Friend of Faux: Determining the Validity of Job Posting Descriptions

Leandra Colley

MAT 231 Final Project

'Real or Fake: Fake Job Posting Description Prediction' is a dataset found using Kaggle; it can be found here. Originally, it was produced by The University of the Aegean| Laboratory of Information & Communication Systems Security.

In this dataset, there are 17,880 job posts from 2012 to 2014, and from that, 17, 014 jobs prove to be real and 866 jobs are fraudulent. The University of the Aegen did the tough work of determining the validity of the job posting for us, so all that we have left to do is parse the data and make it make sense.

The idea of what is 'sensitive' information is slowly changing day, and fraudulent organizations are taking advantage of our complacency. We know to be wary of scam phone calls telling us we won one million dollars for a contest we never entered, to avoid clicking banners of websites asking us for our email adress and phone number in exchange for a prize, and to never give out our passwords for our accounts. Even with all this caution, however, the University has revealed that there is another vastly understudied area where fraudulent companies and individuals are easily finding new victims: online job boards. We unknowingly release our sensitive information in the forms of resumes— phone numbers, emails, addresses, social media, former education, and other potential security question answers. This dataset, at its core, is meant to equip job seekers with the understanding that each job listing- real or fake- should be scrutinized before applying. As you will see, it is not a simple task to distinguish between what is fact or fiction.

How did the University determine which jobs were real and fake?

This dataset contains real-life job ads posted by Workable, a talent acquisition software, and specialized Workable employees manually annotated each entry. The criteria for the classification were based on client's suspicious activity, false contact or company information, complaints from candidates, and periodic analysis of the clients.

Additionally, the University sanitized all entries and filtered out non-English words, removed standard English stop-words, then taught machines what to look for using the bag of words model. In short, the University determined which job listings were fake by examining both the company itself and the job description, using both human and AI investigation techniques.

What questions can this analysis help answer?

- What are the major differences between the job descriptions for real job listings and fake job listings?
  - Is this a similar comparison to the difference in job requirements?
- Who is being targeted?
  - Which industries have the most fraudulent entries?
  - What are the experience requirements of the most fraudulent entries?
  - Which education levels are impacted the most by fraudulent entries in this dataset?

In [1083]:

```
%matplotlib inline
import pandas as pd
import matplotlib
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sbn
from altair import Chart, X, Y, Color, Scale
import altair as alt
from vega_datasets import data
import string
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\lilie\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

Now that we have everything imported, let's read in our data.

In [1084]:

```
fj=pd.read_csv('fake_job_postings.csv') #fj for fake jobs
```

In [1085]:

```
#Here, let's look at our column titles and our first five rows of information
print(fj.columns)
fj.head()
```

```
Index(['job_id', 'title', 'location', 'department', 'salary_range',
       'company_profile', 'description', 'requirements', 'benefits',
       'telecommuting', 'has_company_logo', 'has_questions', 'employment_type',
       'required_experience', 'required_education', 'industry', 'function',
       'fraudulent'],
      dtype='object')
```

Out[1085]:

	job_id	title	location	department	salary_range	company_profile	description	requirements	benef
0	1	Marketing Intern	US, NY, New York	Marketing	NaN	We're Food52, and we've created a groundbreaki...	Food52, a fast-growing, James Beard Award-winn...	Experience with content management systems a m...	N
1	2	Customer Service - Cloud Video Production	NZ, , Auckland	Success	NaN	90 Seconds, the worlds Cloud Video Production ...	Organised - Focused - Vibrant - Awesome!Do you...	What we expect from you:Your key responsibilit...	What y will g fr usThrou being p c
2	3	Commissioning Machinery Assistant (CMA)	US, IA, Wever	NaN	NaN	Valor Services provides Workforce Solutions th...	Our client, located in Houston, is actively se...	Implement pre-commissioning and commissioning ...	N
3	4	Account Executive - Washington DC	US, DC, Washington	Sales	NaN	Our passion for improving quality of life thro...	THE COMPANY: ESRI – Environmental Systems Rese...	EDUCATION: Bachelor's or Master's in GIS, busi...	C culture anyth I corpor —we ha
4	5	Bill Review Manager	US, FL, Fort Worth	NaN	NaN	SpotSource Solutions LLC is a Global Human Cap...	JOB TITLE: Itemization Review ManagerLOCATION:...	QUALIFICATIONS:RN license in the State of Texa...	F Bene Offer

Our column titles are very straightforward and easy to grasp. First, each entry has a unique job_id differentiating itself from the other listings. Next, each entry lists its title and location. As you can see from the first few entries, some locations are outside of the United States. 'Department' indicates the corporate department, such as sales, and many listings do not have a value in this column. 'Salary_range' is bare for the most part, and we can infer that the reasoning is because many comapnies- real or fake- choose to not disclose their salary information. 'Company_profile' is how the company establishes themselves in the job listing. For may of these entries, we can see that the company's profile is a key factor of whether the job listing is real. Further on, we have the job description along with its requirements and benefits. The 'telecommuting', 'has_company_logo', and 'has_questions' (screening questions) are the weakest sources of information, in my opinion, but nevertheless, each row contains 1 if true and 0 if false. 'Employment_type' tells of whether the position if full-time, contract, etc.. 'Required_experience' indicates what the company is looking for in terms of an executive, an entry-level employee, or an intern; and 'required_education' indicates the desired degree level. 'Industry' includes categories such as automotive, IT, health care, and real estate; and the 'function' category relates to that with the position's field- consulting, engineering, research, sales, and various others. Our 'fraudulent' category indicates why the University of the Aegean simplified our tasks because they did the work and classified each entry as fraudulent, indicated by the 1 (true), or real, 0.

First, we are concerned with whether or not the posting is fraudulent. Creating a data frame that only shows fraudulent entries will ensure that if we want to revisit the original data frame, it's entries won't be disrupted. We should isolate our fraudulent job listings. In this column, 0= false and 1= true for fraudulent, meaning our 1's are most important at the moment.

In [1086]:

```
fakes = fj[fj['fraudulent'] == 1] #from the 'fraudulent' column in fj, add the entry to the new da
taframe if its value is 1
```

```
dataframe if its value is 1
fakes.shape
#fakes.head(20)
```

Out[1086]:

```
(866, 18)
```

Of the original 17,880 job postings, there are 866 posts that were found to be fake. With the fraudulent postings in their own data frame now, we can get a clearer image of the information being presented to us. From the first 20 entries, it seems that there are too many unnecessary rows with missing values, so let's delete them from our data frame. However, I will still be keeping some columns that are sparse. This is because 'salary_range' and 'company_profile' can provide valuable insights about the company and their outward appearance that columns like 'has_company_logo' cannot. Additionally, many companies do not post their offered salaries online, and hiring managers may not see a reason to post their logos onto their listing.

In [1087]:

```
fakes.drop(columns=['telecommuting', 'department', 'has_company_logo','has_questions'])
```

Out[1087]:

	job_id	title	location	salary_range	company_profile	description	requirements	benefits	emp
98	99	IC&E Technician	US, , Stocton, CA	95000-115000	...	IC&E Technician | Bakersfield, CA Mt. Poso...	QualificationsKnowledge, Skills & Abilitie...	BENEFITSWhat is offered:Competitive compensati...	
144	145	Forward Cap.	NaN	NaN	NaN	The group has raised a fund for the purchase o...	NaN	NaN	
173	174	Technician Instrument & Controls	US	NaN	Edison International and Refined Resources hav...	Technician Instrument & ControlsLocation D...	JOB QUALIFICATIONS-Ability to understand proce...	we are a team of almost 8,000 employees who he...	
180	181	Sales Executive	PK, SD, Karachi	NaN	NaN	Sales Executive	Sales Executive	Sales Executive	
215	216	IC&E Technician Mt Poso	US, CA, Bakersfield, CA / Mt. Poso	95000-115000	...	IC&E Technician | Bakersfield, CA Mt. Poso...	QualificationsKnowledge, Skills & Abiliti...	BENEFITSWhat is offered:Competitive compensati...	
...	...	...	...	...	...	...	...	...	
17827	17828	Student Positions Part-Time and Full-Time.	US, CA, Los Angeles	NaN	NaN	Student Positions Part-Time and Full-Time.You ...	NaN	NaN	
17828	17829	Sales Associate	AU, NSW, Sydney	NaN	NaN	LEARN TO EARN AN EXECUTIVE LEVEL INCOMEFULL TR...	What You Can Do. • Have the potential to earn ...	Who We Are We are a Global Leadership Developm...	
17829	17830	Android Developer	PL, MZ, Warsaw	NaN	NaN	inFullMobile Sp. z o.o. is a mobile software d...	• A proven track record in Android / JAVA proj...	attractive salary (adequate to the candidate s...	
17830	17831	Payroll Clerk	US, NY, New York	NaN	NaN	JOB DESCRIPTIONWe are seeking a full time payr...	JOB REQUIREMENTS• High school diploma or eq...	We offer a competitive salary and benefits pac...	
17831	17832	Furniture mover	US, IL, Chicago	NaN	Anthony Warren is a Marketing and Advertising ...	earn $500 to $1000 a week as a mover. Must ha...	Good liftflexibile schedulegeat attitude	NaN	

866 rows × 14 columns

Now that our information is a bit condensed, let's see what we can do with it.

What Do The Descriptions Have In Common?

Now let's see if there are any common words that continuously appear in job descriptions. To examine that, let's use a word cloud. In order to do this we must convert the text to lowercase since pandas is case sensitive, remove all punctuation because 'hello.' appears as a different entry than 'hello', break the string into a list of words, and remove "stop words" from the list. "Stop words" usually refers to the most common words in a language, but any group of words can be chosen as the stop words for a given purpose. For example, 'the' and 'that' are typical stop words, but I am able to create any list of these kinds of words.

In [1088]:

```
fakes['description'].dtypes
# There is a float within the description column, so we must change the dtypes in the column to pr
oceed
```

Out[1088]:

```
dtype('O')
```

Case, Punctuation, Tokenization

In [1089]:

```
fakes['description'] = fakes['description'].astype(str)
fakes['description'].dtypes
fakes.head() #successfully converted to strings

# This line makes all words in the column lowercase
fakes['description'] = fakes.description.apply(lambda x: x.lower())

# This is a lambda funtion that replaces punctuation with white space
fakes['description'] = fakes.description.apply(lambda x: x.translate(str.maketrans(string.punctuati
on, ' '*len(string.punctuation))))

# This removes all instances of 'amp' in the description because it is a text processiing error an
d replacement for '&'
fakes['description'] = fakes['description'].str.replace('amp', '')

# This adds a new column, called word_list, to our dataframe, and
# nltk.word_tokenize breaks down text into pieces of information like words and sentences into a l
ist
fakes['word_list'] = fakes.description.apply(nltk.word_tokenize)
fakes[['description', 'word_list']].head()
#now we have created a new column, 'word_list,' where each entry is a list of all of the words in
the description column
```

Out[1089]:

	description	word_list
98	ic e technician bakersfield ca mt posospri...	[ic, e, technician, bakersfield, ca, mt, posop...
144	the group has raised a fund for the purchase o...	[the, group, has, raised, a, fund, for, the, p...
173	technician instrument controlslocation dewe...	[technician, instrument, controlslocation, dew...
180	sales executive	[sales, executive]
215	ic e technician bakersfield ca mt posospri...	[ic, e, technician, bakersfield, ca, mt, posop...

Stop Words

Now, we can count the frequency of each word.

In [1090]:

```
from collections import Counter
c = Counter(fakes.word_list.sum())
c.most_common(10)
```

Out[1090]:

```
[('and', 8016),
 ('to', 4472),
 ('the', 4313),
```

```
('of', 2767),
 ('a', 2362),
 ('in', 2234),
 ('for', 2158),
 ('with', 1767),
 ('are', 1217),
 ('is', 1163)]
```

It is great to see that the words 'is' and 'the' appear often in these job descriptions, but they don't really tell us any new information about the job post descriptions. Here, we will set the stop words.

In [1091]:

```
sw = set(stopwords.words('english'))
#You can also modify the list by adding words of your choice

len(sw) #there are 179 stop words
#sw #prints all of the stop words
```

Out[1091]:

```
179
```

Removing Stop Words

In [1092]:

```
#This is a lambda fuction that takes a list of words as input then gives back a new list where eac
h word that appears in sw is removed
fakes['word_list'] = fakes.word_list.apply( lambda list_of_words: [word for word in list_of_words i
f word not in sw])

d = Counter(fakes.word_list.sum())
d.most_common(10)
```

Out[1092]:

```
[('work', 1050),
 ('team', 518),
 ('experience', 496),
 ('company', 491),
 ('time', 489),
 ('position', 477),
 ('management', 463),
 ('project', 453),
 ('customer', 453),
 ('service', 449)]
```

In [1093]:

```
# Now we can make our word cloud

from wordcloud import WordCloud

fakes['clean_string'] = fakes.word_list.apply(lambda list_of_words: " ".join([word for word in list
_of_words if word not in sw]))
#word cloud can only process strings, so we have to combine our list of words into a new column

fakes.head(1)
```

Out[1093]:

	job_id	title	location	department	salary_range	company_profile	description	requirements	benefits	tel
98	99	IC&E Technician	US, , Stocton, CA	Oil & Energy	95000-115000	...	ic e technician bakersfield ca mt posopri...	QualificationsKnowledge, Skills & Abilitie...	BENEFITSWhat is offered:Competitive compensati...	

In [1094]:

```
wordcloud_fakes = WordCloud(width = 1500, height = 1500,
                background_color ='white',
                min_font_size = 15).generate(fakes.clean_string.sum())
plt.imshow(wordcloud_fakes, interpolation='bilinear')
plt.title('Fake Job Descriptions Wordcloud')

plt.axis("off")
```

Out[1094]:

```
(-0.5, 1499.5, 1499.5, -0.5)
```

And there we have our word cloud!

What can we infer about the types of jobs offered?

We see that they're positiions that are customer service/ client based. From my knowledge of jobs, these could be either call center positions or commission-based sales because of how large the words 'client', 'team', 'company', and 'support' are. For jobs of these types, you mainly need public speaking abilities because the employer provides the training for what the position requires. This is especially interesting because if you look at 'fakes.head(30)', you will see that many of the industries are healthcare, IT, and oil& energy. Can we infer that these positions are posted because they're an easy way for someone to get their start within an industry that both requires a specialized degree but is also making good amounts of money right now? Alternatively, many of the STEM field entries have an experience requirement of mid-senior level, which translates to about 7-9 years of experience. Would this kind of person, who has probably been off the job market for those 7-9 years, know which jobs are fake? Additionally, would they want to transition into more of a client-focused/ consulting role, which is why these words are prominent?

What stands out most right now is how this seems to be comprised of mostly soft skills that most applicants with a college degree, or even a high school diploma, could already have. The words seem to be vague- nothing about specific job industries or degree labels (if we don't count oil and gas). Anyone can do it. The goal of these false listings is to be indistinguishable from real listings. Furthermore, we are trained to include many of these ideas concepts, such as team building, responsibility, assurance, and people/ customer skills in our resumes, so these job descriptions seem like they are casting a wide net into the recent college-grad and desperate job-hunters pool in order to pull in the greatest amount of applicants possible.

What would it look like if we only examined the non-fraudulent job descriptions?

In [1095]:

```
#Below is the same process we went through in order to isolate the fake job postings with 'real' r
eplacing 'fakes'

real = fj[fj['fraudulent'] == 0]
real.drop(columns=['telecommuting', 'department', 'has_company_logo','has_questions'])
real['description'] = real['description'].astype(str)

real['description'] = real.description.apply(lambda x: x.lower())
real['description'] = real.description.apply(lambda x: x.translate(str.maketrans(string.punctuation
, ' '*len(string.punctuation))))
real['description'] = real['description'].str.replace('amp', '')
real['word_list'] = real.description.apply(nltk.word_tokenize)
sw = set(stopwords.words('english'))
#newsw = ['gas', 'oil', 'work', 'company', 'project']
#sw.update(newsw)
```

```
real['word_list'] = real.word_list.apply( lambda list_of_words: [word for word in list_of_words if
word not in sw])
real['clean_string'] = real.word_list.apply(lambda list_of_words: " ".join([word for word in list_o
f_words if word not in sw]))
wordcloud_real = WordCloud(width = 1500, height = 1500,
                background_color ='white',
                min_font_size = 15).generate(real.clean_string.sum())
plt.imshow(wordcloud_real, interpolation='bilinear')
plt.title('Real Jobs Descritions')
plt.axis("off")
```

Out[1095]:

```
(-0.5, 1499.5, 1499.5, -0.5)
```

How does a wordcloud of the job requirements for real jobs look?

The job description is where the company markets themselves to an applicant; it explains a snippet of who the company is and what an applicant can expect when they come onboard. The job requirements explains who the company is looking for to fill their position. Is there a noticeable difference between the wordclouds for the descriptions and the wordclouds for the requirements?

In [1096]:

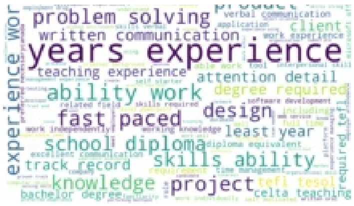
```
real = fj[fj['fraudulent'] == 0]
real.drop(columns=['telecommuting', 'department', 'has_company_logo','has_questions'])
real['requirements'] = real['requirements'].astype(str)

real['requirements'] = real.requirements.apply(lambda x: x.lower())
real['requirements'] = real.requirements.apply(lambda x: x.translate(str.maketrans(string.punctuati
on, ' '*len(string.punctuation))))
real['requirements'] = real['requirements'].str.replace('amp', '')
real['word_list'] = real.requirements.apply(nltk.word_tokenize)
sw = set(stopwords.words('english'))


real['word_list'] = real.word_list.apply( lambda list_of_words: [word for word in list_of_words if
word not in sw])
real['clean_string'] = real.word_list.apply(lambda list_of_words: " ".join([word for word in list_o
f_words if word not in sw]))
wordcloud_real_reqs = WordCloud(width = 1500, height = 1500,
                background_color ='white',
                min_font_size = 15).generate(real.clean_string.sum())
plt.imshow(wordcloud_real_reqs, interpolation='bilinear')
plt.title('Real Job Requirements')
plt.axis("off")
```

Out[1096]:

```
(-0.5, 1499.5, 1499.5, -0.5)
```



Alternatively, what do fake job requirements look like?

In [1097]:

```
fakes = fj[fj['fraudulent'] == 1]
fakes.drop(columns=['telecommuting', 'department', 'has_company_logo','has_questions'])
fakes['requirements'] = fakes['requirements'].astype(str)

fakes['requirements'] = fakes.requirements.apply(lambda x: x.lower())
fakes['requirements'] = fakes.requirements.apply(lambda x: x.translate(str.maketrans(string.punctua
tion, ' '*len(string.punctuation))))
fakes['requirements'] = fakes['requirements'].str.replace('amp', '')
fakes['word_list'] = fakes.requirements.apply(nltk.word_tokenize)
sw = set(stopwords.words('english'))


fakes['word_list'] = fakes.word_list.apply( lambda list_of_words: [word for word in list_of_words i
f word not in sw])
fakes['clean_string'] = fakes.word_list.apply(lambda list_of_words: " ".join([word for word in list
_of_words if word not in sw]))
wordcloud_fakes_reqs = WordCloud(width = 1500, height = 1500,
                background_color ='white',
                min_font_size = 15).generate(real.clean_string.sum())
plt.imshow(wordcloud_fakes_reqs, interpolation='bilinear')
plt.title('Fake Job Requirements')
plt.axis("off")
```

Out[1097]:

```
(-0.5, 1499.5, 1499.5, -0.5)
```

They look the same! I proved nothing! ....and something.

Interestingly, many of the most used words in the fraudulent job listings are the same as the ones used in the real listings. This is how job seekers unwittingly fall into the dangerous trap of false job listings. Really, all a fake employer would have to do is copy and paste a real job listing and edit out minor details. There would be no place for fraudulent job listings if the consumer could easily tell the difference between what is real and what is fake. In the area of job requirements, the fraudulent jobs succeeded in blending in with the other entries.

A few differences that I notice in the job description wordclouds are words such as 'develop', 'manage' rather than 'manager', 'need', and 'social media' being present in the real job listing wordcloud. I especially note 'social media' because it shows the company has or desires an online presence; an applicant can easily search and verify a company's social media in addition to their website.

Who is Being Targeted?

To further our inferences made from the word cloud, let's examine the fields with fraudulent postings. First, this chart examines both real and fraudulent entries across all industries and all entries in this dataset.

Which industries are most affected?

In [1098]:

```
ind_grp=fj.groupby('industry',as_index= False)['fraudulent'].count() # .count counts occurences of all industries, both real and fake, meaning this chart considers all 17,880 entries
alt.Chart(ind_grp, title= "Count of All Industries, Real and Fake").mark_bar().encode(
   x='industry',
   y= 'fraudulent')
```

Out[1098]:

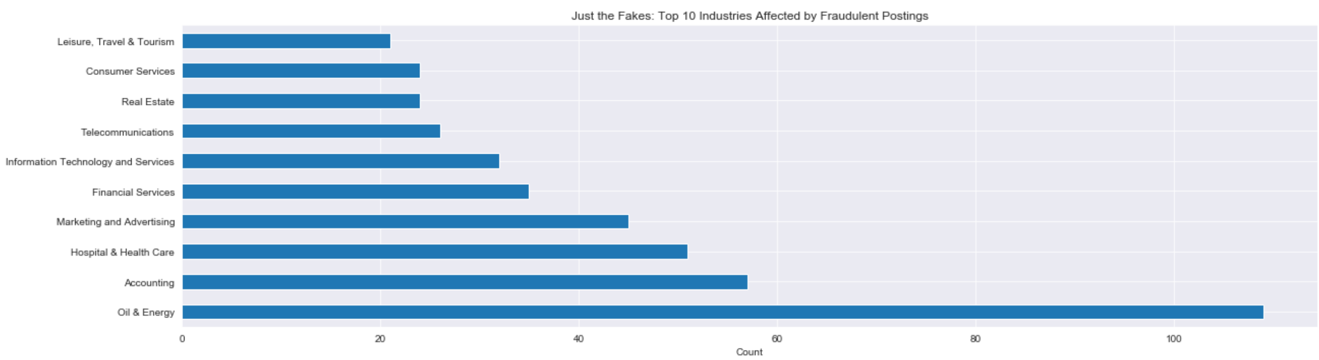
The "Count of All Industries, Real and Fake" chart shows us an overview of all industries and their amount of entries, including both real and fake listings. It is a great visualization tool for the overall count of industry entries in the dataset, but it would be more beneficial to see a comparison between fraudulent listings and real listings since our exploration has been based on attempting to differentiate the two. Furthermore, we should create a chart focusing on the top 10 industries affected in our fraudulent and real datasets so that we are not overwhelmed with the 131 unique industries.

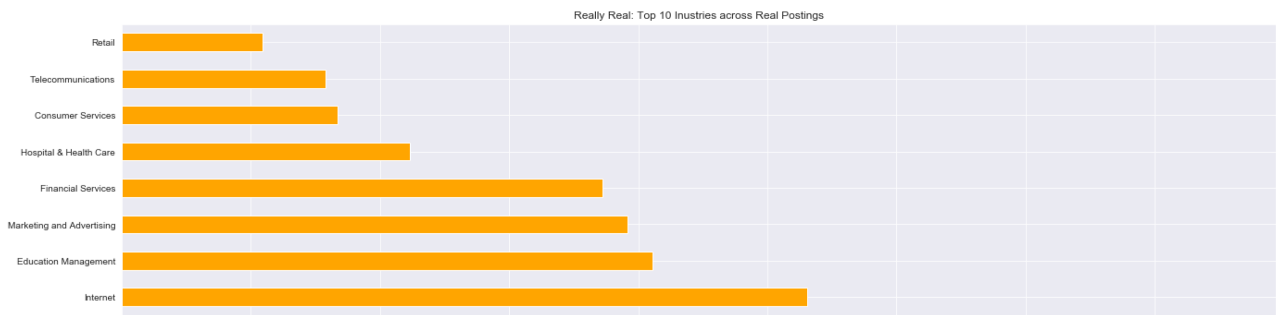
From what we see now, the fields of Computer Softwarre, Consumer Service, Education Management, Financial Services, Hospital & Health Care, Information Technology and Services, Internet, and Marketing and Advertising are the largest spikes on this graph; will our fraudulent listings reflect this overall industry chart?

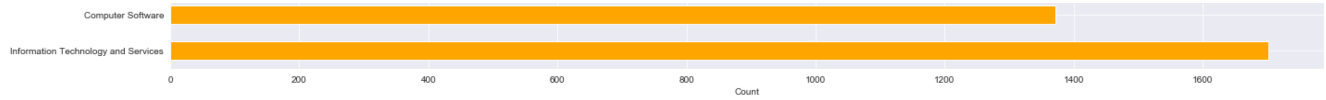
In [1105]:

```
fake_industry = fakes.industry.value_counts() [:10] # Limit to top 15
plt.figure(figsize = (20,12))
plt.subplot(2,1,1)
fake_industry.plot(kind = 'barh')
plt.title('Just the Fakes: Top 10 Industries Affected by Fraudulent Postings')
plt.xlabel('Count')


real_industry = real.industry.value_counts() [:10]
plt.figure(figsize = (20,12))
plt.subplot(2,1,2)
real_industry.plot(kind = 'barh', color= 'orange')
plt.title('Really Real: Top 10 Inustries across Real Postings')
plt.xlabel('Count')
plt.tight_layout()
plt.show()
```







While the top 10 respective industries of both the fake listings and real listings are not the exact same, it is interesting that overall, the top ten list of both are similar. Following this entire dataset's pattern, the comparison of industries affected in *real* and *fake* job postings could show that fraudulent job listings follow job market trends to target its applicants. It seems that fraudulent organizations push out their applications in fields that are of high interest but not in fields that are in so high of an interest that the fraudulent job listings get overlooked.

As we can see from our graphs, Oil & Energy has the most fraudulent entries, while Accounting and Hospital & Healthcare follow its lead. Oil & Energy are the most unexpected and interesting entries on the list because of its niche field, so let's take a closer look at these fake listings.

In [1109]:

```
OE= fakes.loc[fakes['industry'] == 'Oil & Energy']
print("The industry of Oil & Energy has" , len(OE) , "FAKE entries.")
print(" ")
print("For each experience requirement level, here is the amount of times it appears in the FRAUDU
LENT Oil & Energy entries:")
print(" ")
print(OE['required_experience'].value_counts())

OE[['company_profile','location','department',
'industry','required_experience','required_education']].head(15)
```

```
The industry of Oil & Energy has 109 FAKE entries.

For each experience requirement level, here is the amount of times it appears in the FRAUDULENT Oi
l & Energy entries:

Mid-Senior level    37
Entry level          6
Executive            2
Associate            1
Name: required_experience, dtype: int64
```

Out[1109]:

	company_profile	location	department	industry	required_experience	required_education
98	...	US, , Stocton, CA	Oil & Energy	Oil & Energy	Mid-Senior level	High School or equivalent
215	...	US, CA, Bakersfield, CA / Mt. Poso	Oil & Energy	Oil & Energy	Mid-Senior level	High School or equivalent
603	Aker Solutions is a global provider of product...	US, TX, Houston	NaN	Oil & Energy	NaN	NaN
628	Jaco Oil and Refined Resources have partnered ...	US, CA, Bakersfield	Oil & Energy	Oil & Energy	Mid-Senior level	Certification
740	...	US, CA, Bakersfield	HSE (Health Safety Environmental)	Oil & Energy	Mid-Senior level	Bachelor's Degree
812	...	US, CA, Bakersfield	Oil & Energy	Oil & Energy	Mid-Senior level	Certification
814	...	US, CA, Bakersfield	Oil & Energy	Oil & Energy	Mid-Senior level	Bachelor's Degree
825	Process Unlimited and Refined Resources have p...	US, CA, Bakersfield	Oil & Energy	Oil & Energy	Mid-Senior level	Bachelor's Degree
937	Aptitude Staffing Solutions has redesigned the...	US, CA, Bakersfield	Refined Resources	Oil & Energy	Mid-Senior level	Bachelor's Degree
1204	Macpherson Oil and Refined Resources have part...	US, CA, Bakersfield	Oil & Energy	Oil & Energy	Entry level	Bachelor's Degree
1518	Aker Solutions is a global provider of product...	US, TX, Houston	Aker Solutions Inc.	Oil & Energy	NaN	NaN
1655	Aker Solutions is a global provider of product...	US, TX, Houston	NaN	Oil & Energy	NaN	NaN
1813	Aker Solutions is a global provider of product...	US, TX, Houston	NaN	Oil & Energy	NaN	NaN

	company_profile	location	department	industry	required_experience	required_education
1821	Aker Solutions is a global provider of product...	US, TX, Houston	NaN	Oil & Energy	NaN	NaN
1822	Aker Solutions is a global provider of product...	US, TX, Houston	NaN	Oil & Energy	NaN	Bachelor's Degree

For the most part, these entries come from the same company, are in the same area, and have similar experience requirements. And again, its real counterparts look eerily similar.

In [1110]:

```
realOE= real.loc[real['industry'] == 'Oil & Energy']
print("The industry of Oil & Energy has" , len(realOE) , "REAL entries.")
print(" ")
print("For each experience requirement level, here is the amount of times it appears in the REAL O
il & Energy entries:")
print(" ")
print(realOE['required_experience'].value_counts())

realOE[['company_profile','location','department',
'industry','required_experience','required_education']].head(10)
```

```
The industry of Oil & Energy has 178 REAL entries.

For each experience requirement level, here is the amount of times it appears in the REAL Oil & En
ergy entries:

Mid-Senior level    58
Associate           40
Entry level         16
Not Applicable       8
Director             3
Executive            1
Name: required_experience, dtype: int64
```

Out[1110]:

	company_profile	location	department	industry	required_experience	required_education
33	Valor Services provides Workforce Solutions th...	US, CA, San Ramon	NaN	Oil & Energy	NaN	Bachelor's Degree
37	NaN	US, TX, HOUSTON	NaN	Oil & Energy	Mid-Senior level	Bachelor's Degree
182	Valor Services provides Workforce Solutions th...	US, TX, Houston	NaN	Oil & Energy	NaN	NaN
271	Valor Services provides Workforce Solutions th...	US, TX, Houston	NaN	Oil & Energy	NaN	NaN
453	WellAware is an oil field communications and s...	US, TX,	Development	Oil & Energy	Mid-Senior level	Bachelor's Degree
552	Novitex Enterprise Solutions, formerly Pitney ...	US, TX, The Woodlands	NaN	Oil & Energy	Entry level	High School or equivalent
610	Valor Services provides Workforce Solutions th...	US, LA,	NaN	Oil & Energy	NaN	Bachelor's Degree
851	WellAware is an oil field communications and s...	US, TX, Austin/San Antonio	R&D	Oil & Energy	Mid-Senior level	Bachelor's Degree
1372	Valor Services provides Workforce Solutions th...	US, WV, Buckhannon	NaN	Oil & Energy	NaN	NaN
1476	Valor Services provides Workforce Solutions th...	US, PA, Waynesburg	NaN	Oil & Energy	Entry level	High School or equivalent

In a way that mirrors the fake entries, these real listings come from the same company, are in the same area, and have similar experience requirements. Interestingly, both companies that appear heavily on their respective lists are in the same location (Valor Services is a real listing, Aker Solutions is fake, and hoth are in Houston, TX). It is almost to say that if an applicant is denied from Valor Services, they could attempt to apply to its 'competitor,' Aker Solutions.

What are the experience requirements?

In [1102]:

```
fig, ax = plt.subplots(1, 2)

chart = sbn.countplot(x = 'required_experience', data=fj[fj['fraudulent']==0], ax=ax[0])
chart.set_xticklabels(chart.get_xticklabels(), rotation=90)
ax[0].set_title('Required Experience- Real')

chart = sbn.countplot(x = 'required_experience', data=fj[fj['fraudulent']==1], ax=ax[1])
chart.set_xticklabels(chart.get_xticklabels(), rotation=90)
ax[1].set_title('Required Experience- Fake')
plt.show()
```

Why does it matter what the experience requirement is? It may seem simple, but having too high of an experience requirement deters potential victims and too low of one makes potential victims suspicious. I hypothesize that recent college graduates and low-income individuals are most susceptible to this grifting scheme because (1) they do not have enough experience to understand what exactly they should be looking for, (2) they are desperate, and (3) they're "easy" to get. There are certain jobs boards sites that have '1-click apply' options that focus on entry-level, no experience needed job openings, and this has most likely influenced why fake job posts rank so high in that area. It is interesting that the count of entry-level positions and mid-senior level positions are inverse between the fake and real job postings. Could this mean anything important?

Conclusion

The University decided what was real and fake by comparing if job descriptions matched requirements, benefits, and salary, along with the internet presence and track record of the company. From the textual exploration, we saw that fraudulent companies do their research to keep up with technology and market trends. For the University to create this dataset, they used both human detection and machine learning to come to their conclusions. How many of our job boards go through that much effort to ensure the safety of their users? In this underexplored field, users— especially entry-level and mid-senior position seekers in Oil & Energy, STEM fields, and customer service— are at a disadvantage.

If you'd like to specifically learn more about how the Univeristy crafted this dataset, here is where you can find the University's explicit process.

The Ethics of Job Listings

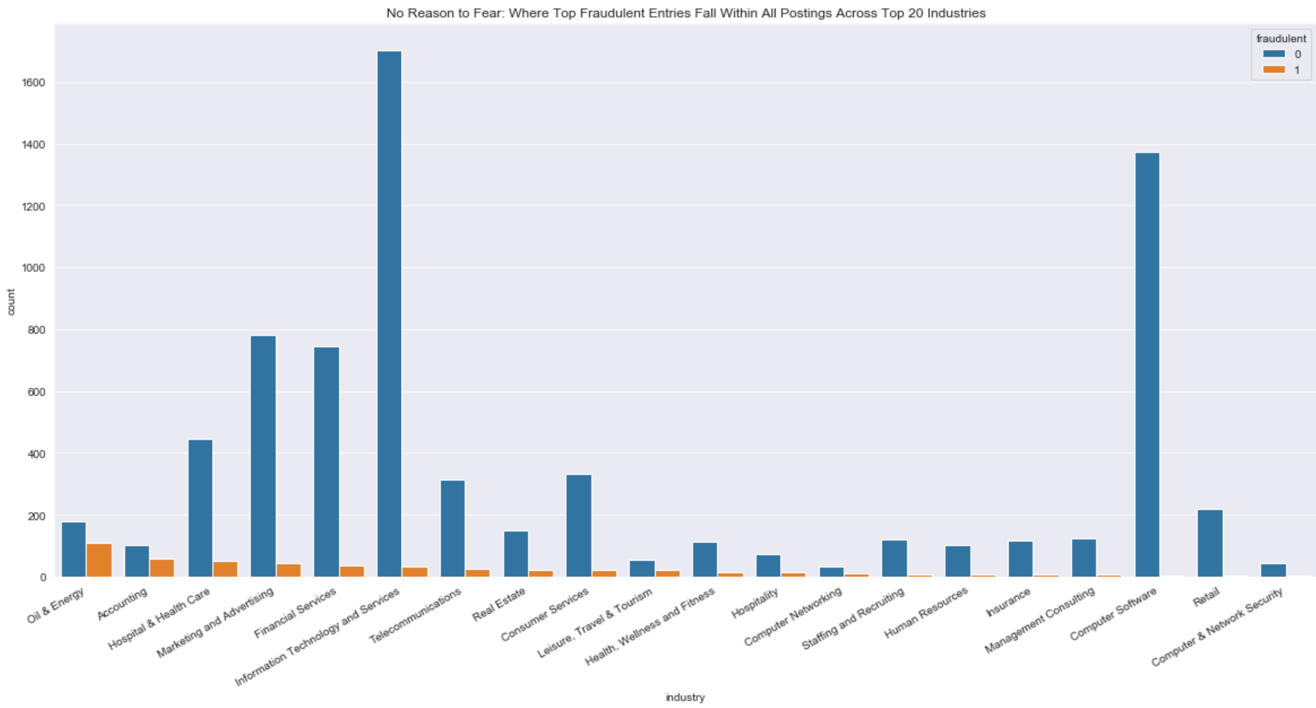
In one effortless click towards a seemingly unchallenging job, individuals release their emails, phones numbers, addresses, and whatever else sensitive information is on their profile and resume and open themselves up to a world of data phishing, scam emails, and loss of security. In a *worst case scenario*, fraudulent job listings could lead to a job interiew that seems real but is actually a funnel for human trafficking. Usually, however, it's just a way to collect our information. Do we realize how much information and password keys we release when we submit resumes that contain home addresses, phone numbers, previous schools, and a plethora of other details?

Should you fear the risk of applying to a fraudulent listing then avoid applying for all jobs posted on job boards? Well, not if you need a job. There is no reason to fear— especially when you acknowledge the difference between fraudulent and real job listings within the dataset:

In [1103]:

```
sbn.set_style('darkgrid')
plt.figure(1,figsize=(20,10))
sbn.countplot(hue=fj.fraudulent,x=fj.industry, order=fakes.industry.value_counts().iloc[:20].index)
plt.title('No Reason to Fear: Where Top Fraudulent Entries Fall Within All Postings Across Top 20 Industries')
plt.gcf().autofmt_xdate()

#remember, 1 means the entry is fake
```



However, the presence of fraudulent jobs in this dataset- no matter how small it might be- is not insignificant data. This dataset was taken from 2012-2014, and the realm of online applications has rapidly expanded since then. The job market has experienced great changes between now and when this dataset was created, so we can only infer from our previous examinations that fraudulent job listings have adapted to match those changes. Furthermore, this datasets' entries were compiled from one job boards site, Workable, and did not take into account the many others that existed at the time. Now, even more of these sites for employment seekers are present on the Internet.

This dataset serves as a reminder to err on the side of caution when it comes to any release of information. There is always a reason to be skeptical when a job listing seems *too* perfect. It would be beneficial to take the same steps that this dataset did in its research: analyze the job listing to see if its benefits and requirements match the position, investigate the company's reputablity, and be wary if you are asked for certain sensitive information.

For the future:

The University notes that it their own research is only a very small stepping stone in a field of unexplored and overlooked fraud. Knowing this, how could we go about adding to the field of research, or at least asking questions, when it comes to companies that look real with real job descriptions but are actually multi-level marketing and pyramid schemes? How can we regulate online jobs boards?