scientists?
- Employ string functions + regexp
- Summarise results by job profile, company, location city, area of the country
- Create new columns : employ Boolean T/F logic
- Handling NULLs in the data

Dataset

Data Scientist Job Market in the U.S.

Link to data: https://www.kaggle.com/sl6149/data-scientist-job-market-in-the-us

The data set used for cleaning is hosted on Kaggle and has been scraped from the web about US data science hires in 2018 (ie pre-covid!).

The dataframe consist of: 6964 rows × 5 columns. A total of 1,682 Null/NaN values were found.

Workflow:

As a group, after a quick review of the data, we were able to locate a number of empty rows to be deleted and discussed how we could further clean each of the columns. For this we used a number of SQL and Python functions, along with some additional libraries for specific tasks, as detailed below:

Data Cleaning/Wrangling:

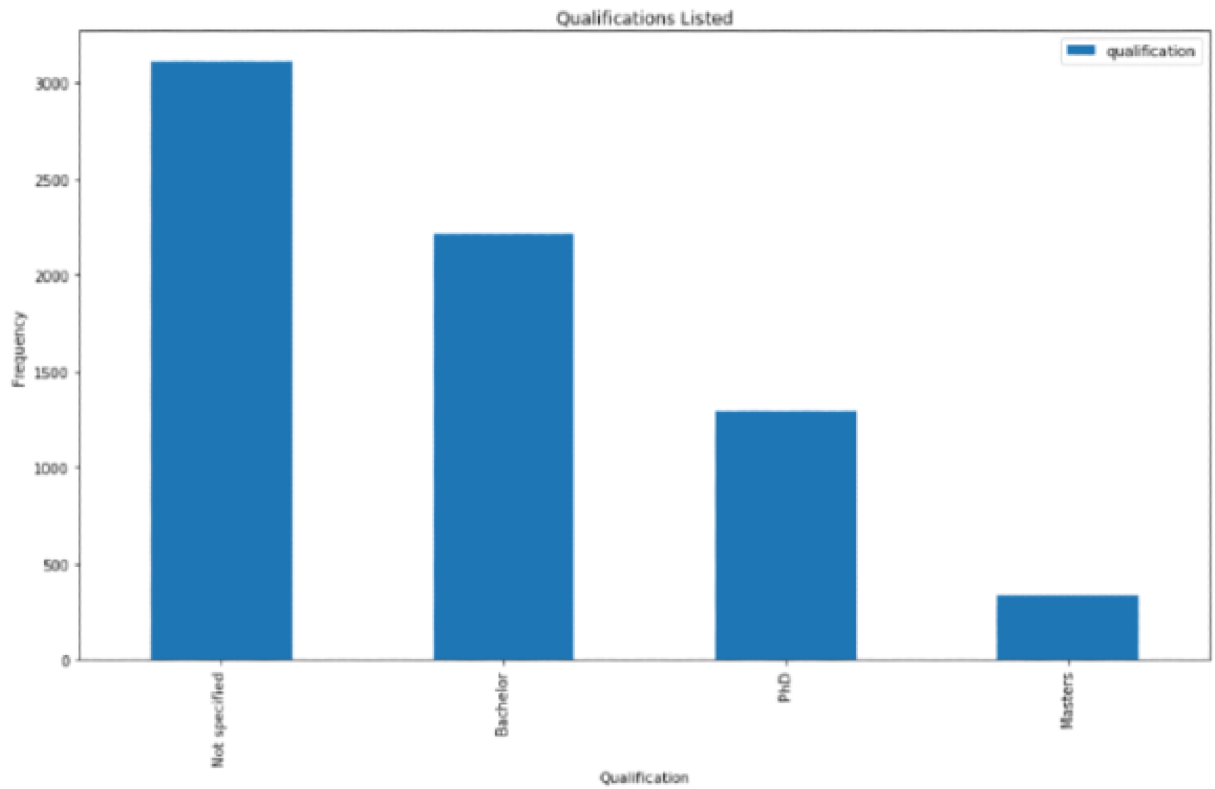
- Company: The data in this column was clear, well formatted and after dropping empty rows, contained no null values, so required no further cleaning.
- Position: We were able to group together roles with similar titles by picking out and linking keywords.
- Location: Split into city, state, zip code. Drop ZIP code and find long and lat coordinates for cities
- Job Description: In order to gain insights on job descriptions such as what skills are needed and what are the keywords that appear most within each of the job descriptions, string functions + regexp was used. Each job description was concatenated, normalized and tokenised to extract keywords and regexp was used to look for 'SQL' and 'Python' skills creating boolean columns with the results.
- Reviews: While the review column could potentially give us some insight into the size of the company, as we are unable to see whether the reviews are positive or negative, we decided to drop the column entirely.

Review

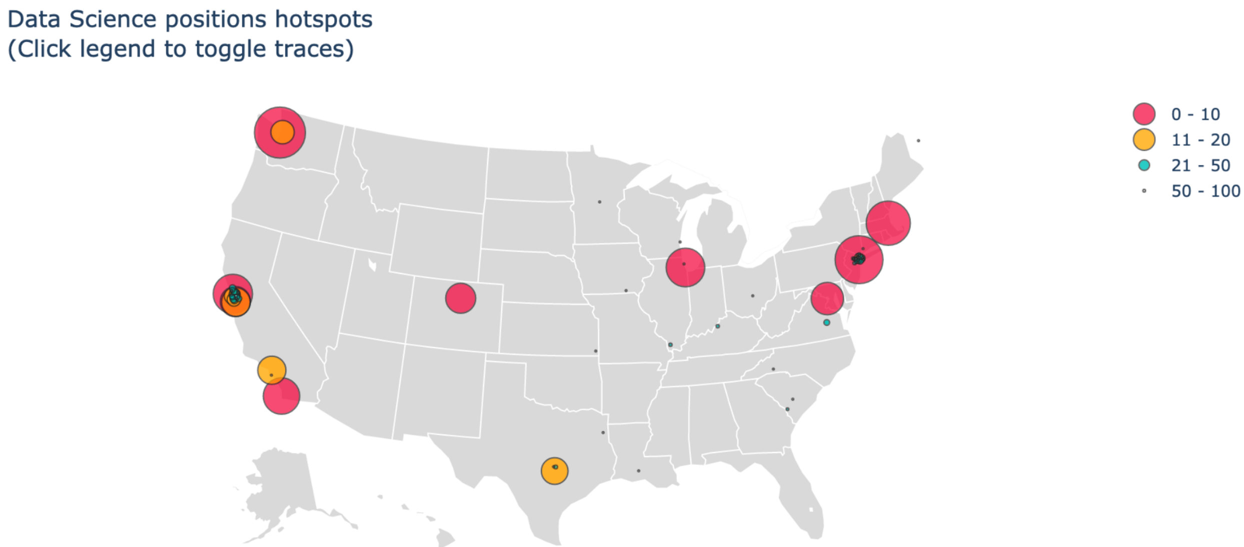
Insights

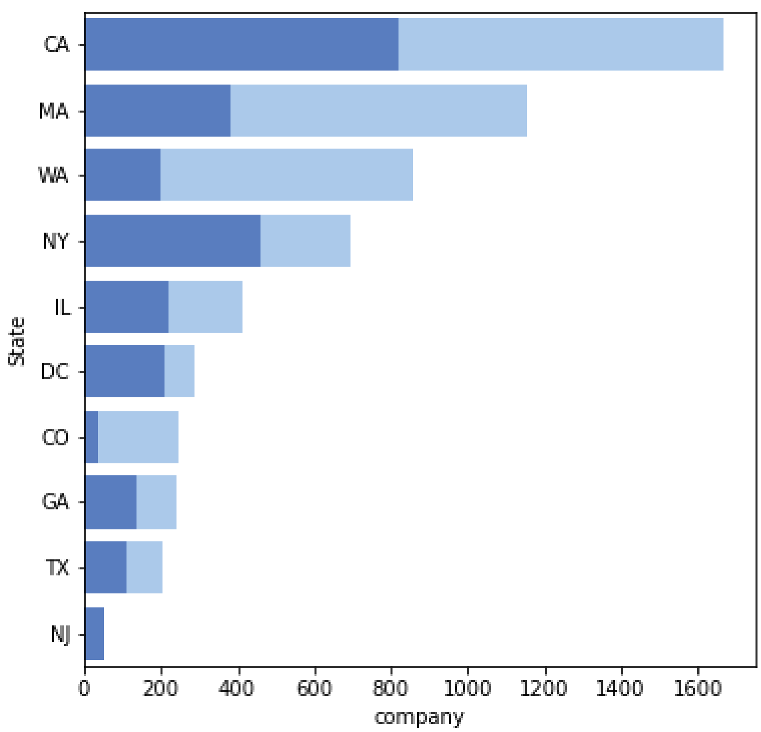
After cleaning and analyzing the data, we were able to find out lots of information about the spread of jobs across the US and the different types of roles available within the Data Science field as well answer the questions and requirements of the project as follow:

1. Who gets hired? What kind of talent do employers want when they are hiring a data scientist?



2. Which location has the most opportunities?





3. What skills, tools, degrees or majors do employers want the most for data scientists?

In [19]:
```
df[['position','skills']]
```

Out[19]:

	position	skills
0	Development Director	,,R,
1	An Ostentatiously-Excitable Principal Research...	,,R,
2	Data Scientist	sql,,R,
3	Data Analyst	sql,,R,
4	Assistant Professor -TT - Signal Processing & ...	,,R,ML
...	...	...
6959	Data Developer / Machine Learning Analyst	sql,,R,ML
6960	Scientist I	,,R,
6961	Intern Scientist	sql,,R,ML
6962	Senior Data & Applied Scientist	,,R,ML
6963	Principal Data Scientist, Deep Learning	sql,,R,ML

6953 rows × 2 columns

	qualification
Not specified	3114
Bachelor	2212
PhD	1295
Masters	332

In [27]:
```
#looking for 'SQL' INSTEAD OF 'sql' in description
df['sql'] = np.where(df['description'].str.contains('SQL'or'sql'), True,False)
```

In [28]:
```
(df['sql'].values == True).sum()
```

Out[28]: 1909

In [29]:
```
df['python'] = np.where(df['description'].str.contains('Python' or 'python'), True,False)
```

In [30]:
```
(df['python'].values == True).sum()
```

Out[30]: 2759

4. Employ string functions + regexp

[8]:
```
#split location into state and city
file[['City', 'State']] = file['location'].str.split(',', 1, expand=True)
```

[ ]:
```
#make new column for zip code
file['Zip-code']=file.location.str.extract('(\d+)')
```

[18]:
```
# delete numbers in State
file['State'] = file['State'].str.replace('\d+', '')
```

[19]:
```
file
```

[19]:

	position	company	description	reviews	location	City	State	Zip-code
0	Development Director	ALS TDI	Development Director\nALS Therapy Development ...	NaN	Atlanta, GA 30301	Atlanta	GA	30301
1	An Ostentatiously-Excitable Principal Research...	The Hexagon Lavish	Job Description\n\n"The road that leads to acc...	NaN	Atlanta, GA	Atlanta	GA	NaN
2	Data Scientist	Xpert Staffing	Growing company located in the Atlanta, GA are...	NaN	Atlanta, GA	Atlanta	GA	NaN
3	Data Analyst	Operation HOPE	DEPARTMENT: Program OperationsPOSITION LOCATIO...	44.0	Atlanta, GA 30303	Atlanta	GA	30303
4	Assistant Professor -TT - Signal Processing & ...	Emory University	DESCRIPTION\nThe Emory University Department o...	550.0	Atlanta, GA	Atlanta	GA	NaN
...	...	...	...	...	...	...	...	...
6959	Data Developer / Machine Learning Analyst	NetApp	Are you data-driven? We at NetApp believe in t...	574.0	Sunnyvale, CA	Sunnyvale	CA	NaN
6960	Scientist I	Pharmacyclics, an Abbvie Company	Pharmacyclics is committed to the development ...	26.0	Sunnyvale, CA	Sunnyvale	CA	NaN
6961	Intern Scientist	Oath Inc	Oath, a subsidiary of Verizon, is a values-led...	5.0	Sunnyvale, CA	Sunnyvale	CA	NaN

5. Summarise results by job profile, company, location city, area of the country

In [91]:
```
#Define positions into smaller groups
```

In [92]:
```
def CleanList (x):
    if 'data scien' in x.lower():
        return 'Data Scientist'
    elif'data analy' in x.lower():
        return 'Data Analyst'
    elif'data engineer' in x.lower():
        return 'Data Engineer'
    elif'research scien' in x.lower():
        return 'Research Scientist'
    elif 'research analy' in x.lower():
        return 'Research Analyst'
    elif 'scientis' in x.lower():
        return 'Scientist'
    elif 'developer' in x.lower():
        return 'Developer'
    elif 'engineer' in x.lower():
        return 'Engineer'
    elif 'senior analyst' in x.lower():
        return 'Senior Analyst'
    elif 'customer s' in x.lower():
        return 'Customer Success'
    elif 'director' in x.lower() or 'executive' in x.lower()or 'head' in x.lower()
        return 'SeniorRole'
    elif 'analy' in x.lower():
        return 'Analyst'
    elif 'research' in x.lower():
        return 'Researcher'
```

In [34]:
```
#isolating the different levels of seniority
df.loc[df['position'].str.contains('Junior|Jr|junior|jr', case=False), 'seniority'] = 'junior'
df.loc[df['position'].str.contains('Senior|Sr|senior|sr', case=False), 'seniority'] = 'senior'
df.loc[df['position'].str.contains('Entry|entry', case=False), 'seniority'] = 'entry level'

df.seniority.value_counts()
```

Out[34]:
```
senior         1486
junior           49
entry level      17
Name: seniority, dtype: int64
```

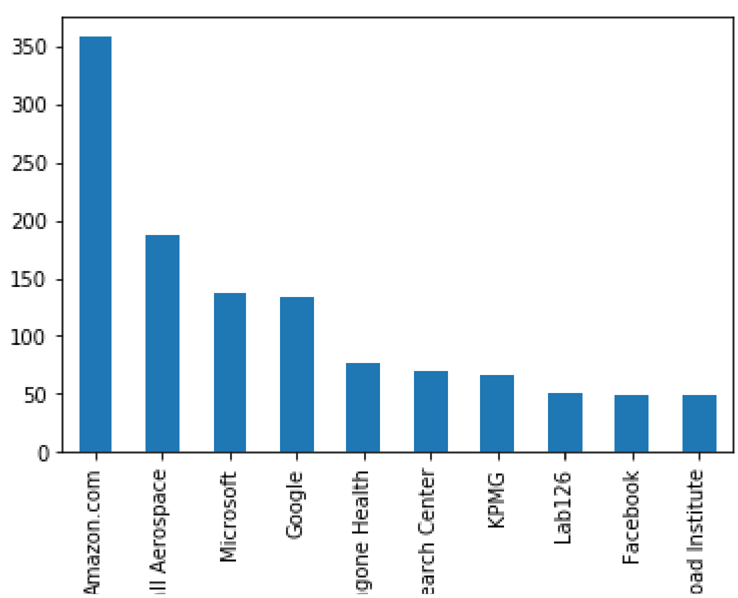
In [95]:
```
alldata['position_grouped'].value_counts()
```

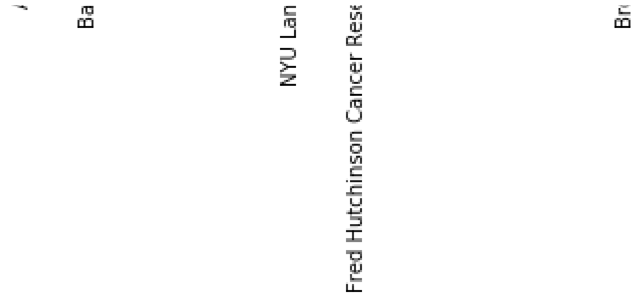
Out[95]:
```
Data Scientist             1444
Scientist                  1093
Engineer                   1007
SeniorRole                  786
Analyst                     373
other                       359
Research Analyst            321
Researcher                  298
Research Scientist          274
Data Analyst                182
Data Engineer               177
Specialist                  110
Developer                   110
Associate                    63
Machine Learning Engineer    61
Architect                    40
Technician                   37
Programmer                   32
Administrator                32
Coordinator                  26
Product Specialist           25
Consultant                   24
Senior Analyst               24
Designer                     19
Recruiter                    12
Writer                       11
Customer Success             10
Junior                        3
Name: position_grouped, dtype: int64
```

6. Create new columns : employ Boolean T/F logic

seniority	sql	python
Director	False	False
NaN	False	False
NaN	True	True
NaN	True	True
NaN	False	False
...	...	...
NaN	True	True
NaN	False	False
NaN	True	True
Senior	False	False
NaN	True	True

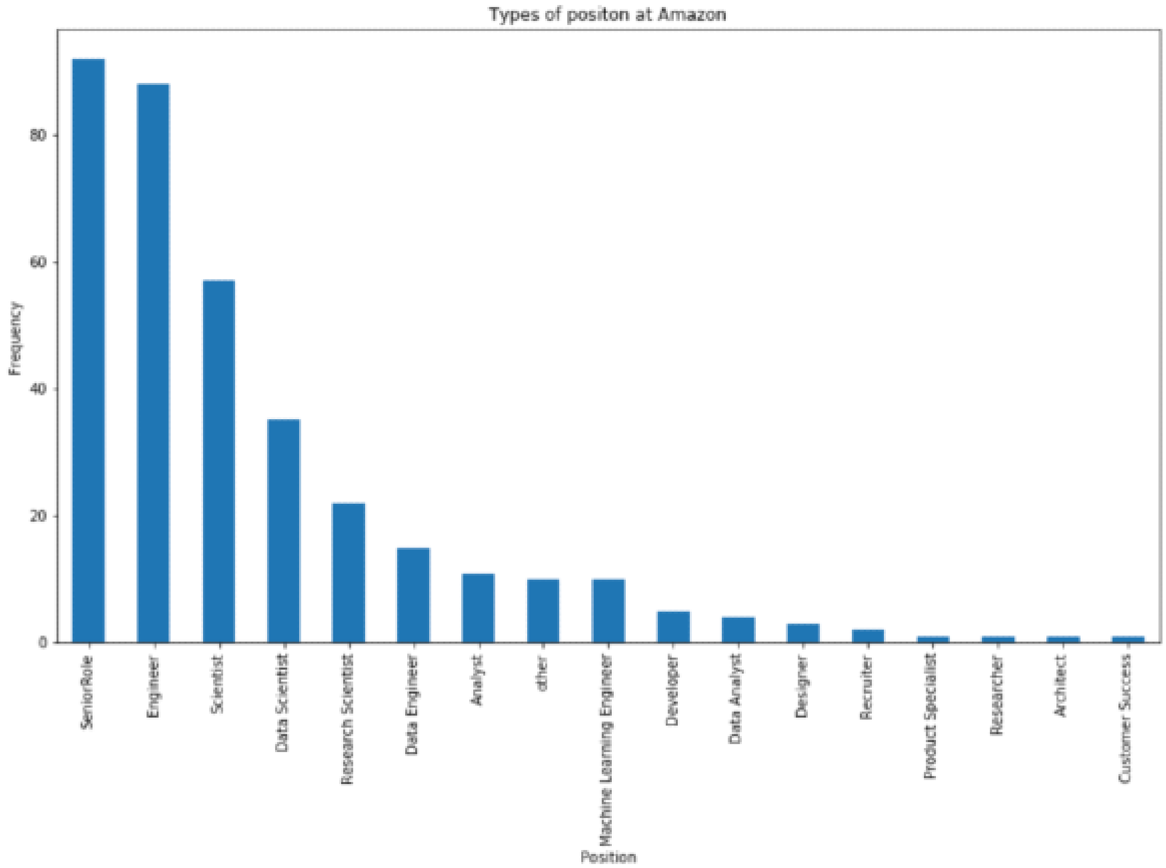
7. Additional Insights

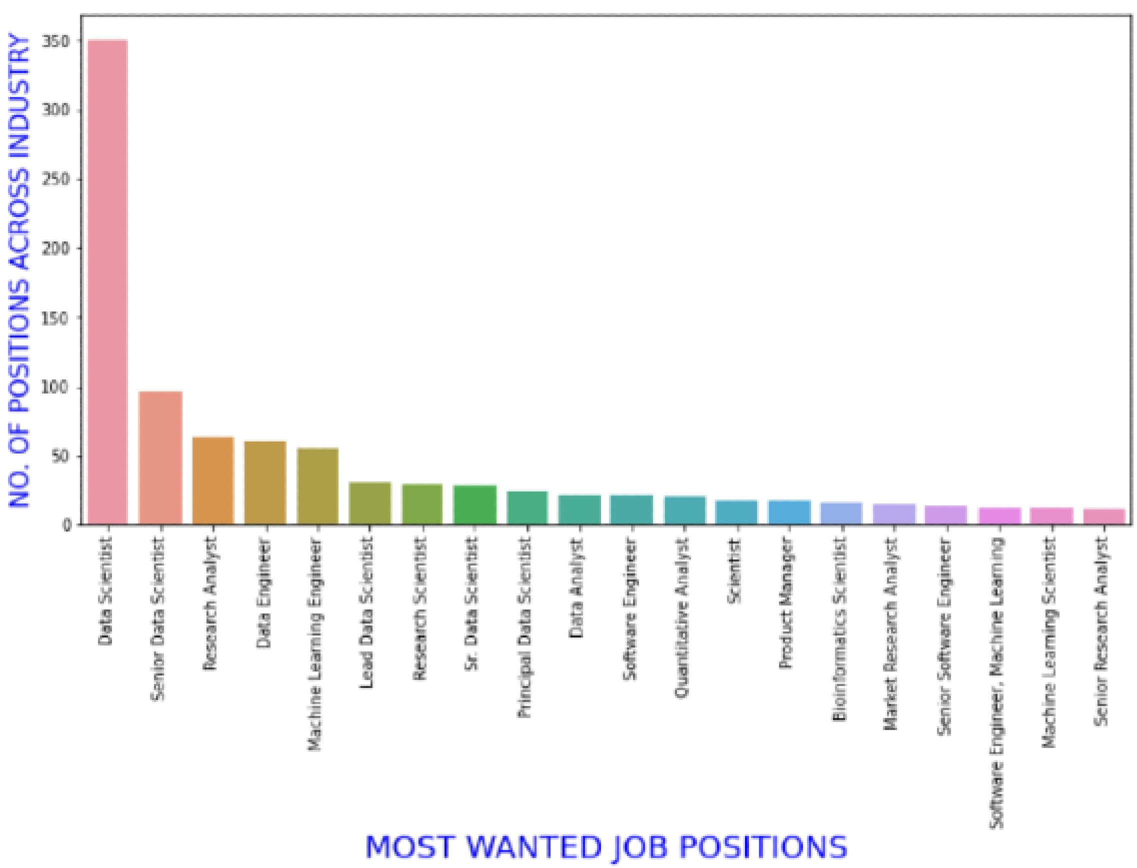

350
300
250
200
150
100
50
0
Amazon.com
ll Aerospace
Microsoft
Google
gone Health
earch Center
KPMG
Lab126
Facebook
oad Institute



In [26]:
```python
# To find the frequency of top 100 words
from nltk.probability import FreqDist
fdist = FreqDist(token_cleaner)
fdist100 = fdist.most_common(100)
fdist100
```

Out[26]:
```
[('data', 37778),
 ('experience', 19441),
 ('work', 16135),
 ('team', 14949),
 ('research', 13582),
 ('development', 11495),
 ('business', 11318),
 ('learning', 9074),
 ('skills', 9042),
 ('new', 8960),
 ('years', 8916),
 ('science', 8793),
 ('including', 8495),
 ('analysis', 7850),
 ('technical', 7573),
 ('machine', 7257),
 ('management', 7239),
 ('software', 7077),
 ('product', 6994),
 ('working', 6925),
 ('support', 6769),
 ('design', 6731),
 ('related', 6557),
 ('engineering', 6512),
 ('ability', 6234),
 ('degree', 6139),
 ('amp', 6115),
 ('systems', 5692),
 ('information', 5479),
 ('opportunity', 5432),
 ('analytics', 5392),
 ('knowledge', 5389),
 ('solutions', 5296),
 ('company', 5243),
 ('teams', 5151),
```


Types of positon at Amazon
Frequency
0
20
40
60
80
SeniorRole
Engineer
Scientist
Data Scientist
Research Scientist
Data Engineer
Analyst
other
Machine Learning Engineer
Developer
Data Analyst
Designer
Recruiter
Product Specialist
Researcher
Architect
Customer Success
Position


NO. OF POSITIONS ACROSS INDUSTRY
0
50
100
150
200
250
300
350
Data Scientist
Senior Data Scientist
Research Analyst
Data Engineer
Machine Learning Engineer
Lead Data Scientist
Research Scientist
Sr. Data Scientist
Principal Data Scientist
Data Analyst
Software Engineer
Quantitative Analyst
Scientist
Product Manager
Bioinformatics Scientist
Market Research Analyst
Senior Software Engineer
Software Engineer, Machine Learning
Machine Learning Scientist
Senior Research Analyst
MOST WANTED JOB POSITIONS

Challenges

We encountered a number of challenges, these included:

- Many crashes before fixing the text mining library on the jupyter notebook.
- Mapping top Tech Hotspots; turns out that most places are the same (e.g. SF bay area) but are grouped differently since city name is different
- Extracting strings from the roles and the description to get seniority and skills and bucket them in meaningful ways.

Further exploration

- Further group columns
- Group the locations more effectively (e.g. combine Manhattan and New York)
- Correlate average salary of position title with location and living cost to get location based purchasing power
- Get review rating, not only number (4 out of 5 stars) to get rating of employer

File structure

```
In the repository the following files are included:
    - alldata.csv
    - cleaned_data.csv
    - Cleaning_Data_Project_The_Messy_Group.ipynb
    - Messy_Graphics directory
    - README
```

Releases

No releases published

Packages

No packages published

Languages

- **Jupyter Notebook** 100.0%