INTRODUCTION TO DATA SCIENCE

Analysis on New York Job Posting Data

Submission on 16 Dec 2019

BITS, pilani WORK INTEGRATED LEARNING PROGRAM

INTRODUCTION TO DATA SCIENCE

NEW YORK JOB POSTING DATA ANALYSIS

(2019_CLUSTER-DSE-IDS_A1_PS2)

Analysed and created in collaboration of

Venkataramanan K | 2018AC04529

Balakavin Pon | 2018AC04531

Ponvani | 2018AC04559

Poornima J | 2018AC04550

Group information

IDS_GROUP006

2019_Cluster-DSE-IDS_A1_PS2

TABLE OF CONTENT

1 | PROBLEM STATEMENT

BUSINESS PROBLEM

Determine the below from the given data set for New York City Current Job Posting data.

a) What are the highest paid Skills in the US market?
b) What are the job categories, which involve above mentioned niche skills?
c) Applying clustering concepts, please depict visually what are the different salary ranges based on job category and years of experience.

SUMMARY OF ANALYSIS

Data Exploration

The data is analyzed to summarize their main characteristics. Steps taken

- ✓ The percentage of null values are computed for each variable. If there are more than 30% null values, that variable is not considered for analysis.
- ✓ Handling missing values – Has been handled by replacing with most frequent occurring data (Mode strategy)

Feature Engineering

Feature engineering is done to make the input dataset compatible with the machine learning algorithm requirements and improve the performance of machine learning models. Steps taken

- ✓ Remove unwanted characters such as special characters, unwanted whitespaces and punctuation.
- ✓ Remove Stop Words
- ✓ Extracting some specific attributes

Analysis Done

- ✓ Highest paid skills was derived by the grouping by job category, preferred skills with salary range mean and picking the top 10 in this list.
- ✓ For clustering , bag of words transformer, word vectorizer was explored. K means clustering mechanism was used.

2 | DATA PREPARATION

Data preparation is the for most important step we could explore the dataset to understand the features and their relationship using both graphical and non-graphical quantitative analysis. The first step is to get to know about the size of the dataset such as the number of records given, number of columns and their data types.

Information about the data

The scheme of the data can be explored by examining the information about the dataset such as number of entries, column count, data type of the columns and the null constraints. As part of the Exploratory Data Analysis, we need to find out the properties of the attributes and identify the suitable variables for further analysis.

```
job_data.info()
print("---"*40)
job_data.describe()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3020 entries, 0 to 3019
Data columns (total 28 columns):
Job ID                          3020 non-null int64
Agency                          3020 non-null object
Posting Type                    3020 non-null object
# Of Positions                  3020 non-null int64
Business Title                  3020 non-null object
Civil Service Title             3020 non-null object
Title Code No                   3020 non-null object
Level                           3020 non-null object
Job Category                    3018 non-null object
Full-Time/Part-Time indicator   2811 non-null object
Salary Range From               3020 non-null float64
Salary Range To                 3020 non-null float64
Salary Frequency                3020 non-null object
Work Location                   3020 non-null object
Division/Work Unit              3020 non-null object
Job Description                 3020 non-null object
Minimum Qual Requirements       3004 non-null object
Preferred Skills                2581 non-null object
Additional Information          1840 non-null object
To Apply                        3019 non-null object
Hours/Shift                     923 non-null object
Work Location 1                 1422 non-null object
Recruitment Contact             0 non-null float64
Residency Requirement           3018 non-null object
Posting Date                    3018 non-null object
Post Until                      972 non-null object
Posting Updated                 3018 non-null object
Process Date                    3018 non-null object
dtypes: float64(3), int64(2), object(23)
memory usage: 660.8+ KB
--------------------------------------------------------------------------------------------------------------------------------
```

Then followed by the overall descriptive analysis of the dataset, which gives us an insight on the statistical view of the numerical features.

	Job ID	# Of Positions	Salary Range From	Salary Range To	Recruitment Contact
count	3020.000000	3020.000000	3020.000000	3020.000000	0.0
mean	383678.501987	2.424172	58140.495550	84325.707875	NaN
std	51779.379258	8.149189	26806.810446	43186.883961	NaN
min	87990.000000	1.000000	0.000000	10.360000	NaN
25%	378264.750000	1.000000	48535.000000	60990.000000	NaN
50%	402960.500000	1.000000	57944.000000	81535.000000	NaN
75%	415741.000000	1.000000	72476.000000	105000.000000	NaN
max	424117.000000	190.000000	218587.000000	234402.000000	NaN

3 | IDENTIFICATION OF VARIABLES

Selection of appropriate variables for analysis plays an important role. We need to explore the data set and see the importance and the relationship of the variables. That will enable us to select the features more suitably for the analysis.

```
# Casting the date fields from string to datetime
job_data['Posting Date'] = pd.to_datetime(job_data['Posting Date'])
job_data['Process Date'] = pd.to_datetime(job_data['Process Date'])
job_data['Post Until'] = pd.to_datetime(job_data['Post Until'])
job_data['Posting Updated'] = pd.to_datetime(job_data['Posting Updated'])

print("No. of numerical columns: {0}".format(len(job_data.select_dtypes(include=np.number).columns.tolist())))
print("No. of non-numerical columns: {0}".format(len(job_data.select_dtypes(exclude=np.number).columns.tolist())))
print("No. of date columns: {0} ".format(len(job_data.select_dtypes(include=np.datetime64).columns.tolist())))
```

```
No. of numerical columns: 5
No. of non-numerical columns: 23
No. of date columns: 4
```

From the requirement doc we need to find below items.

- Skills
- Skills Vs Salary in (desc order)
- Job Category
- Job Category belong to Skills
- Year of Exp
- Salary-Range Vs Job-Category Vs Year of Exp

From the requirements we found that the numerical columns like

1. `Salary Range From`
2. `Salary Range To` are required.

The categorical and text variables like

1. `Job category` ,
2. `Preferred Skills` are required.

The data on `Years of experience` is `Preferred Skills` field. It needs to be extracted from the preferred skills and made as a separate variable.

After we performed the preliminary analysis on the variables data type and the their values the decision was made that there are about 5 features would contribute enough for the further requirements in the analysis. The features are such as,

1. Salary Range From
2. Salary Range To
3. Salary Frequency
4. Preferred Skills
5. Job category

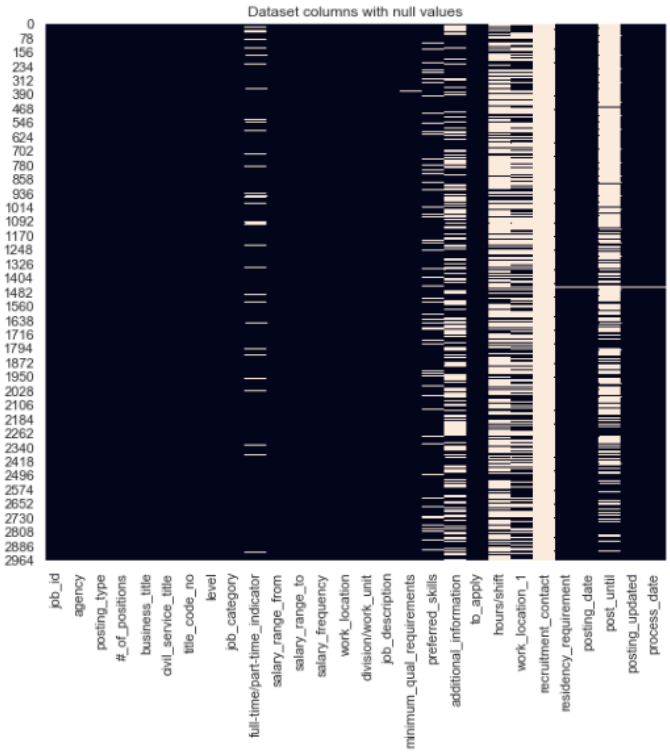
The derived fields are,

1. Years of experience
2. Projected salary from
3. Projected salary to

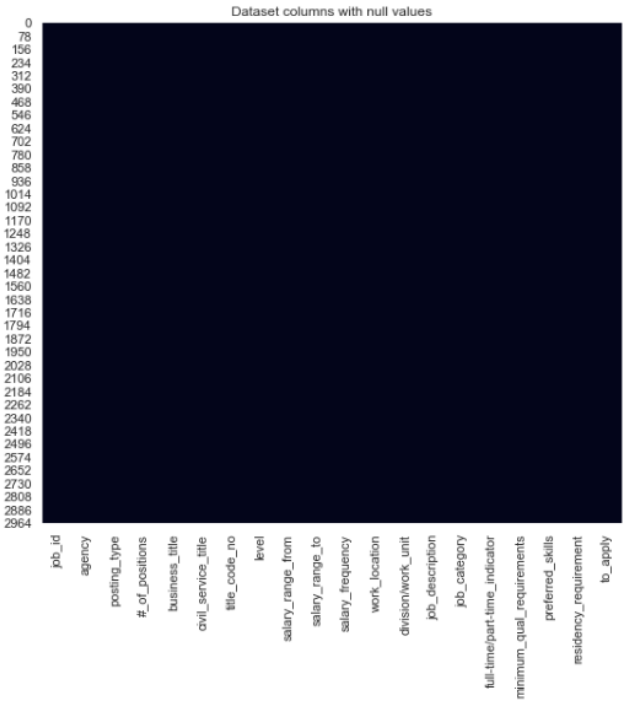
4 | MISSING VALUES AND VARIABLE SELECTION

Imputing the missing values is a crucial step in data preprocessing. The null or empty data in the data fields will not give an expected result in the performance of the built model. We need to impute or fill an appropriate value for those missing fields.

The following type of heatmap depicts the null values of variables against the rows.



After the imputing the fields with most frequent and appropriate values the fields null or NaN data are removed. The plot of after imputed data is given below, where there are no white spaces in the any of the rows of the variables.




```
from sklearn.impute import SimpleImputer

imputer = SimpleImputer(strategy="most_frequent")
nan_cols = ['job_category', 'full-time/part-time_indicator','minimum_qual_requirements', 'preferred_skills','residency_requiremer
cols = [col for col in identified_cols if col not in nan_cols]

# print(cols)
# Apply the imputation on the dataset
imputed_data = pd.DataFrame(imputer.fit_transform(job_data[nan_cols]), columns=nan_cols)

# # imputed_data.head()
posting_data = pd.concat([job_data[cols], imputed_data], axis=1)
# Empty values plot
# combined_data.isnull().sum()
plot_null_data(posting_data)
```

5 | FEATURE ENGINEERING

Feature engineering is concerned with the process of creating features or manipulating the features to make more meaningful for the analysis.

Here in the given dataset, most of the feature data are text based in nature. In order to indulge them in the analysis we need to perform certain feature engineering mechanism such as text parsing, n-gram phrase generation, field extraction, manipulation and deduplication. The 3 import features were taken for feature engineering for this analysis,

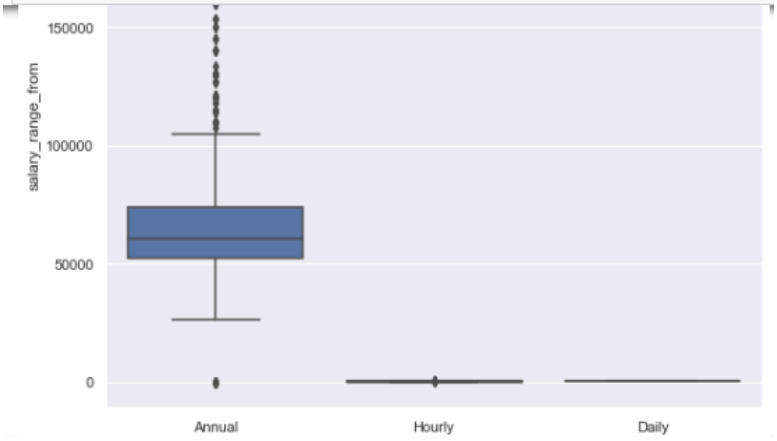
1. Salary Range From and Salary Range To
2. Years of Experience
3. Preferred skills

Salary Range

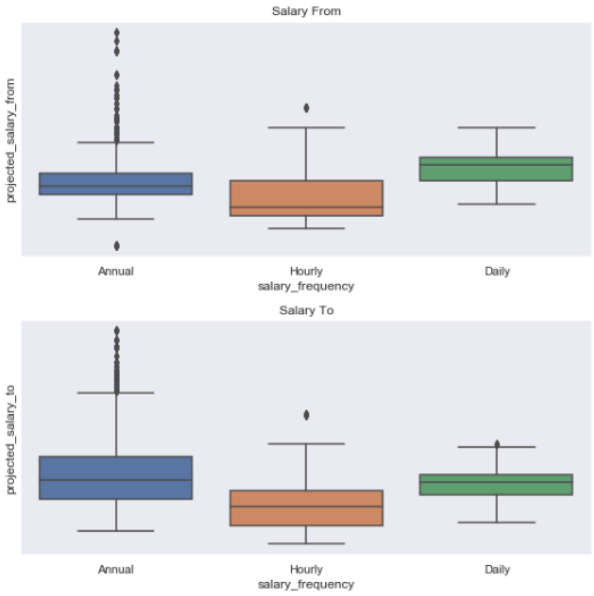
The fields salary range from and to are the important fields with respect to arrive at the results for the analysis, because they are necessary to compare and group the niches skills and job category. However, these fields data are not properly given in terms of the value units. Some records have the money value in hours and some of them are in daily and annullay. It is important we need to standardize them based on the common frequency.

Therefore, the all the records salary information will be commonly treated based on the one frequency either daily or annually. We took the frequency as annually for further steps in the analysis.

```
sns.boxplot(x='salary_frequency',y='salary_range_from', data=job_data)
plt.show()
```



Once the salary range fields are standardized using the salary frequency field, they are plotted in whisker or box plot to visualize the data spread across different quartile.



Years of Experience

3. Feature manipulation - Years of experience

The year of experience is a required information to understand more on the relationship between the salary and years of experience. However, the data is hidden in the text data of `Preferred Skills` field. We need to perform a search to identify the pattern of years and transform the extracted information into a numerical values.

```
# posting_data['Experience Phrase'] = posting_data.head(50)['Preferred Skills'].apply(extract_years_exp)
posting_data['years_of_experience'] = posting_data['preferred_skills'].apply(extract_years_exp)
```

```
posting_data['years_of_experience']
```

```
0       2
1       2
2       2
3       2
4       2
       ..
3015    2
3016    2
3017    2
3018    2
3019    2
Name: years_of_experience, Length: 3020, dtype: int64
```

Preferred Skills

The field preferred skills are the most important feature of the given job posting dataset. Because it had information for skills as well as the years of experience. Based on these only the entire analysis was involved. We need to implement many text manipulation mechanisms to extract meaningful information.

2. Feature manipulation - Preferred Skills

```
# Removing stop words from the two text fields Job description and Preferred skills
for col in str_cols:
    posting_data[col+'new'] = posting_data[col].apply(remove_stopwords)

posting_data[str_cols]
```

	job_description	preferred_skills
0	Division of Economic Financial Opportunity DE...	Excellent interpersonal and organizational ski...
1	The New York City Department of Small Business...	ERROR NAME
2	Under direct supervision assist in the routine...	A High School Diploma or GED CDL Drivers Li...
3	Under direct supervision assist in the routine...	A High School Diploma or GED CDL Drivers Li...
4	Responsibilities of selected candidates will i...	ERROR NAME
...	...	...
3015	The City of New York Department of Housing Pre...	Excellent judgment editing writing and interpe...
3016	Your Team The Office of Enforcement and Neigh...	Must possess excellent written and verbal comm...
3017	Your Team The Office of Enforcement and Neigh...	Must possess excellent written and verbal comm...
3018	The Commission on Human Rights the Commission ...	Advanced working proficiency in Microsoft Offi...
3019	The Commission on Human Rights the Commission ...	Advanced working proficiency in Microsoft Offi...

3020 rows × 2 columns

N-Gram text phrase parsing

We need to convert the Preferred skills data into n-gram string phrases. Then we need to remove `communication and written skill` as it common

Algorithms used for text manipulations,

1. Bag of words
2. Bigram and N-Gram phrase detection
3. Pattern recognition
4. Data Aggregation with min and max functions

Finally displaying picked Top 10 from identified Top 35 values from n-Gram

Note: We can take top 10 skill directly, but since n-gram is used, its better for a domain expert to pick the top 10, which will be more meaningful

Conditions Used

- Atleast 5 occurance in any Preferred Skill Desc
- Bag of Word was generated by picking all noun synonyms of word skill
- Sorted using Average salary

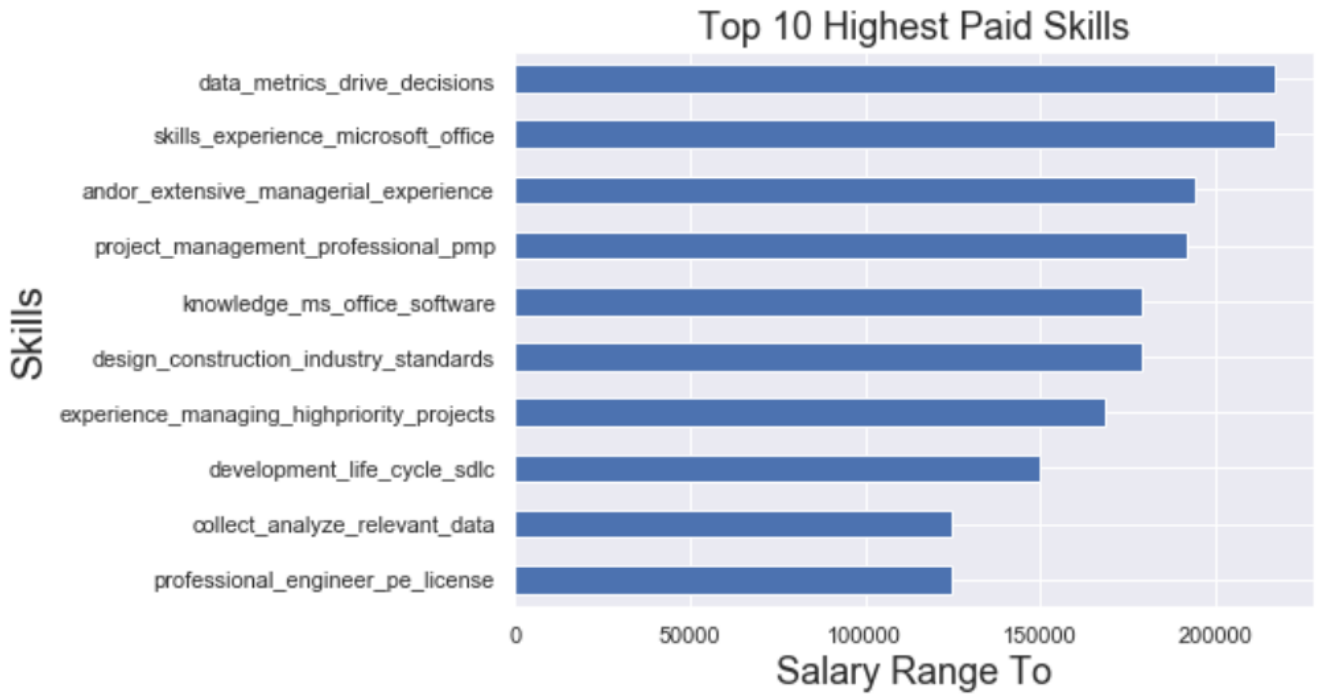
```
# To pick top 35 from n-grams
tetra_df = tetra_df.iloc[:35]

#Handpicked Top 10 from the top 35
tetra_df['selected'] = tetra_df['index_num'].apply(lambda x: x in [1,4,6,12,15,16,22,26,29,32])
color = (tetra_df['selected'] == True).map({True: 'background-color: yellow', False: ''})
tetra_df.style.apply(lambda s: color)
```

	preferred_skills_tetragram	max_salary	sum_salary	cnt_salary	Average_Salary	is_present	index_num	selected
0	knowledge_city_federal_environmental	194395	845801	5	169160	2	0	False
1	andor_extensive_managerial_experience	194395	845801	5	169160	1	1	True
2	skills_experience_writing_reviewing	178873	997478	6	166246	1	2	False
3	related_nonrelated_tasks_proficiency	178873	997478	6	166246	1	3	False
4	knowledge_ms_office_software	178873	997478	6	166246	2	4	True
5	knowledge_design_construction_industry	178873	997478	6	166246	3	5	False
6	design_construction_industry_standards	178873	997478	6	166246	1	6	True
7	experience_writing_reviewing_contract	178873	829045	5	165809	1	7	False
8	towards_technical_nontechnical_audience	194395	980234	6	163372	1	8	False
9	skills_towards_technical_nontechnical	194395	980234	6	163372	1	9	False
10	skills_knowledge_ms_office	178873	1.06178e+06	7	151684	2	10	False
11	computer_skills_knowledge_ms	178873	1.06178e+06	7	151684	2	11	False
12	project_management_professional_pmp	192152	720485	5	144097	1	12	True
13	management_professional_pmp_certification	192152	720485	5	144097	1	13	False
14	knowledge_operations_design_construction	186555	1.10088e+06	8	137610	3	14	False
15	data_metrics_drive_decisions	217244	667785	5	133557	1	15	True
16	skills_experience_microsoft_office	217244	912515	7	130359	1	16	True
17	possess_following_relevant_experience	150000	645225	5	129045	1	17	False
18	skills_project_management_experience	170000	738850	6	123142	1	18	False
19	candidate_possess_following_experience	165000	2.63181e+06	22	119628	1	19	False
20	masters_degree_accredited_college	217244	1.43484e+06	12	119570	1	20	False
21	word_excel_powerpoint_knowledge	149500	592770	5	118554	2	21	False
22	development_life_cycle_sdlc	150000	560323	5	112065	1	22	True
23	priorities_manage_experience_managing	168433	744232	7	106319	1	23	False
24	manage_experience_managing_highpriority	168433	744232	7	106319	1	24	False
25	competing_priorities_manage_experience	168433	744232	7	106319	1	25	False
26	experience_managing_highpriority_projects	168433	838993	8	104874	1	26	True
27	preference_candidates_experience_government	168433	523555	5	104711	1	27	False
28	candidates_experience_government_agencies	168433	523555	5	104711	1	28	False
29	collect_analyze_relevant_data	125000	625130	6	104188	1	29	True
30	analyze_relevant_data_spreadsheets	125000	625130	6	104188	1	30	False
31	experience_solutions_project_teams	114000	518647	5	103729	1	31	False
32	professional_engineer_pe_license	125000	511347	5	102269	1	32	True
33	knowledge_nyc_construction_codes	114000	599496	6	99916	3	33	False
34	understanding_project_management_principals	118610	498520	5	99704	1	34	False

6 | Analysis

6.A Highest Paid Skills in US Market

The skillset information was present in the attribute 'Preferred Skills'. The text corpus had various combination of the string phrase to denote the skills. The analysis employed text parsing, and extraction techniques to identify and sort them out from the unstructured data. The below horizontal bar chart depicts the Top 10 highest paid skills in the US market against the salary range.



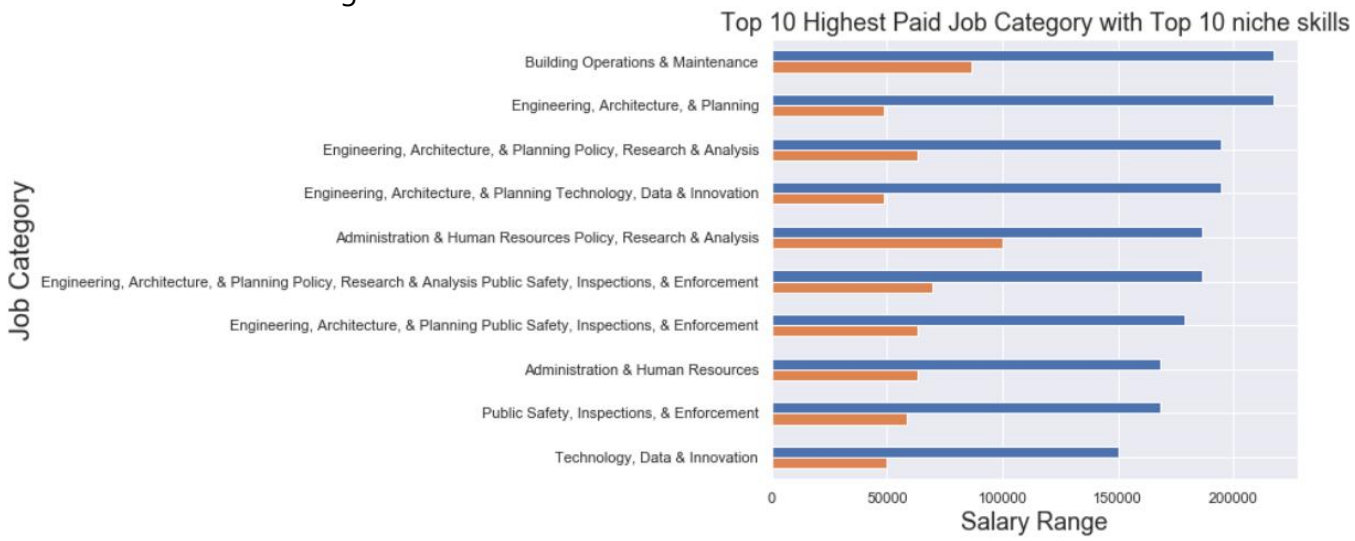
6.B Job categories involve the mentioned Niche Skills

The second question in the analysis is to look for the top 10 job categories with a niche skills in the US Market. The niche skills were identified from the extracted preferred skills and sorted by the calculated salary range.

The column `max_salary` denotes the values of normalized salary from the given range of salary from and salary to.

preferred_skills_tetragram	preferred_skills_tetragram	max_salary
data_metrics_drive_decisions	data_metrics_drive_decisions	217244.0
skills_experience_microsoft_office	skills_experience_microsoft_office	217244.0
andor_extensive_managerial_experience	andor_extensive_managerial_experience	194395.0
project_management_professional_pmp	project_management_professional_pmp	192152.0
knowledge_ms_office_software	knowledge_ms_office_software	178873.0
design_construction_industry_standards	design_construction_industry_standards	178873.0
experience_managing_highpriority_projects	experience_managing_highpriority_projects	168433.0
development_life_cycle_sdlc	development_life_cycle_sdlc	150000.0
collect_analyze_relevant_data	collect_analyze_relevant_data	125000.0
professional_engineer_pe_license	professional_engineer_pe_license	125000.0

The graph below depicts the job category wise highest paid skills in the US market. The skills are NICHES SKILLS which are rare in market but high in demand.



6.c Clustering on the data

Clustering technique is usually applied on the unstructured data to see the affinity of the data between each datapoints and their arrangements in the given dimensional space. Here, the clustering techniques were applied to see the salary and years of experience data points grouping based on the job category. In the clustering technique, first and foremost step is to standardize the feature values. Standardization is useful for data which has negative values. It arranges the data in normal distribution.

Normalize the data

The StandardScalar class object is used to fit and transform the feature values into a normalized form.

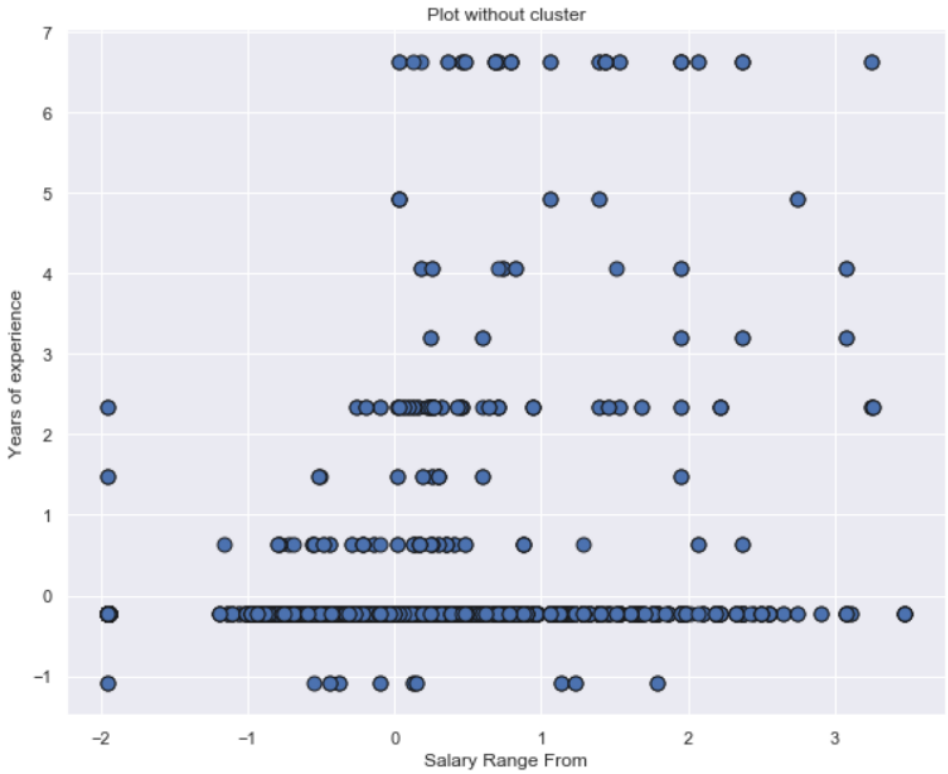
In [35]:
```
from sklearn.feature_extraction.text import CountVectorizer, TfidfTransformer
from sklearn.preprocessing import MinMaxScaler, StandardScaler
# Normalizing the data using Sclaing method
normalized_data = StandardScaler().fit_transform(posting_data[['salary_range_to', 'years_of_experience']])
```

Building Cluster Model

K-Means cluster algorithm was used to build the cluster model. The normalized data was used to fit into the k-means algorithm.

K-Means is a centroid based clustering algorithm, which forms a cluster based on the calculated centroids.

In [118]:
```
from sklearn.cluster import KMeans
from sklearn.preprocessing import MinMaxScaler

# Plot the values before clustering the data
x = normalized_data[:, 0] #salary_range_from
y = normalized_data[:, 1] #years_of_experience

fig, axes = plt.subplots(figsize=(10,8))
axes.set_title("Plot without cluster")
axes.set_xlabel("Salary Range From")
axes.set_ylabel("Years of experience")
plt.scatter(x, y, s=81, cmap='viridis', edgecolor='k')

plt.show()
```



The scaled data of years of experience and salary range are plotted in a scatter graph.

The normalized data are fitted into the K-Means model object with basic set of parameters.

```
In [37]: # Apply clustering algorithm
         km_model = KMeans(n_clusters=5, init="k-means++", n_init=10, max_iter=100)

         km_model.fit(normalized_data)
```

```
Out[37]: KMeans(algorithm='auto', copy_x=True, init='k-means++', max_iter=100,
                n_clusters=5, n_init=10, n_jobs=None, precompute_distances='auto',
                random_state=None, tol=0.0001, verbose=0)
```

Once the cluster is fitted with a data, the labels of the cluster were taken from the properties of the model object and used to plot the data with different color code for each cluster.

The following plot depicts the data points of years of experience and salary range in 2-D space. The data points based on the clusters were colored appropriately with different color codes.