Job Fiction - Indexing Jobs Data to Build a Training Model

Objective

Objective of this notebook is to build a training model based on JOBFICTION database, a collection of job posts, job titles, company, location, job post URL acquired from Indeed Web Services API. Using the training model, we will be able to predict right job title based on the job descriptions passed to the model. Output from the training model would include - a corpus based on vector space model, key words and phrases, skill identifiers, predicted job titles and corresponding scores. All the results will be persisted and updated with the new jobs being collected.

Based on the input from job seekers i.e. job descriptions submitted we will able to determine

- Job titles closest to the job description or keywords submitted (based on the weights associated)
- Recommended job posts
- Keywords to search for the right job posts

The first part of this notebook will explore how jobs in the JOBFICTION database can be classified.

Why do we have to classify the job posts?

A `truck driver` job post is way different from a `database administrator` job post. With the help of clustering algorithms we categorize similar jobs into same cluster based purely on the job description. Similar to movie genres this classifier is expected to create job categories based on similarity of job descriptions. We can then study the job titles under the same cluster to see how true clusters. Since there is no training data set available we resort to unsupervised clustering and the challenge is to define the number of clusters.

We focus only on the data related job posts i.e. job posts with the word "data" in either job title or job description.

Approach

- Export job descriptions, job title, company and job id from JOBFICTION database
- Remove stop words
- Tokenize and stem each job description
- Transforming the corpus into vector space using tf-idf
- Clustering the documents using the k-means algorithm
- Plot the clusters
- Using multidimensional scaling to reduce dimensionality within the corpus (LSI)
- Topic modeling using Latent Dirichlet Allocation (LDA)
- Named entity recognition against occupation skills and title taxonomies to identify skills

(Future Work)

- Hierarchical clustering on the corpus using Ward clustering (http://en.wikipedia.org/wiki/Ward%27s_method)
- Plot the clusters with hierarchial clustering

Imports

In [1]:

```
%matplotlib inline

from nltk.tokenize import RegexpTokenizer
from nltk.stem.porter import PorterStemmer
from nltk.stem.snowball import SnowballStemmer
from stop_words import get_stop_words
from nltk.corpus import stopwords
from gensim import corpora, models, similarities
from sklearn.cluster import KMeans, MiniBatchKMeans
from collections import Counter
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
from wordcloud import WordCloud
import logging
import random
import gensim
import nltk
import re
import os
```

In [2]:

```
logging.basicConfig(format='%(asctime)s : %(levelname)s : %(message)s', level=lo
gging.INFO)
```

Configuration

In [3]:

```
DATA_DIR = os.path.join("/home", "rt", "wrk", "jobs", "data")
MODEL_DIR = os.path.join("/home", "rt", "wrk", "jobs", "models")
```

1. Export data from JOBFICTION database

Let's extract jobs from JOBFICTION database

In the jobs table, job description is an array of sentences. In order to export job description, this mongo javasript will be run to combine array elements as a string. For traceback we will add __id field to every record.

In [7]:

```
%%writefile export_jobs_with_title.js
db.jobs.find({"summary": /data/}, { _id: 1, jobtitle: 1, company: 1, summary: 1}
).forEach( function (x)
    {
        var jobdesc = '';
        var s = ''
        x.summary.forEach( function (y) {
            s = y.replace(new RegExp('\r?\n','g'), ' ').replace(new RegExp('[|]'
,'g'), '');
            jobdesc += s + ' ';
        });
        print(x._id + "|" + x.jobtitle + "|" + x.company + "|" + jobdesc);
    });
```

```
Overwriting export_jobs_with_title.js
```

In [4]:

```
!mkdir ./data ./models
```

```
mkdir: cannot create directory ‘./data’: File exists
```

Run export script to dump data to text file

In [8]:

```
!time mongo JOBFICTION --quiet export_jobs_with_title.js > ./data/export_jobs_w_
title.txt
```

```
real    4m54.169s
user    0m46.439s
sys     0m3.300s
```

In [9]:

```
!wc -l ./data/export_jobs_w_title.txt
!head -1 ./data/export_jobs_w_title.txt
```

```
144554 ./data/export_jobs_w_title.txt
indeed_6ed966da9f33ffc1|Associate|Potbelly Sandwich Shop|Presidentia
l Towers!!!!!! A Potbelly Associateâs job is to make our customers r
eally happy. Since they are the primary point of customer contact, i
t is up to them to provide our customers and excellent experience by
providing fast, friendly and efficient service and by delivering a q
uality and consistent product every time, in a clean and inviting en
vironment. Essential ï§ Demonstrates and reinforces Potbellyâs Behav
iors and Valuesâ Integrity, Food Loving, Teamwork, Accountability, P
ositive Energy, Coaching, Delivering Results through Execution, Buil
ding and Inspiring Teams, Creating Potbelly âFansâ-- through all int
eractions. ï§ Ability to discuss Potbelly history with others. ï§ Pr
epare quality finished products (sandwiches, salads, soups, cookies,
```

ice cream, etc.) efficiently per Potbelly recipe manual standards. ï § Comply with health and safety standards for food, cleanliness and safety of shop. ï§ Maintain personal hygiene standards, including we aring clean Potbelly uniform. ï§ Comply with established food safety requirements and practices. ï§ Comply with shop security and safety standards. ï§ Be speedy and accurate in fulfilling orders. ï§ Handle raw and finished waste according to established procedures. ï§ Make customers really happy. ï§ Engage in friendly conversation with cust omers in line. ï§ Act with a sense of urgency toward all customers i n the shop. Other Key Functions ï§ Restock food line, chips and cool er. ï§ ï§ Work multiple stations (load, dress, shakes, cash, prep, f ront) as directed by Manager. ï§ Deliver catering orders as detailed in the Catering Driver and Delivery Agreement. ï§ Clean tables, coun ters, floors, bathrooms, kitchen and utensils; take out trash. ï§ Op erate cash register: handle, balance and follow all cash handling pr ocedures. ï§ Effectively handle customer complaints/issues. ï§ Take catering and delivery orders over the phone. ï§ PHYSICAL FUNCTIONS ï § Ability to stand/walk a minimum of 3 hours or as needed. ï§ Must b e able to exert well-paced and frequent mobility for periods of up t o 3 hours or as needed. ï§ Be able to lift up to 10 pounds frequentl y. ï§ Will frequently reach, feel, bend, stoop, carry, finely manipu late and key in data. ï§ Able to work in both warm and cool environm ents, indoors (95%) and outdoors (5%). ï§ Must be able to tolerate h igher levels of noise from music, customer and employee traffic. ï§ Must be able to tolerate potential allergens: peanut products, egg, dairy, gluten, soy, seafood and shellfish. EXPERIENCE, EDUCATION AND BEHAVIORS ï§ Must represent Potbelly Advantage and Our Values. ï§ Mu st be at least 16 years of age ï§ For Illinois employees, all employ ees are required to become food safety certified within 30 days of e mployment. Failure to do so will result in termination of employment . ï§ Must be friendly and customer service-oriented. ï§ Strong verba l communication skills. ï§ Must possess neat and clean hygiene. ï§ A bility to handle a knife confidently. ï§ Must be able to work in a f ast-paced environment and have a sense ofï§ Ability to work as a tea m-player. ï· Ability to comprehend and communicate in English via ve rbal and written communication, such that employee can perform his o r her job responsibilities. ï§ Must demonstrate leadership behaviors and values that align with Potbelly urgency. Potbelly.Com/Careers Jo b Type: Part-time Local candidates only: Chicago, IL 60661 Required education: High school or equivalent

2. Create training data set

We will export random 10K job descriptions as training data set. We will use unsupervised clustering to see how similar job descriptions are. based on clusters we can do topic modeling with LDA for each cluster. We can keep updating the model with new job posts.

Below sort to be optimized by randomized only job ids instead of entire text.

In [10]:

```
!time sort -t'|' -k1 -R ./data/export_jobs_w_title.txt | head -10000 > ./data/tr
ain_w_complete_text.txt
```

```
sort: write failed: standard output: Broken pipe
sort: write error

real    8m32.986s
user    8m30.838s
sys     0m2.053s
```

In [11]:

```
!time awk -F'|' 'BEGIN{OFS="|"}{print $1, $2, $3}'
./data/train_w_complete_text.txt > ./data/train_labels.txt
!time awk -F'|' 'BEGIN{OFS="|"}{print $4}' ./data/train_w_complete_text.txt > ./
data/train.txt
```

```
real    0m0.570s
user    0m0.068s
sys     0m0.021s

real    0m2.101s
user    0m1.167s
sys     0m0.125s
```

In [12]:

```
!head ./data/train_labels.txt
```

```
indeed_6d13e1749c444e23|Financial Examiner (EL)|GA Dept of Banking &
Finance
indeed_6d16914061219ee4|Analytics Payer/Provider Healthcare Analytic
s Manager|PRICE WATERHOUSE COOPERS
indeed_50c9ebbfb19f9ed7|Aircraft Maintenance Analyst|Ronkonkoma, NY
indeed_6d1fbfcd14cf79e9|Operations Center Representative - All Shift
s|Ascent LLC.
indeed_9a61d5c6de9dec4b|Administrator, Payroll|Community Action Proj
ect
indeed_53c5e81c18aa4202|Project Coordinator/Data Analyst|The Fund fo
r Public Health in New York, Inc.
indeed_bf4b755eadef6b10|Plant Manager|IEC Holden Inc.
indeed_e5ee1725b888eeb0|IT Infrastructure & Security Manager|Collibr
a
indeed_08b4c32dcb730ba2|Material Control Specialist l|PRIMUS
indeed_3aede0ed8048b044|Licensed Financial Advisor|Scient Federal Cr
edit Union
```

In [13]:

```
!tail -1 ./data/export_jobs_w_title.txt > ./data/test_w_complete_text.txt
!awk -F'|' 'BEGIN{OFS="|"}{print $1, $2, $3}' ./data/test_w_complete_text.txt >
./data/test_labels.txt
!awk -F'|' 'BEGIN{OFS="|"}{print $4}' ./data/test_w_complete_text.txt > ./data/t
est.txt
```

In [17]:

```
!head -2 ./data/train.txt | tail -1 > ./data/sample.txt
```

3. Cleansing Data - Stop words, Tokenizing and Stemming

Failing to cleanse and normalize the data properly can decrease the overall effectiveness of the model. Let's define few functions before we take off

In [8]:

```
# replace forward and back slash, hyphen, underscores and other characters
def preprocess(text):
    clean = text
    clean = re.sub("[/_-]", " ", clean)
    clean = re.sub("[^a-zA-Z.+3]", " ", clean) # get rid of any terms that aren'
t words
    return clean
```

In [9]:

```
# define a tokenizer and stemmer to returns the set of stems in the text passed

def tokenize_and_stem(text):
    # tokenize by sentence, then by word to catch any punctuations
    tokens = [word.lower() for sent in nltk.sent_tokenize(text) for word in nltk
.word_tokenize(sent)]
    filtered_tokens = []

    # remove stop words from tokens
    en_stop = set(get_stop_words('en') + stopwords.words("english"))
    stopped_tokens = [i for i in tokens if not i in en_stop]

    # filter out tokens not containing alphanumeric
    for token in stopped_tokens:
        if re.search('[a-zA-Z]', token):
            filtered_tokens.append(token)

    stems = [stemmer.stem(t) for t in filtered_tokens]

    return stems

def tokenize_only(text):
    # tokenize by sentence, then by word to catch any punctuations
    tokens = [word.lower() for sent in nltk.sent_tokenize(text) for word in nltk
.word_tokenize(sent)]
    filtered_tokens = []

    # remove stop words from tokens
    en_stop = set(get_stop_words('en') + stopwords.words("english"))
    stopped_tokens = [i for i in tokens if not i in en_stop]

    # filter out tokens not containing alphanumeric
    for token in stopped_tokens:
        if re.search('[a-zA-Z]', token):
            filtered_tokens.append(token)

    return filtered_tokens
```

In [10]:

```
# create p_stemmer of class SnowballStemmer
stemmer = SnowballStemmer("english")
```

Read training data

In [11]:

```
# compile training docs into a list
train = [ preprocess(line.decode('unicode_escape').encode('ascii', 'ignore')) fo
r line in open(os.path.join(DATA_DIR, 'train.txt'), 'r') ]
```

In [12]:

```
# compile training labels for tracking and debugging purposes only
train_labels = [ line.strip('\n').split('|') for line in open(os.path.join(DATA_
DIR, 'train_labels.txt'), 'r') ]
```

In [13]:

```
train_labels[0]
```

Out[13]:

```
['indeed_6d13e1749c444e23',
 'Financial Examiner (EL)',
 'GA Dept of Banking & Finance']
```

Creating persistent files with words (i) tokenized and stemmed and (ii) tokenized separetely.

In [16]:

```
FILE_STEM = os.path.join(DATA_DIR, 'train_stem.txt')
FILE_TOKEN = os.path.join(DATA_DIR, 'train_token.txt')
```

Calling tokenizer and stemmer functions on the training data

In [17]:

```
f_stem = open(FILE_STEM, 'w')
f_token = open(FILE_TOKEN, 'w')

for jobdesc in train:
    stemmed = tokenize_and_stem(jobdesc)
    f_stem.write(' '.join(stemmed).encode('utf-8').strip() + '\n')

    tokenized = tokenize_only(jobdesc)
    f_token.write(' '.join(tokenized).encode('utf-8').strip() + '\n')
```

4. Bag-of-Words (BoW) Corpus & Dictionary

Creating Dictionary

In [25]:

```
%time
dictionary = corpora.Dictionary([line.lower().split() for line in open(FILE_TOKE
N)])
dictionary.compactify()
dictionary.save(os.path.join(MODEL_DIR, "train_jobs.dict"))
print(dictionary)
```

```
CPU times: user 0 ns, sys: 0 ns, total: 0 ns
Wall time: 8.82 µs
Dictionary(47674 unique tokens: [u'fawn', u'nordisk', u'raining', u'
environments.investment', u'prologistix']...)
```

Corpus

For scalability reason, using iterator to stream job description one by one instead of reading all jobs at a time in memory

Each document in the tokenized file is converted to bag-of-words model before storing as a corpus

In [26]:

```
class jobCorpus(object):
    def __iter__(self):
        for line in open(FILE_TOKEN):
            # assume there's one document per line, tokens separated by whitespa
ce
            yield dictionary.doc2bow(line.lower().split())
```

In [29]:

```
jobs_corpus = jobCorpus()
corpora.MmCorpus.serialize(os.path.join(MODEL_DIR, "train_jobs.mm"), jobs_corpus
)
```

In [30]:

```
corpus = corpora.MmCorpus(os.path.join(MODEL_DIR, "train_jobs.mm"))
print corpus
```

```
MmCorpus(10000 documents, 47674 features, 2170358 non-zero entries)
```

5. Dimensionality Reduction using Latent Semantic Indexing

Since we do not know how many topics this corpus should yield so we decided to compute this by reducing the features to n = 10 dimensions, then clustering the points for different values of K (number of clusters) to find an optimum value. Gensim offers various transforms that allow us to project the vectors in a corpus to a different coordinate space. One such transform is the Latent Semantic Indexing (LSI) transform, which we use to project the original data to 50D.

In [55]:

```
MAX_LSI_TOPICS = 10
```

In [31]:

```
%%time
dictionary = corpora.Dictionary.load(os.path.join(MODEL_DIR, "train_jobs.dict"))
corpus = corpora.MmCorpus(os.path.join(MODEL_DIR, "train_jobs.mm"))

tfidf = models.TfidfModel(corpus, normalize=True)
corpus_tfidf = tfidf[corpus]

# reduce the vector space by projecting  to 10 dimensions
lsi = gensim.models.LsiModel(corpus_tfidf, id2word=dictionary, num_topics = MAX_
LSI_TOPICS)
```

```
CPU times: user 1min 56s, sys: 5.14 s, total: 2min 2s
Wall time: 2min 15s
```

In [54]:

```
# write coordinates to file
fcoords = open(os.path.join(MODEL_DIR, "train_jobs_lsi_coords.csv"), 'wb')
for vector in lsi[corpus]:
    if len(vector) != MAX_LSI_TOPICS:
        continue
    v = '\t'.join([ "{:6.6f}".format(x[1]) for x in vector ])
    fcoords.write(v + '\n')
fcoords.close()
```

In [58]:

```
!wc -l ./models/train_jobs_lsi_coords.csv
!head -2 ./models/train_jobs_lsi_coords.csv
```

```
10000 ./models/train_jobs_lsi_coords.csv
5.612125        -0.142342       -0.005066       1.373977
-4.532514       4.445029        4.274249        1.326633
0.668706        -1.276696
12.383553       5.060576        -2.666577       0.423928
3.141846        1.353790        0.326339        0.964861
-0.058433       0.539940
```

6. K-Means Clustering

Next we clustered the points in the reduced dimension LSI space using K-Means, varying the number of clusters (K) from 1 to 50. The objective function used is the Inertia of the cluster, defined (http://scikit-learn.org/stable/modules/clustering.html#k-means) as the sum of squared differences of each point to its cluster centroid. This value is fed from Scikit-Learn K-Means algorithm.

Reference:

- Stackoverflow (http://stackoverflow.com/questions/6645895/calculating-the-percentage-of-variance-measure-for-k-means)
- Data science central post by Vincent Granville (http://www.analyticbridge.com/profiles/blogs/identifying-the-number-of-clusters-finally-a-solution)

Determine Number of Topics

In [42]:

```
MAX_K = 100
```

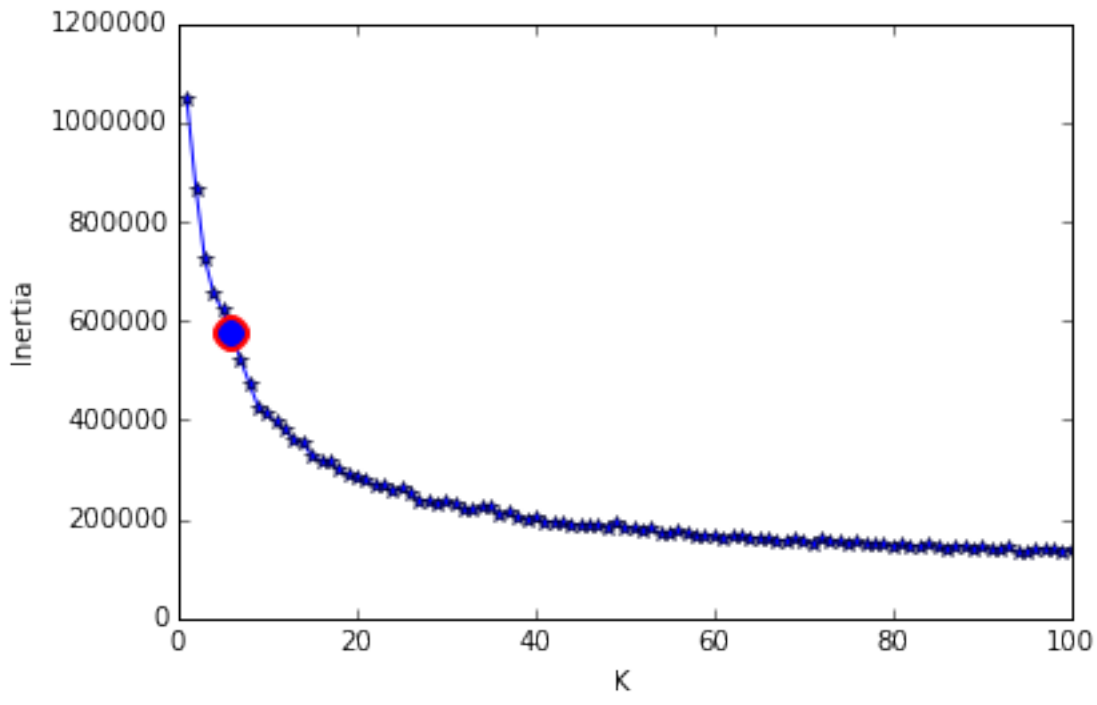
In [43]:

```
X = np.loadtxt(os.path.join(MODEL_DIR, "train_jobs_lsi_coords.csv"), delimiter="
\t")
ks = range(1, MAX_K + 1)

inertias = np.zeros(MAX_K)
diff = np.zeros(MAX_K)
diff2 = np.zeros(MAX_K)
diff3 = np.zeros(MAX_K)
```

In [45]:

```
for k in ks:
    #kmeans = KMeans(k).fit(X)
    kmeans = MiniBatchKMeans(n_clusters=k, init='k-means++', n_init=1, init_size
=1000, batch_size=1000).fit(X)
    inertias[k - 1] = kmeans.inertia_
    # first difference
    if k > 1:
        diff[k - 1] = inertias[k - 1] - inertias[k - 2]
    # second difference
    if k > 2:
        diff2[k - 1] = diff[k - 1] - diff[k - 2]
    # third difference
    if k > 3:
        diff3[k - 1] = diff2[k - 1] - diff2[k - 2]

elbow = np.argmin(diff3[3:]) + 3
print elbow

plt.plot(ks, inertias, "b*-")
plt.plot(ks[elbow], inertias[elbow], marker='o', markersize=12,
         markeredgewidth=2, markeredgecolor='r', markerfacecolor=None)
plt.ylabel("Inertia")
plt.xlabel("K")
plt.show()
```

5



We plotted the inertias for different values of K from 1 to 100. Using the approach of calculating the third differential to find an elbow point, the elbow point happens here for K=6 or 7 and is marked with a red dot

In [58]:

```
from pandas.tools.plotting import scatter_matrix
X = np.loadtxt(os.path.join(MODEL_DIR, "train_jobs_lsi_coords.csv"), delimiter="
\t")
df = pd.DataFrame(X, columns=range(10))
```

In [90]:

```
NUM_TOPICS = 5

X = np.loadtxt(os.path.join(MODEL_DIR, "train_jobs_lsi_coords.csv"), delimiter="
\t")
kmeans = MiniBatchKMeans(n_clusters=NUM_TOPICS, init='k-means++', n_init=1, init
_size=1000, batch_size=1000).fit(X)
y = kmeans.labels_

colors = [ "peru", "dodgerblue", "brown", "darkslategray", "lightsalmon", "orang
e", "springgreen", "orangered", "yellow", "firebrick" ]
```

In [78]:

```
Counter(y)
```

Out[78]:

```
Counter({0: 3994, 1: 107, 2: 1968, 3: 197, 4: 3734})
```

In [92]:

```
#Plotting

df = pd.DataFrame(X, columns=range(10))
scatter_matrix(df, figsize=(50,50), alpha=0.2, marker='.', c=colors, diagonal=No
ne, edgecolors='None')

#for j in range(10):
#    for k in range(10):
#        if j < k:
#            plt.figure(figsize=(10,10))
#            plt.title("Scatter plot for ({}, {})".format(j, k))
#            for i in range(X.shape[0]):
#                plt.scatter(X[i][j], X[i][k], c=colors[y[i]], s=10)
#            plt.show()
```

Out[92]:

```
array([[<matplotlib.axes._subplots.AxesSubplot object at 0x7ff05eefa
690>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff0603e6
350>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05e32b
350>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff049a53
```

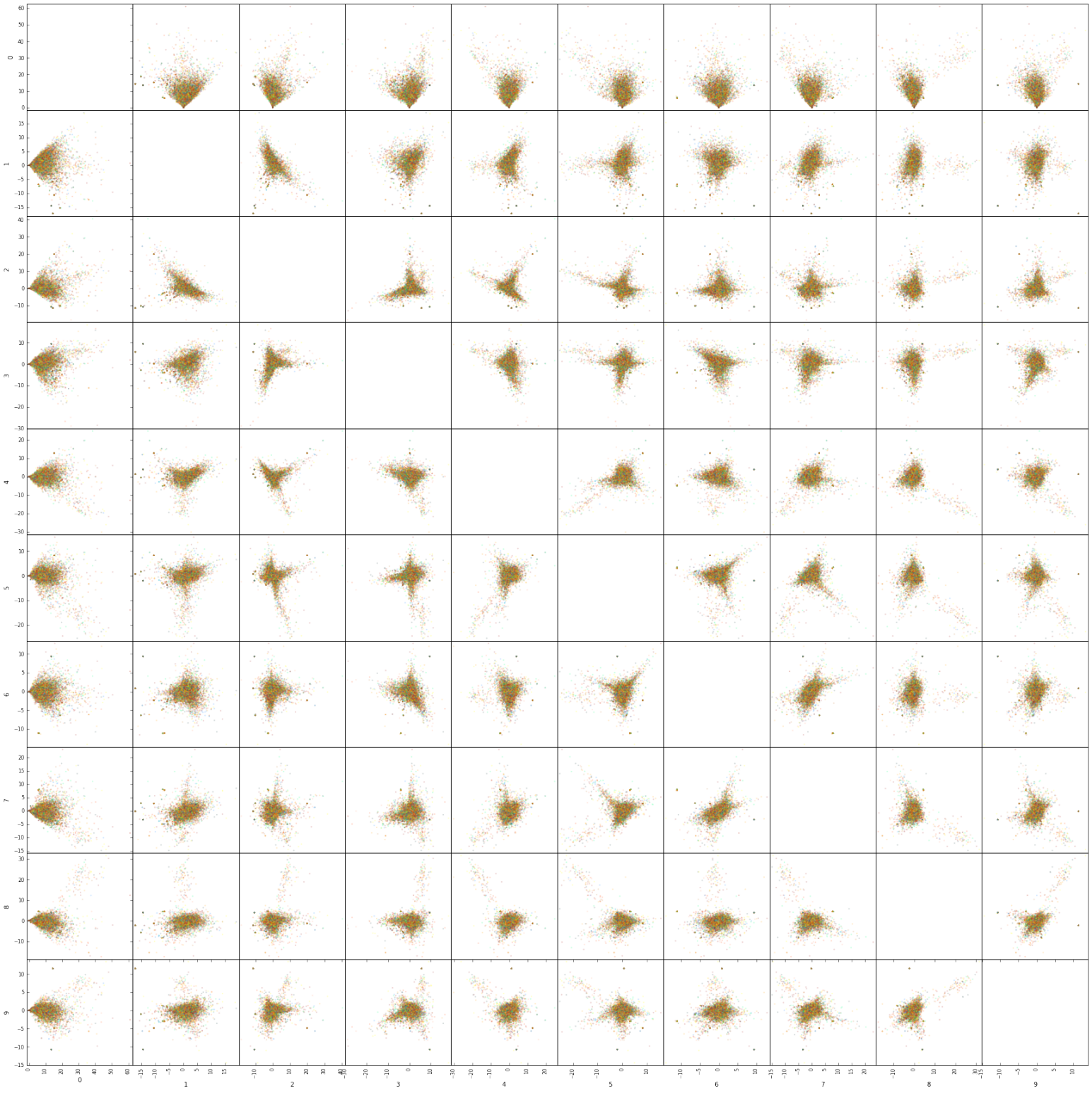
```
090>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff049a8e
d90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff049a16
810>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05e24f
d50>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05e1c7
e90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05e0a3
e90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05dfa1
c90>],
       [<matplotlib.axes._subplots.AxesSubplot object at 0x7ff05def9
650>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05ddc6
890>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05dd4b
5d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05cc8e
f10>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05cc21
090>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05cb7a
b10>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05cafa
d50>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05cba8
f10>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05ca6f
350>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c974
2d0>],
       [<matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c957
890>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c89b
5d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c801
050>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c777
ed0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c6fd
c10>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c6eb
850>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c670
7d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c5d7
510>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c499
550>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c545
cd0>],
```

```
       [<matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c37f
d10>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c304
c90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c273
290>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c26a
f90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c1c6
a10>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c156
8d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c0d9
610>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05ef84
f10>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff060b6f
310>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff061a36
590>],
       [<matplotlib.axes._subplots.AxesSubplot object at 0x7ff062ede
c10>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05c06b
a90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057fa5
550>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057f22
4d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057e87
210>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057e0a
250>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057eb4
090>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057d71
a10>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057cf7
990>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057c59
f50>],
       [<matplotlib.axes._subplots.AxesSubplot object at 0x7ff057bde
c90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057b45
710>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057ac8
5d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057a4d
310>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057a3d
050>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff0579b1
c50>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057919
```

ad0>,

        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05789c
810>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05793d
fd0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057790
410>],
       [<matplotlib.axes._subplots.AxesSubplot object at 0x7ff057715
050>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff0576f8
690>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05767b
3d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff0575ca
fd0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057557
e90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff0574db
ad0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05744a
910>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff0573d0
550>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff0573b6
3d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05733a
110>],
       [<matplotlib.axes._subplots.AxesSubplot object at 0x7ff057359
d90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05721f
cd0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff0571a5
910>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057109
f50>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05708d
c90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff057066
8d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056ff6
790>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056f79
3d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056ee8
210>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056e61
e10>],
       [<matplotlib.axes._subplots.AxesSubplot object at 0x7ff056dc6
c90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056d49
9d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056deb
7d0>,

```
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056cbd
5d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056bc2
210>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056ba5
850>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056b29
590>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056a8f
1d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056a13
090>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff05698b
c90>],
       [<matplotlib.axes._subplots.AxesSubplot object at 0x7ff05697b
ad0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056880
710>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056865
590>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff0567e7
2d0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056807
110>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff0566cd
e90>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056652
ad0>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff0565c6
150>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff0565bb
e50>,
        <matplotlib.axes._subplots.AxesSubplot object at 0x7ff056513
a90>]], dtype=object)
```



7. Topic Modeling using LDA

In [52]:

```
%%time
dictionary = corpora.Dictionary.load(os.path.join(MODEL_DIR, "train_jobs.dict"))
corpus = corpora.MmCorpus(os.path.join(MODEL_DIR, "train_jobs.mm"))

# Project to LDA space
NUM_TOPICS = 7
lda = gensim.models.LdaModel(corpus, id2word=dictionary, num_topics=NUM_TOPICS,
                             chunksize=2000,
                             passes=20,
                             alpha='auto',
                             eval_every=10,
                             minimum_probability=0.01
                            )
```

```
CPU times: user 26min 55s, sys: 3.91 s, total: 26min 59s
Wall time: 27min 16s
```

Topic Terms

In [53]:

```
lda.print_topics(NUM_TOPICS, 50)[0]
```

Out[53]:

```
(0,
 u'0.020*experience + 0.014*data + 0.011*systems + 0.010*management
+ 0.010*support + 0.009*technical + 0.009*project + 0.009*business +
0.007*skills + 0.007*knowledge + 0.007*years + 0.007*requirements +
0.006*software + 0.006*system + 0.006*work + 0.006*security + 0.006*
information + 0.006*development + 0.006*required + 0.006*analysis +
0.005*design + 0.005*ability + 0.005*related + 0.004*team + 0.004*so
lutions + 0.004*technology + 0.004*projects + 0.004*engineering + 0.
004*degree + 0.004*processes + 0.004*including + 0.004*provide + 0.0
04*working + 0.003*process + 0.003*testing + 0.003*applications + 0.
003*issues + 0.003*test + 0.003*services + 0.003*strong + 0.003*job
+ 0.003*application + 0.003*network + 0.003*analyst + 0.003*tools +
0.003*environment + 0.003*must + 0.003*quality + 0.003*complex + 0.0
03*database')
```

In [54]:

```
ftopics = open(os.path.join(MODEL_DIR, "train_jobs_topics.txt"), 'wb')
for t in lda.print_topics(NUM_TOPICS, 50):
    ftopics.write(str(t[0]) + ':' + t[1] + '\n')
ftopics.close()
```

Job Topics

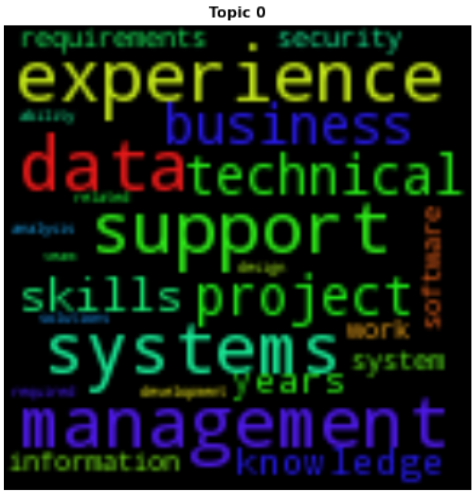
In [55]:

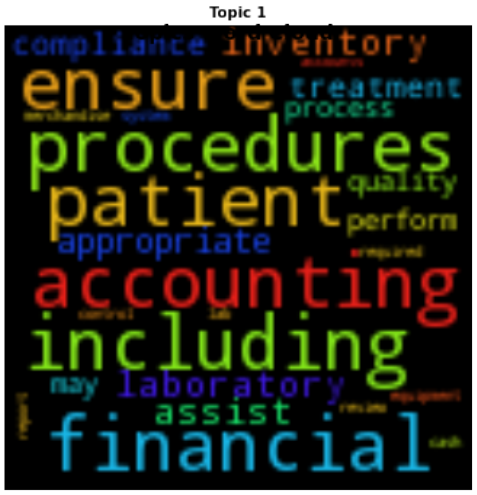
```
fjobtopics = open(os.path.join(MODEL_DIR, "train_jobs_topics.csv"), 'wb')
for doc_id in range(len(corpus)):
    docbow = corpus[doc_id]
    doc_topics = lda.get_document_topics(docbow)
    for topic_id, topic_prob in doc_topics:
        fjobtopics.write("%d\t%d\t%.3f\n" % (doc_id, topic_id, topic_prob))
fjobtopics.close()
```

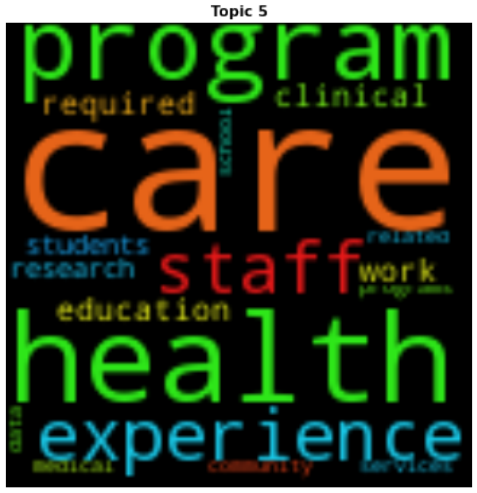
Topic wordcloud representation for analysis

In [65]:

```
final_topics = open(os.path.join(MODEL_DIR, "train_jobs_topics.txt"), 'rb')
number_of_subplots=NUM_TOPICS
v = 0
fig = plt.figure(figsize=(15,15))
fig.subplots_adjust(top = 0.85)

for line in final_topics:
    line = line.strip('\n')
    curr_topic = line.split(':')[0]
    topic_scores = ''.join(line.split(':')[1:])

    scores = [float(x.split("*")[0]) for x in topic_scores.split(" + ")]
    words = [x.split("*")[1] for x in topic_scores.split(" + ")]

    freqs = []
    for word, score in zip(words, scores):
        freqs.append((word, score))

    elements = WordCloud(width=120, height=120).fit_words(freqs)

    v += 1
    ax1 = fig.add_subplot(int(NUM_TOPICS/3)+1, 3, v)
    ax1.set_title("Topic {}".format(curr_topic), fontsize=10, fontweight='bold')
    ax1.imshow(elements)
    ax1.axis("off")

fig.suptitle("Topics Word Cloud", fontsize=14, fontweight='bold')
plt.tight_layout()
plt.show()
final_topics.close()
```







Topic Probability Distribution for Given List of jobs

In [57]:

```
NUM_TOPICS = 7
```

In [58]:

```
topic_df = pd.read_csv(os.path.join(MODEL_DIR, "train_jobs_topics.csv"), sep="\t
",
                       names=["doc_id", "topic_id", "topic_prob"],
                       skiprows=0)

#doc_ids = []
#for i in range(6):
#    doc_ids.append(int(random.random() * max_doc_id))

def plot_job_distr(df, search_job_ids, train_labels):
    job_idx = [ x[0] for x in train_labels ]

    for job_id in search_job_ids:
        index = job_idx.index(job_id)
        filt = df[df["doc_id"] == index]
        topic_ids = filt["topic_id"].tolist()
        topic_probs = filt["topic_prob"].tolist()
        prob_dict = dict(zip(topic_ids, topic_probs))

        ys = []
        for i in range(NUM_TOPICS):
            if prob_dict.has_key(i):
                ys.append(prob_dict[i])
            else:
                ys.append(0.0)

        plt.title("Job ID: {}; Title: {}".format(train_labels[index][2], train_l
abels[index][0]))
        plt.ylabel("P(topic)")
        plt.ylim(0.0, 1.0)
        plt.xticks(range(NUM_TOPICS), ["Topic#%d" % (x) for x in range(NUM_TOPIC
S)])
        plt.grid(True)
        plt.bar(range(NUM_TOPICS), ys, align="center")
        plt.show()
```

In [91]:

```
topic_df.head()
```

Out[91]:

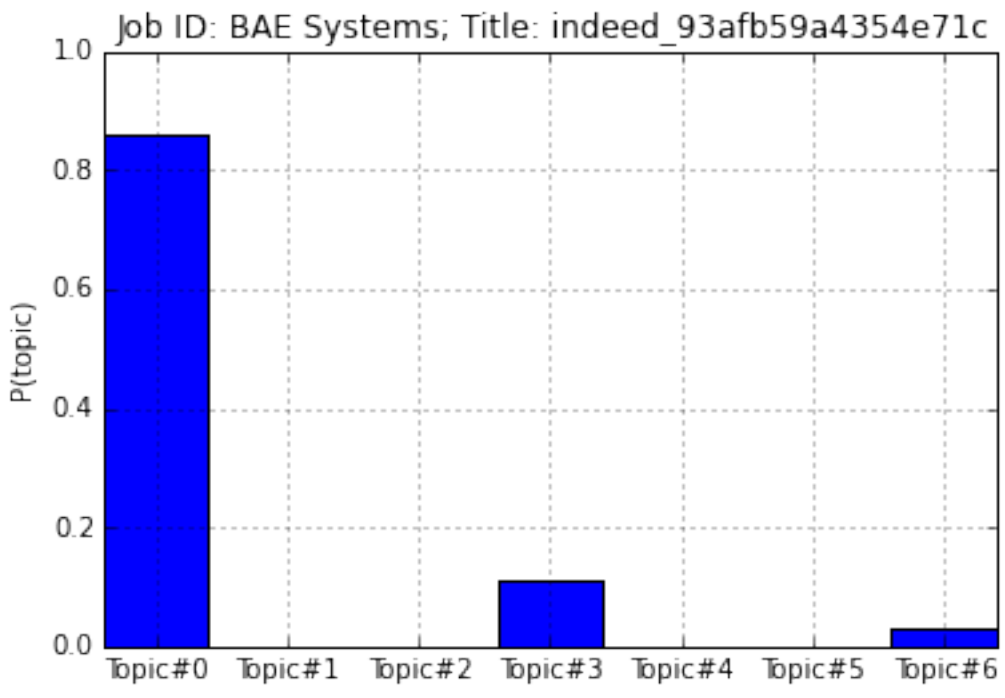
	doc_id	topic_id	topic_prob
0	0	1	0.351
1	0	2	0.086
2	0	3	0.074
3	0	4	0.150
4	0	5	0.337

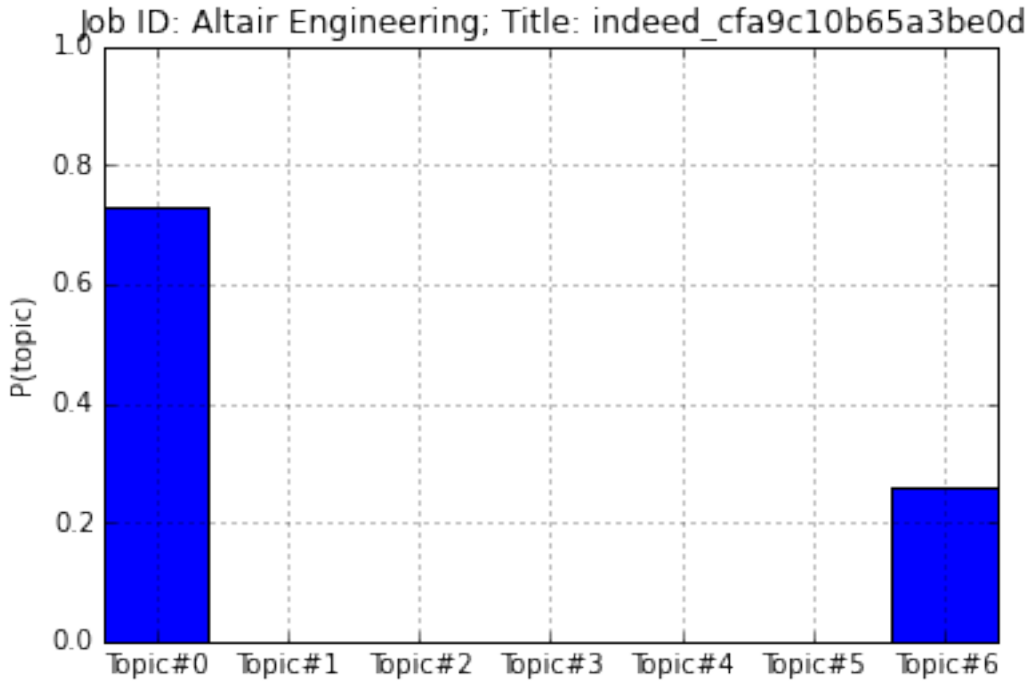
In [59]:

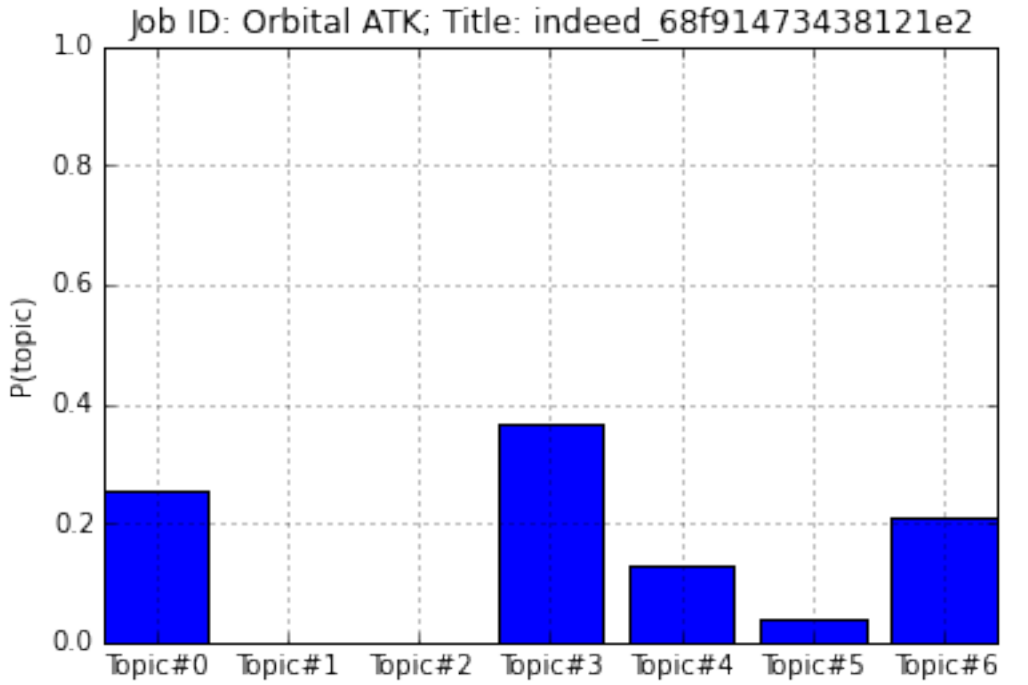
```
search_job_ids = [
    'indeed_93afb59a4354e71c',
    'indeed_cfa9c10b65a3be0d',
    'indeed_68f91473438121e2'
]
```

In [60]:

```
plot_job_distr(topic_df, search_job_ids, train_labels)
```







Topic wise distribution

Particular job can be tagged in multiple topics. We will assign topic # to a job based on top score

In [61]:

```
topic_idx = topic_df.groupby(['doc_id'])['topic_prob'].transform(max) == topic_d
f['topic_prob']
top_topics = topic_df[topic_idx]
top_topics.groupby(['topic_id'])['topic_id'].agg(['count'])
```

Out[61]:

	count
topic_id	
0	2175
1	324
2	1832
3	1646
4	636
5	1331
6	2067

In [102]:

```
topic_docs = [ train_labels[x] for x in top_topics[top_topics['topic_id'] == 5][
'doc_id'] ]
topic_docs_df = pd.DataFrame.from_records(topic_docs,
                                  columns=["Job Id", "Job Title", "Company"])
topic_docs_df
```

Out[102]:

	Job Id	Job Title	Company
0	indeed_53c5e81c18aa4202	Project Coordinator/Data Analyst	The Fund for Public Health in New York, Inc.
1	indeed_3550996c6c6a4cab	ABA Therapist - SE Houston	Unlocking The Spectrum
2	indeed_3cabcc318228ceab	SUBSTITUTE: ABA Therapist/PCA	Early Autism Project, Inc.
3	indeed_edb90543eb022346	Department Business Administrator	University of Houston
4	indeed_74136ac954c1da4a	Sales and Marketing / Clinical Liaison / Hospice	ACUITY PROFESSIONAL PLACEMENT SOLUTIONS
5	indeed_40f3f2a5d50f550f	Administrative Assistant - Office of Campus Life	Curry College

6	indeed_1d30fbfb4053839b	Senior Associate Dean for Finance and Administ...	VA Commonwealth Univ
7	indeed_574a6d74f4fd98ca	Area Manager	Fresenius Medical Care
8	indeed_f9aa330d6be55993	Clinical Manager	UIS Technologies Inc
9	indeed_ef811c95a31a6629	Senior Policy Analyst	ADMIN FOR CHILDREN'S SVCS
10	indeed_411f0827d730c12f	Bilingual Case Manager	Wheeler Clinic
11	indeed_1812c036a757b6ac	Research Specialist IV	Dept of State Health Services
12	indeed_cb969053ccb2c183	Employment Representative	WVU Healthcare
13	indeed_aef068c346dbad75	Medical Assistant - Credentialed (Imaging Center)	University of Minnesota Physicians
14	indeed_47489517028dbcce	Student Services And Instructional Support Coo...	Contra Costa Community College District
15	indeed_e2df18ee99bceafe	Research Specialist V	Health & Human Services Comm
16	indeed_0928ba4698c6469c	EDUCATION SUPERVISOR II-SES	PeopleFirst Florida HR
17	indeed_bfa799633a83bd0a	Licensed Clinical Counselor	Children's Home Society of Florida
18	indeed_fa92482d472df0f9	Jr. Project Manager	DOCS
19	indeed_85a2216291ff440b	Research Associate - Molecular Biology- Divison S	UT Southwestern Medical Center
20	indeed_a6865d6177a3c6a3	Leasing Manager	Common Ground
21	indeed_2ad94b97e6f64cbf	Trauma Registrar	Henrico Doctors Hospital Forest
22	indeed_fa190bacb5bf20e5	Coordinator, Information Systems	The University of Maine, Hutchinson Center
23	indeed_e5dce83d128807d4	4th Grade Teacher - Temporary for the remainde...	North East Florida Educational Consortium
24	indeed_038b5d0616b20106	Fundraising and Development Coordinator	The Child and Family Network Centers

25	indeed_c5ed60d9fa2687f9	Youth Care Worker	New Jersey MENTOR
26	indeed_98e580e2962bf395	Advanced Practice Professional - Emergency Med...	West Virginia University Health Associates
27	indeed_4918dc2a72e94f07	Medical Records Assistant	Covenant Dove
28	indeed_38fa6acd55edddad	Admissions Assistant II (ADM-Operations)	University of Florida
29	indeed_0bb5a63aee231dd5	Senior Award Manager	Save the Children
...	...	...	...
1301	indeed_b0bbb5264517f48c	Career Services Specialist	University of North Georgia
1302	indeed_c9fe379c7c845b00	Certified Surgical Tech	Warren Memorial Hospital
1303	indeed_a9c0fcf0b7b4a8d5	Elementary School Science Teacher	Success Academy Charter Schools
1304	indeed_b05d87697dac8ef4	Trauma Registrar	UPMC
1305	indeed_92081f01267ee055	Administrator, Office of Student Services, Car...	Yeshiva University
1306	indeed_d9ab2c43af34ccc2	Deputy Agency Chief Contracting Officer, Procu...	HRA/DEPT OF SOCIAL SERVICES
1307	indeed_57241a1e1ab2e22b	Service Coordinators - Bilingual Spanish Speaking	Easter Seals Metropolitan Chicago
1308	indeed_0280359e4d88dd98	Part-time Certified Fitness Assistant	TCA Health, Inc., N.F.P.
1309	indeed_539c553143586d00	Violence and Injury Prevention Program Plannin...	King County
1310	indeed_cd0b75e0fe01abe5	DAE 2016 S-Term 9-12 Gordon Parks Licensed Tea...	Saint Paul Public Schools
1311	indeed_17c63a60f8cd55bd	Licensed Specialist in School Psychology	Diagnostic Assessment Services, Inc.
1312	indeed_b7583bc1973ac758	Senior Grants Accountant	Dept for Aging & Rehabilitative Service
1313	indeed_fe72945f8d5d8380	Degree Progress Specialist (Administrative Ana...	San Francisco State University

		Ana...	
1314	indeed_fd4cbe5ddb5f00b8	Behaviorist - (Ohana IDD Program)	Legacy Treatment Services
1315	indeed_f8c966ce72a5c307	Medical Records Specialist	Easter Seals UCP North Carolina
1316	indeed_66a27b9e95ea3a6d	Institutional Research Analyst	The University of Texas System
1317	indeed_990f61b8f3a511ec	Arts Communication Manager for Visual and Perf...	Millersville University of Pennsylvania
1318	indeed_4c94e31fea6a294b	Patient Safety/Quality Specialist Supervisor	BJC HealthCare
1319	indeed_a9aa0e7549abf979	Senior Associate/Associate Director Processing...	Virginia Tech
1320	indeed_25733e49cd873efe	Respiratory Therapy Clinical Team Lead	LewisGale Hospital-Pulaski
1321	indeed_d8bd5fa469e5e602	Hourly Outreach Specialist	Seattle Colleges
1322	indeed_0af9116e5e5187bd	Part-time Senior Administrative Assistant (3118)	American University
1323	indeed_d994a0e035798314	Grant Writer	Woodland Park Zoo
1324	indeed_bc7391a38256b5fb	Behavior Line Therapist for Autism - part time	Autism Home Support
1325	indeed_f094c2e94e7d2f05	Associate I, Maternal Newborn Health	Population Council
1326	indeed_205af7586a210394	CASE MANAGER	Lakeview Center Inc.
1327	indeed_ab8ca717e18ac87d	Medical Writer	BMS
1328	indeed_62e37d85d375f946	Data Coordinator	The University of Pittsburgh
1329	indeed_6b4e421f33945571	CLINICAL COORDINATOR	Carilion Clinic
1330	indeed_f4a30ee7fc6deaf0	Dir, Financial Aid	Appalachian State University

1331 rows × 3 columns

8. Testing with Random Job Post

In [94]:

```
!tail -1 ~/wrk/jobs/data/export_jobs_w_title.txt | awk -F'|' '{print $5}' > ~/wr
k/jobs/data/test.txt
!tail -1 ~/wrk/jobs/data/export_jobs_w_title.txt | awk -F'|' '{print $1"|"$2"|"$
3"|"$4}' > ~/wrk/jobs/data/test_labels.txt
!cat ~/wrk/jobs/data/test.txt
!cat ~/wrk/jobs/data/test_labels.txt
```

```
McCoyâs Building Supply is looking for a strong candidate for a new
Pricing Analyst position based at our Headquarters facility in San M
arcos, Texas. This is an exempt-level position, and the final salary
for this position is to be determined.     Our ideal  Pricing Analys
t candidate will be responsible for driving price optimization and e
xecuting pricing strategies at McCoyâs.  This includes gathering com
petitor pricing, developing pricing scenarios that fit each category
âs overall strategy, and supporting your recommendations to McCoyâs
Merchants, with maximizing profitable market share growth for the bu
siness as the main goal. You need to be collaborative and persuasive
, have a technical eye, and be able to communicate with non-technica
l teammates. Fact based, data driven decision-making is a key part o
f what youâll do to deliver the best pricing plans to our Merchandis
ing and Operations Teams, and ultimately to our Born to Build Custom
ers.      SOME OF THE DUTIES AND RESPONSIBILITIES OF THIS POSITION I
NCLUDE THE FOLLOWING  :     Â·  Price Optimization  : Incorporating
competitive intelligence, develop pricing scenarios, and make recomm
endations to Merchants in support of category strategies. Provide fi
nancial analysis and analytical support to the Merchant community to
assist group in making better pricing decisions.     Â·  Execute Cat
egory Pricing Strategies  : Present options, facilitate decisions, a
nd implement pricing strategies, build and manage business rules and
strategic pricing plan for all categories, and work collaboratively
across the Merchandising organization     Â·  Deliver Competitive In
telligence  : Collect and Monitor competitorsâ prices and analyze re
sults to drive changes to individual prices, and potential changes t
o pricing strategies. Execute âwhat ifâ scenarios. Analyze and track
progress on strategic pricing decisions and strategic pricing plans
Â·  General Responsibilities  : Manage the pricing calendar to balan
ce workload in the stores. Coordinate the day-to-day pricing activit
ies within each merchandise category. Proactively communicate releva
nt information as necessary to appropriate levels in the organizatio
n, formally and informally, in both written and oral forms     Requi
rements    SOME OF THE QUALIFICATIONS OF THIS POSITION INCLUDE :
Â· Bachelor's degree from four-year college or university; or one to
two years of applicable merchandising analysis experience; or equiva
lent combination of education and experience     Â· Ability to utili
ze Microsoft Office (Word, Excel, Access and PowerPoint) and other s
oftware programs at an intermediate level     Â· Must be regularly a
vailable and willing to work at least 8 hours per day, 40 hours per
week or such other hours per day or hours per week as the employer d
```

etermines are necessary or desirable to meet business needs Â· T
his position requires occasional travel with overnight stays, so you
must be able to meet the driverâs license and insurance requirements
of the Company PREFERRED QUALIFICATIONS Â· Retail experienc
e is strongly preferred Â· Experience with data warehousing and
statistical analysis software packages (e.g., Cognos, SAS, SPSS, Sta
ta) Â· Specific experience and proficiency with retail pricing s
oftware packages Â· Experience with BI/Data Warehousing Tool (Co
gnos, BI10+ or related tools) Â· Certified Pricing Professional
(CPP) certification NOTE: A full job description will be provid
ed to initially qualified candidates during the interview process.
indeed_f7b2b78d308b2e7b|Pricing Analyst|McCoy's Building Supply|http
://www.indeed.com/viewjob?jk=f7b2b78d308b2e7b&qd=PuuFZTQAvQAUoZwXvww
yddUYJIifLepZz3H4vGYPJ2-_LiCPa505cRTtNIIqqYAPjqV6NiOfT96MeYswXFwOESu
Hnh4d5TNqhbGUJLosmuM&indpubnum=3869750015307590&atk=1aeas3r7bb9fkfmm

In [99]:

```
!grep indeed_50bf5026f812b820 ~/wrk/jobs/data/export_jobs_w_title.txt | awk -F'|
' '{print $5}' > ~/wrk/jobs/data/test.txt
!grep indeed_50bf5026f812b820 ~/wrk/jobs/data/export_jobs_w_title.txt | awk -F'|
' '{print $1"|"$2"|"$3"|"$4}' > ~/wrk/jobs/data/test_labels.txt
!cat ~/wrk/jobs/data/test.txt
!cat ~/wrk/jobs/data/test_labels.txt
```

Teachers hold primary responsibility for the implementation and deve
lopment of Uncommonâs curriculum and the success of its students. Th
erefore, Uncommon Schools seeks teachers who are committed to contin
uously improving curriculum and instruction through collaboration as
part of a grade level team. Implement curricula and activities to me
et academic standards;Design and implement assessments that measure
progress towards academic standards;Use assessment data to refine cu
rriculum and inform ins o-o-o
indeed_50bf5026f812b820|High School Algebra 1 Teacher (2016-2017 Sch
ool Year)|Preparatory Charter Schools|http://www.indeed.com/viewjob?
jk=50bf5026f812b820&qd=PuuFZTQAvQAUoZwXvwwydVJX_fBthdM8Fvcy9hVLgMsm1
Jstv5h9RbSRH07keVMyhGW0PtQg12oEkmVRFhi1RJifobd018Nm_bbbb0NA9MI&indpu
bnum=3869750015307590&atk=1abet3edfbqnj8lk

In [100]:

```
# compile sample documents into a list
test_set = [ preprocess(line.decode('unicode_escape').encode('ascii', 'ignore'))
for line in open('/home/rt/wrk/jobs/data/test.txt', 'r') ]

# list for tokenized documents in loop
test_tokenized = tokenize_only(test_set[0])
test_dict = corpora.Dictionary([test_tokenized])
test_bow = dictionary.doc2bow(test_tokenized)
```

Let's see what topis test document belongs to

In [101]:

```
for topics in lda[test_bow]:
    print topics
```

```
(5, 0.98190453974814329)
```

So the test document belongs to topics 0, 2, 3, 7 and 9

In [58]:

```
print test_set
```

```
[u"Now Hiring Company Truck Drivers. At Transport America We Raised
Pay!  Company Truck Driver Benefits: Top 10% Industry Pay Year Round
Steady Freight Performance Pay   Experienced Drivers Earn Top Scale
in 2 Years Flexible Home Time, Including Get Home Certificates 24 7
Support, 365 Days A Year Pick Your Schedule Option Lease Purchase Op
tions Day 1 Medical Dental Vision Disability Benefits Package Transf
er Opportunities Available E Logs and an InCab Communication Hub Rol
l Stability and OnGuard System CSA Safe Carrier New Fleet of Equipme
nt  New Kenworths In Delivery  At Transport America, our goal is to
deliver excellence in all that we do. At a time when others are movi
ng to asset lite models, we are committed to running assets in netwo
rks, which gives you reliable capacity with an excellence of service
unsurpassed in the transportation industry. We are big enough to cre
ate meaningful solutions, but small enough to provide you the level
of customer service you deserve. We believe in hiring the best truck
drivers in the industry and empower them to create solutions for our
customers. Because of our asset intensity, we attract and retain the
best drivers in the trucking industry. The technology we employ is f
ocused on enhancing your service experience. Our experienced driver
base, with retention levels well above the industry average, sets us
apart from our competitors. Transport America's fleet of company tru
ck drivers is the best and most experienced on the road. We welcome
you to fill out the form above to be contacted by one of our recruit
ers! Call us for details at 877 957 3117\n"]
```

Appendix

1. Tokenizing and Stemming

In [209]:

```
vocab_stemmed = []
vocab_tokenized = []

for jobdesc in train:
    stemmed = tokenize_and_stem(jobdesc)
    vocab_stemmed.extend(stemmed)

    tokenized = tokenize_only(jobdesc)
    vocab_tokenized.extend(tokenized)
```

```
337 337
337
```

In [210]:

```
print "{}, {}".format(len(vocab_stemmed), len(vocab_tokenized))
```

```
337, 337
```

In [211]:

```
df_vocab = pd.DataFrame({'words': vocab_tokenized}, index = vocab_stemmed)
df_vocab = df_vocab.drop_duplicates()
print 'there are ' + str(df_vocab.shape[0]) + ' items in vocab_frame'
```

```
there are 235 items in vocab_frame
```

In [212]:

```
print df_vocab.head(20)
```

```
              words
potbelli    potbelly
associ    associates
job              job
make            make
custom     customers
realli        really
happi          happy
sinc           since
primari      primary
point          point
custom      customer
contact      contact
provid       provide
excel      excellent
experi    experience
provid     providing
fast            fast
friend      friendly
effici     efficient
servic       service
```