TopicAnalysisMinusPersonality

March 17, 2016

```
In [3]: from textstat.textstat import textstat
        import csv
        import pandas
        import matplotlib
        #matplotlib.style.use('ggplot')
        %matplotlib inline
        import ast
```

```
In [4]: trump_df = pandas.read_csv('data/all_trump_w_topics.csv')
        trump_df['candidate'] = 'trump'
        clinton_df = pandas.read_csv('data/all_clinton_w_topics.csv')
        clinton_df['candidate'] = 'clinton'
        sanders_df = pandas.read_csv('data/all_sanders_w_topics.csv')
        sanders_df['candidate'] = 'sanders'
        cruz_df = pandas.read_csv('data/all_cruz_w_topics.csv')
        cruz_df['candidate'] = 'cruz'
        ORGS = ['nyt', 'wsj', 'cnn', 'fox', 'ap', 'reuters', 'politico', 'mcclatchy', 'buzzfeed', 'huff
```

```
In [5]: n = len(clinton_df)
        clinton_df.index = xrange(len(trump_df), (len(trump_df) + n))
        m = len(sanders_df)
        sanders_df.index = xrange(max(clinton_df.index), max(clinton_df.index) + m)
        c = len(cruz_df)
        cruz_df.index = xrange(max(sanders_df.index), max(sanders_df.index) + c)
```

```
In [6]: all_df = pandas.concat([trump_df,clinton_df, sanders_df, cruz_df])
        all_df['gunning_fog'] = all_df['body'].apply(lambda x: textstat.gunning_fog(x) if type(x) == st
        all_df['flesch'] = all_df['body'].apply(lambda x: textstat.flesch_kincaid_grade(x) if type(x) =
        all_df['readability'] = all_df['body'].apply(lambda x: textstat.flesch_reading_ease(x) if type(
```

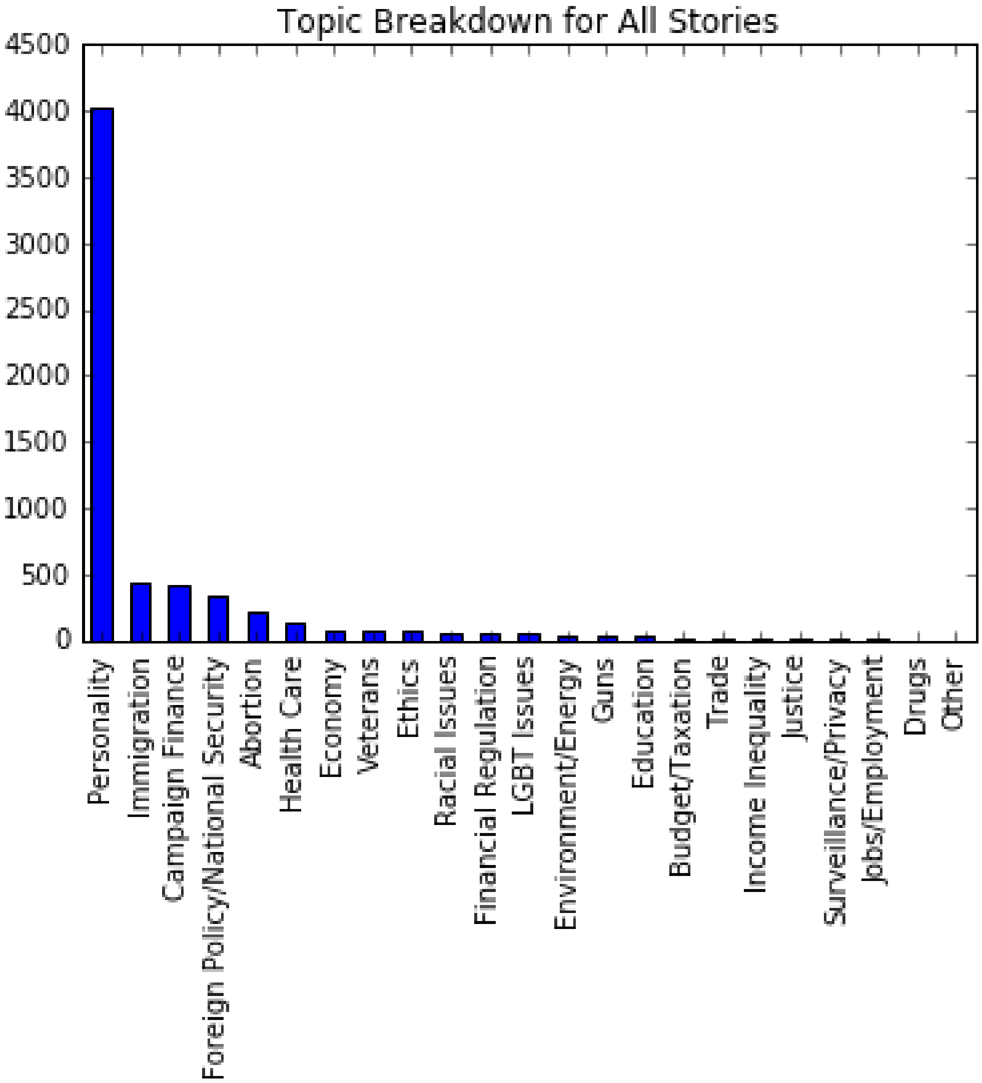
1 Convert topics to Dict and Filter by > 0.1

```
In [9]: all_df['topic_dict'] = all_df['topic'].apply(lambda d: ast.literal_eval(d))
        all_df['top_topics'] = all_df['topic_dict'].apply(lambda d: {k:v for k, v in d.iteritems() if v
        all_df['topic_list'] = all_df['top_topics'].apply(lambda d: d.keys())
        all_df['top_topic'] = all_df['topic_dict'].apply(lambda d: max(d, key=lambda i: d[i]))
```

2 Breakdown of Story Topics

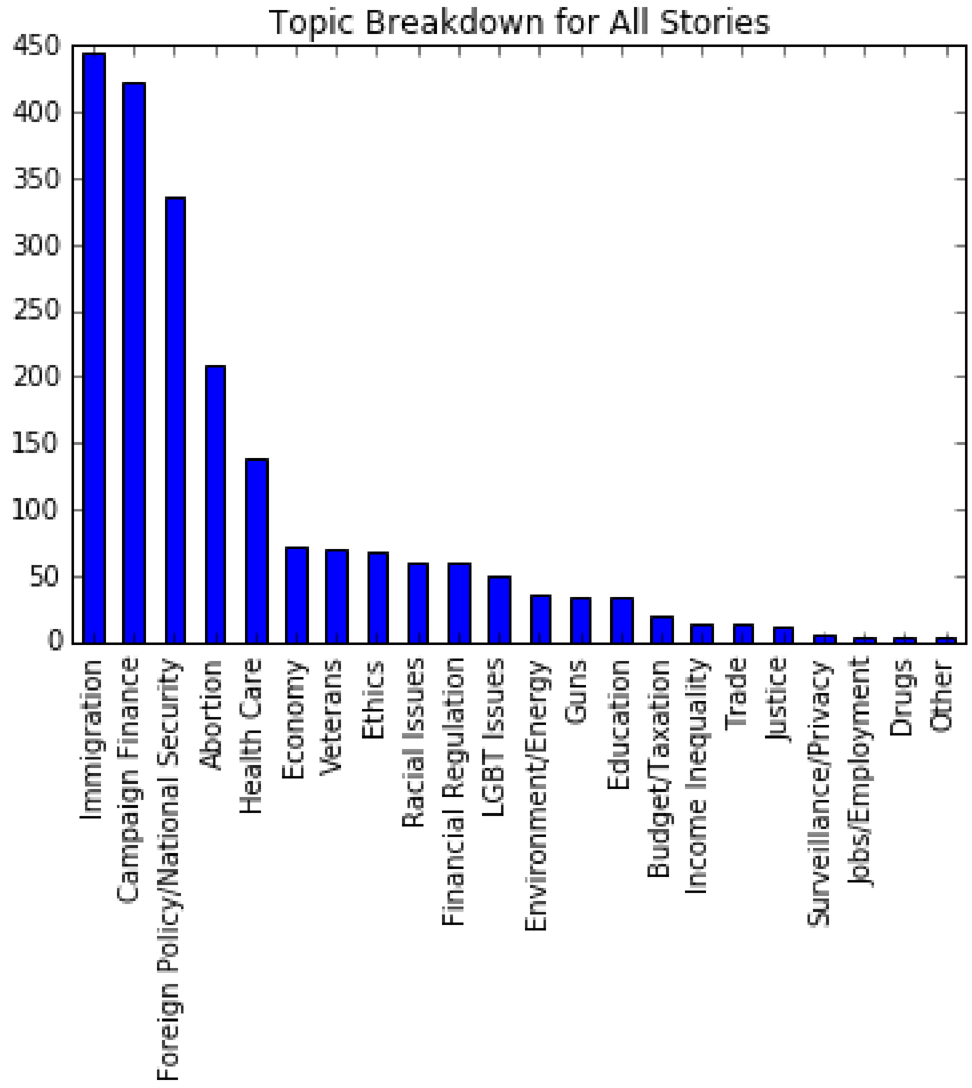
```
In [10]: all_df['top_topic'].value_counts().plot(kind="bar", title="Topic Breakdown for All Stories")
         # Top 10: Personality, Immigration, Campaign Finance, Foreign Policy/National Security, Aborti
         # Health Care, Economy, Veterans, Ethics, Racial Issues
```

```
Out[10]: <matplotlib.axes._subplots.AxesSubplot at 0x1072ef6d0>
```

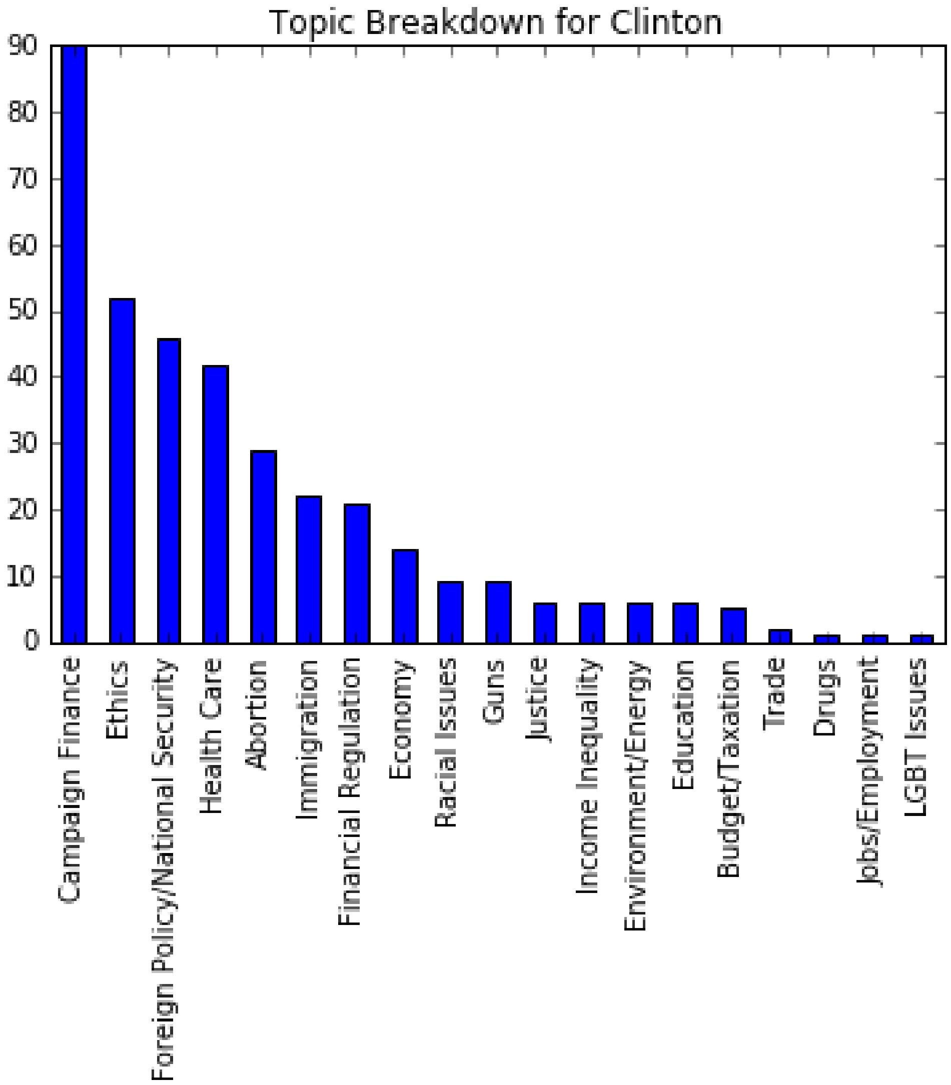


2.1 Now Remove Personality Topic (Which is Other)

```
In [11]: all_df = all_df[all_df[’top_topic’] != ’Personality’]
         all_df[’top_topic’].value_counts().plot(kind="bar", title="Topic Breakdown for All Stories")
         # Top 10: Personality, Immigration, Campaign Finance, Foreign Policy/National Security, Aborti
         # Health Care, Economy, Veterans, Ethics, Racial Issues

Out[11]: <matplotlib.axes._subplots.AxesSubplot at 0x106e938d0>
```

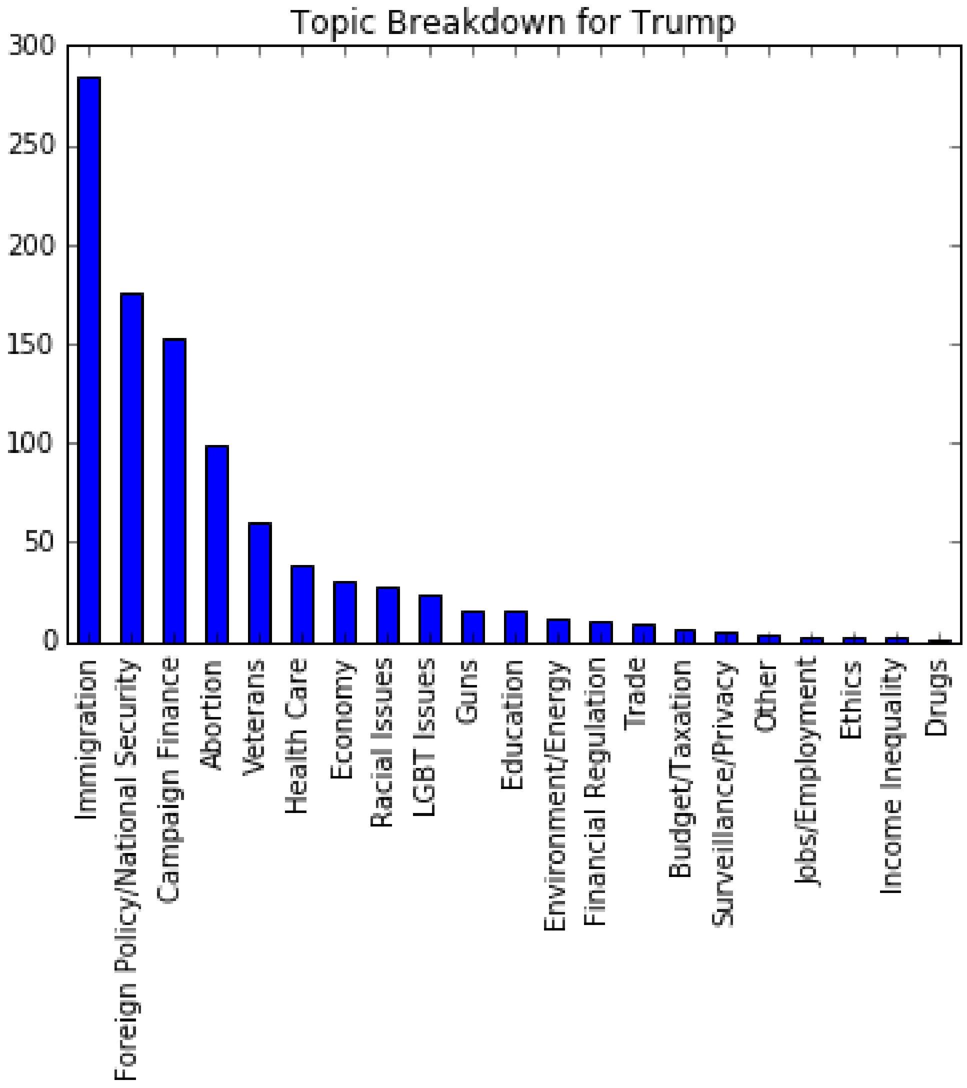


```
In [12]: all_df[all_df['candidate'] == 'clinton']['top_topic'].value_counts().plot(kind="bar", title="To

Out[12]: <matplotlib.axes._subplots.AxesSubplot at 0x107e9a490>
```

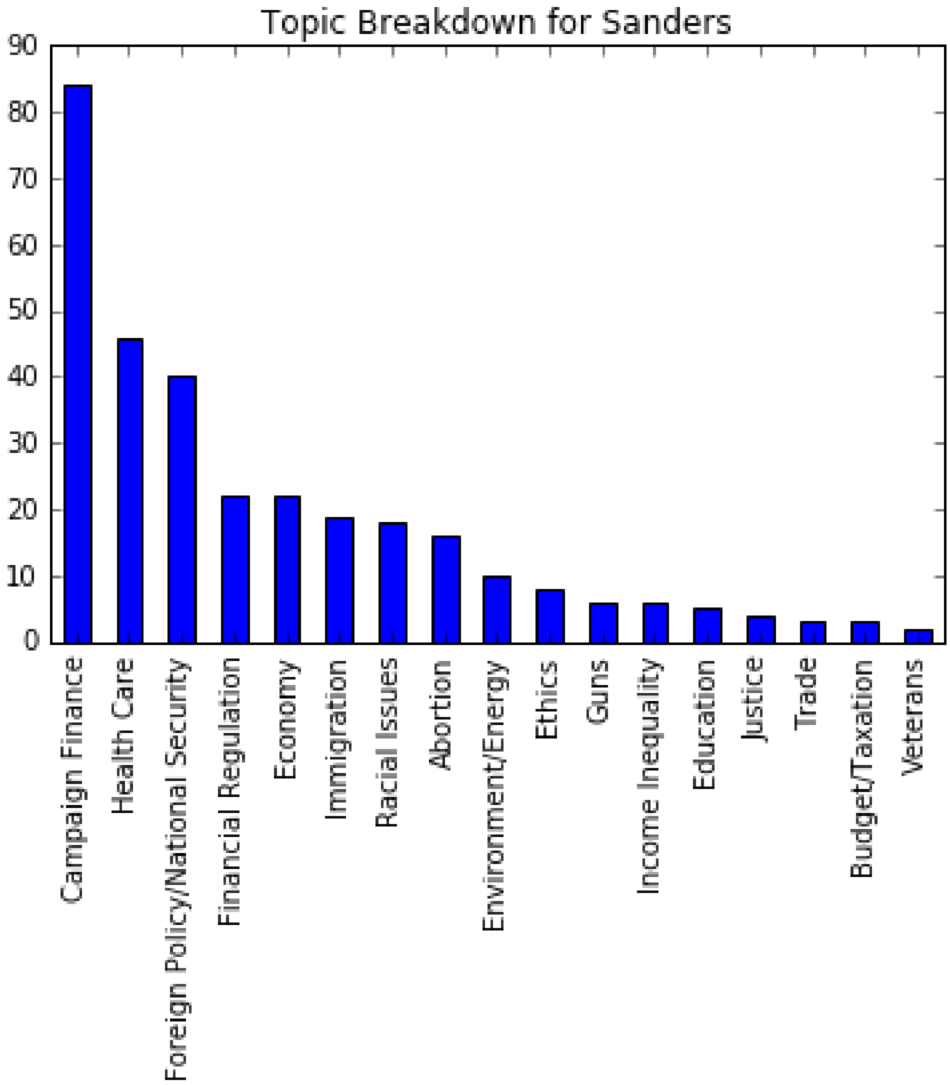


```
In [13]: all_df[all_df['candidate'] == 'trump']['top_topic'].value_counts().plot(kind="bar", title="Top
```

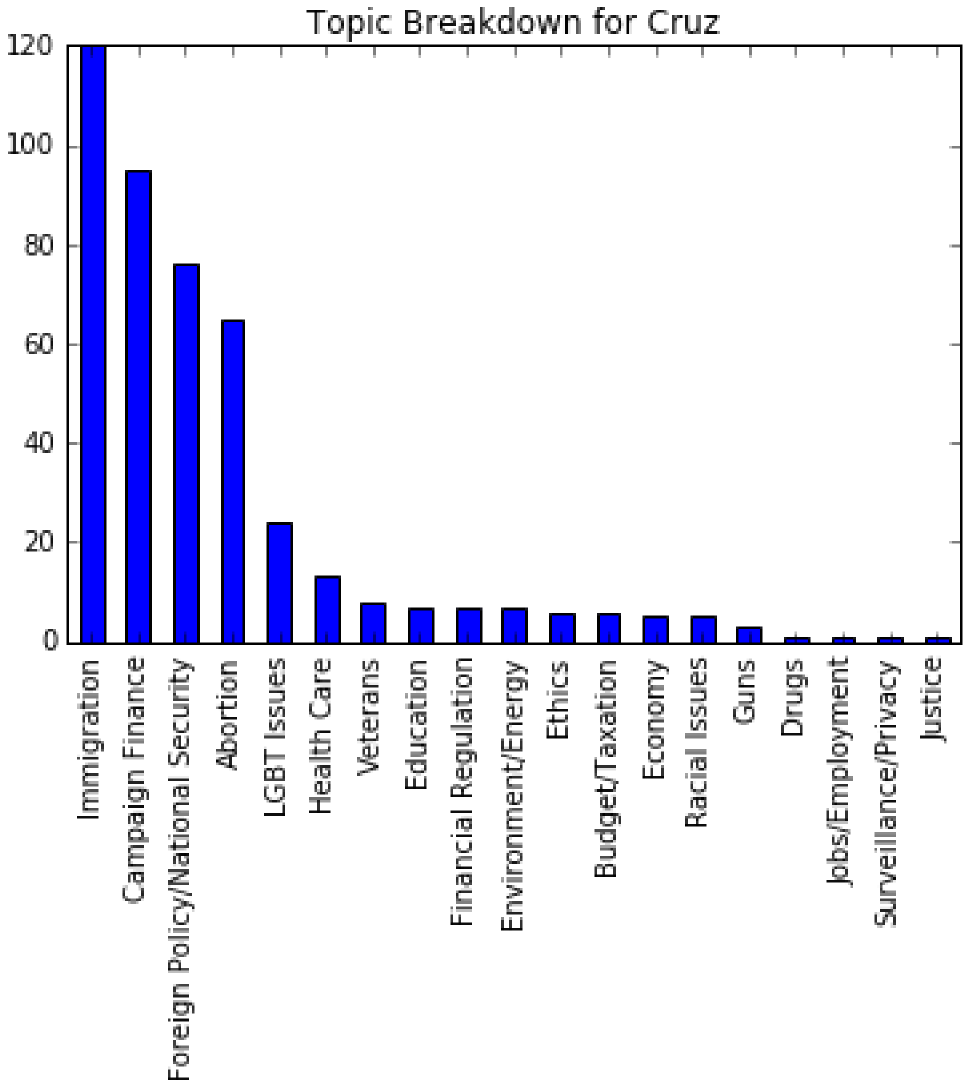
```
Out[13]: <matplotlib.axes._subplots.AxesSubplot at 0x107efdf10>
```



```
In [14]: all_df[all_df['candidate'] == 'sanders']['top_topic'].value_counts().plot(kind="bar", title="To

Out[14]: <matplotlib.axes._subplots.AxesSubplot at 0x107cae590>
```



```
In [15]: all_df[all_df['candidate'] == 'cruz']['top_topic'].value_counts().plot(kind="bar", title="Topi

Out[15]: <matplotlib.axes._subplots.AxesSubplot at 0x106dc4fd0>
```



2.2 Reading Level Breakdown by Topic

```
In [16]: # Top 10: Personality, Immigration, Campaign Finance, Foreign Policy/National Security, Aborti
         # Health Care, Economy, Veterans, Ethics, Racial Issues

         TOPICS = [’Immigration’, ’Campaign Finance’, ’Foreign Policy/National Security’,
                   ’Abortion’, ’Health Care’, ’Economy’, ’Veterans’, ’Ethics’, ’Racial Issues’, ’Financi

         print "Average Flesch Scores by Topic"
         for t in TOPICS:
             df_t = all_df[all_df[’top_topic’] == t]
             print t, ":", ’%.2f’ % df_t[’flesch’].mean()
```

```
Average Flesch Scores by Topic
Immigration : 9.28
Campaign Finance : 9.01
```

```
Foreign Policy/National Security : 9.19
Abortion : 9.08
Health Care : 9.86
Economy : 9.75
Veterans : 9.06
Ethics : 10.90
Racial Issues : 9.93
Financial Regulation : 9.74
```

2.3 Topic Breakdown By Candidate

```
In [26]: CANDIDATES = ['clinton', 'sanders', 'trump', 'cruz']
         for c in CANDIDATES:
             print c
             print "\t\t\t\t\t%"
             print 100* all_df[all_df['candidate'] == c]['top_topic'].value_counts(normalize=True)[:5]
             print
```

```
clinton
                                        %
Campaign Finance                    24.456522
Ethics                              14.130435
Foreign Policy/National Security    12.500000
Health Care                         11.413043
Abortion                             7.880435
Name: top_topic, dtype: float64

sanders
                                        %
Campaign Finance                    26.751592
Health Care                         14.649682
Foreign Policy/National Security    12.738854
Financial Regulation                 7.006369
Economy                              7.006369
Name: top_topic, dtype: float64

trump
                                        %
Immigration                         29.128205
Foreign Policy/National Security    17.948718
Campaign Finance                    15.692308
Abortion                            10.153846
Veterans                             6.153846
Name: top_topic, dtype: float64

cruz
                                        %
Immigration                         26.607539
Campaign Finance                    21.064302
Foreign Policy/National Security    16.851441
Abortion                            14.412417
LGBT Issues                          5.321508
Name: top_topic, dtype: float64
```

2.4 Average Reading Scores by Candidate per Topic

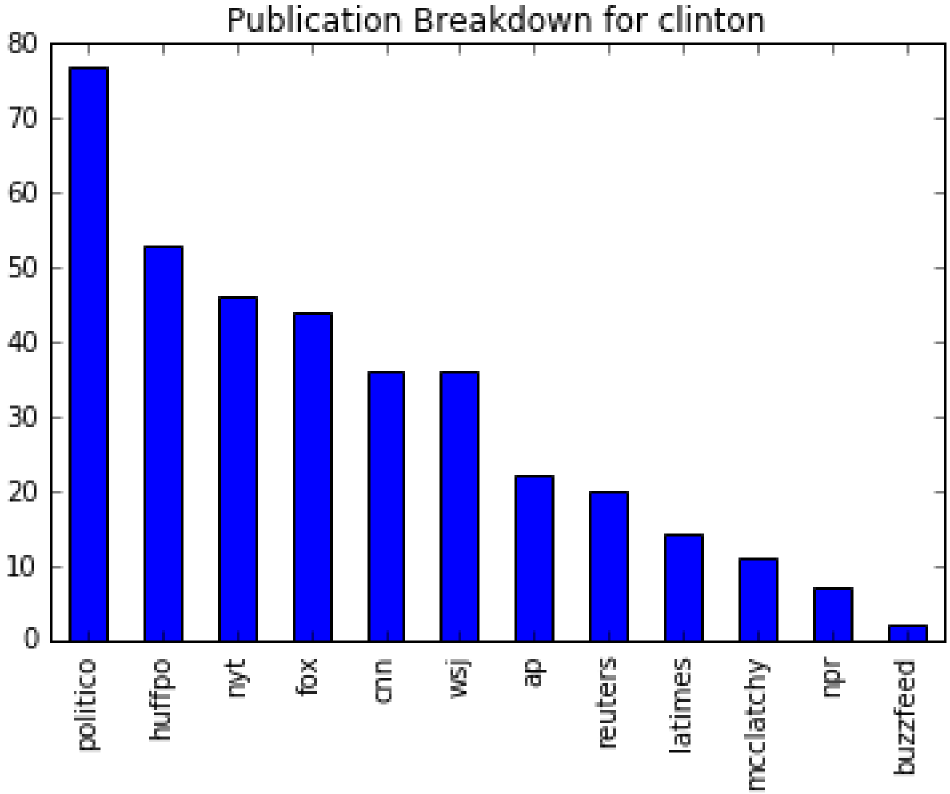
```
In [45]: CANDIDATES = ['clinton', 'sanders', 'trump', 'cruz']
         for c in CANDIDATES:
             print c, 'average Flesch score', '%.2f' % all_df[all_df['candidate'] == c]['flesch'].mean(
         print
         for t in TOPICS:
             scores = []
             for c in CANDIDATES:
                  scores.append((c,all_df[(all_df['candidate'] == c) & (all_df['top_topic'] == t)]['fle
                                len(all_df[(all_df['candidate'] == c ) & (all_df['top_topic'] == t)])))
             scores.sort(key=lambda x: x[1], reverse=True)

             print t
             for s in scores:
                 print s[0], '%.2f' % s[1], "avg", "(", s[2], "stories )"
             print
```

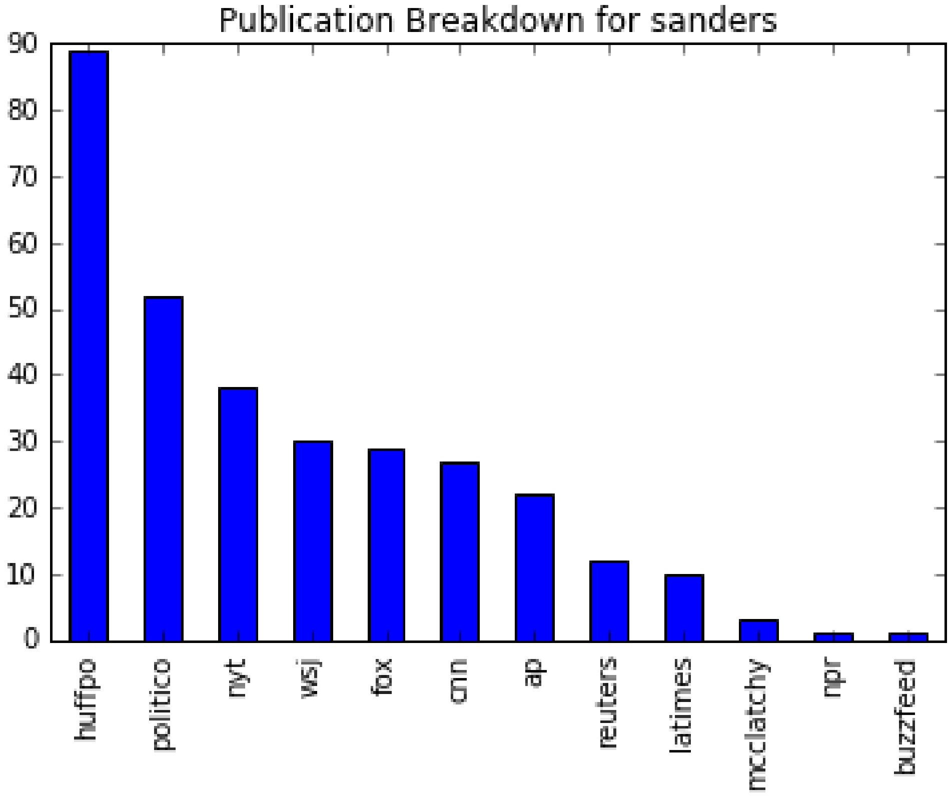
```
clinton average Flesch score 10.06
sanders average Flesch score 9.96
trump average Flesch score 9.07
cruz average Flesch score 8.94

Immigration
sanders 10.19 avg ( 19 stories )
clinton 10.09 avg ( 22 stories )
trump 9.31 avg ( 284 stories )
cruz 8.91 avg ( 120 stories )

Campaign Finance
sanders 9.40 avg ( 84 stories )
clinton 9.34 avg ( 90 stories )
trump 8.79 avg ( 153 stories )
cruz 8.73 avg ( 95 stories )

Foreign Policy/National Security
clinton 10.16 avg ( 46 stories )
sanders 9.77 avg ( 40 stories )
trump 8.93 avg ( 175 stories )
cruz 8.89 avg ( 76 stories )

Abortion
clinton 10.19 avg ( 29 stories )
sanders 9.45 avg ( 16 stories )
cruz 8.90 avg ( 65 stories )
trump 8.81 avg ( 99 stories )

Health Care
sanders 10.52 avg ( 46 stories )
clinton 9.85 avg ( 42 stories )
cruz 9.55 avg ( 13 stories )
trump 9.16 avg ( 38 stories )

Economy
clinton 10.42 avg ( 14 stories )
```

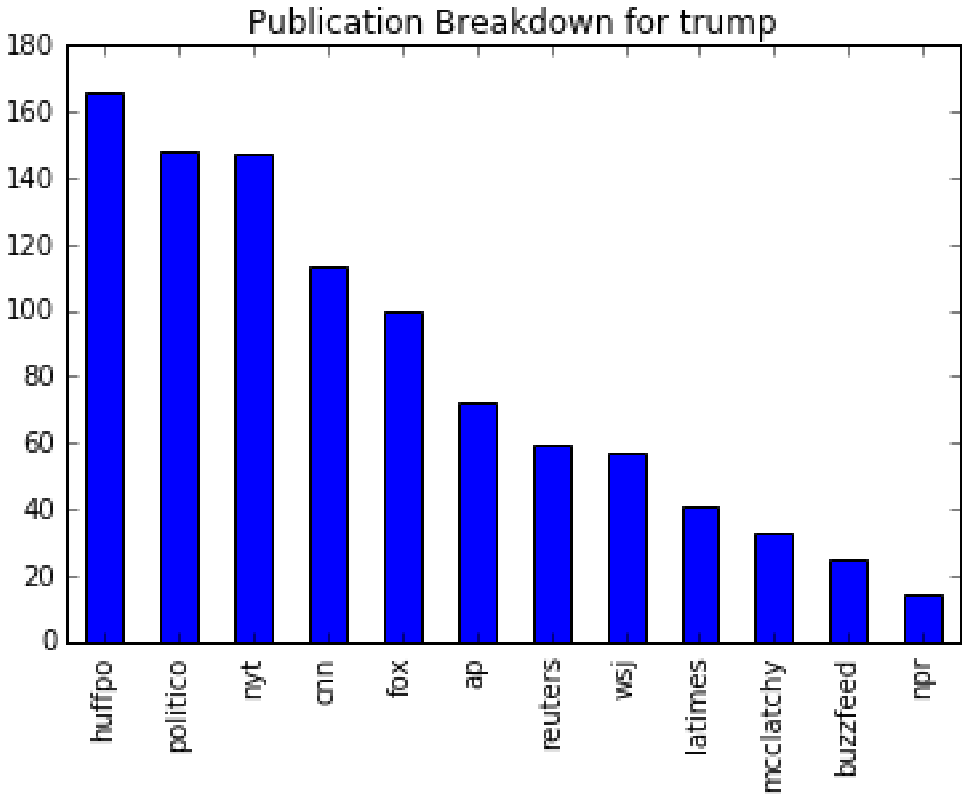
```
sanders 9.98 avg ( 22 stories )
trump 9.48 avg ( 30 stories )
cruz 8.50 avg ( 5 stories )

Veterans
clinton nan avg ( 0 stories )
trump 9.08 avg ( 60 stories )
cruz 9.07 avg ( 8 stories )
sanders 8.30 avg ( 2 stories )

Ethics
sanders 11.11 avg ( 8 stories )
trump 11.10 avg ( 2 stories )
cruz 10.88 avg ( 6 stories )
clinton 10.86 avg ( 52 stories )

Racial Issues
sanders 10.99 avg ( 18 stories )
clinton 10.76 avg ( 9 stories )
trump 9.20 avg ( 28 stories )
cruz 8.66 avg ( 5 stories )

Financial Regulation
sanders 10.26 avg ( 22 stories )
clinton 9.90 avg ( 21 stories )
cruz 9.23 avg ( 7 stories )
trump 8.61 avg ( 10 stories )
```

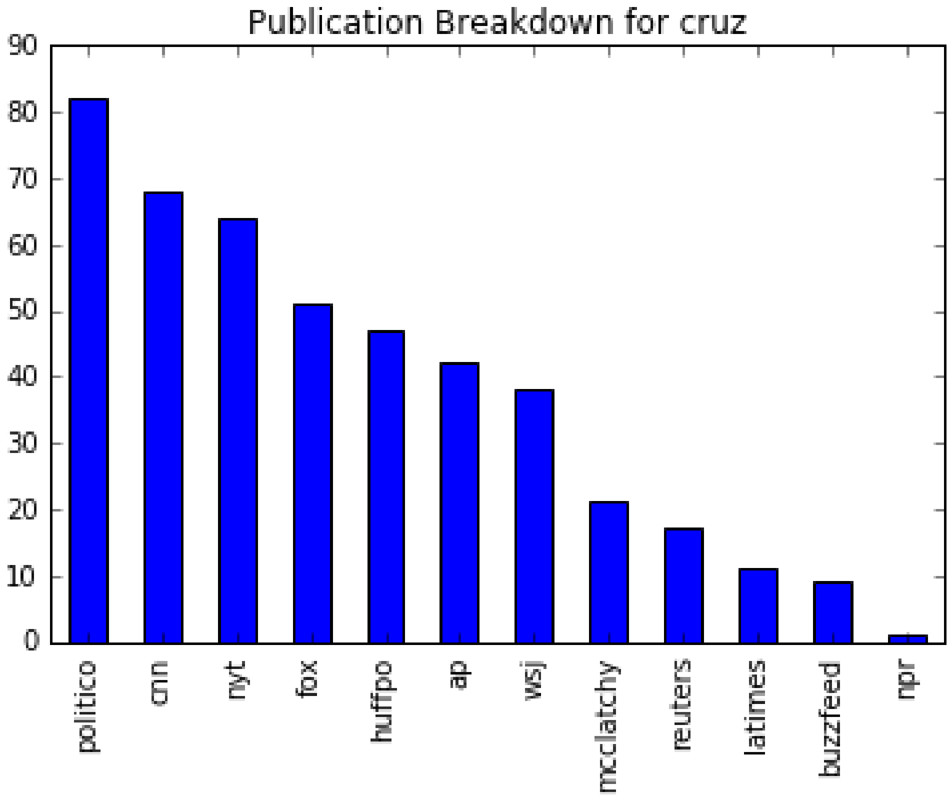
2.5 Story Distrubtion Per Candidate

```
In [46]: CANDIDATES = ['clinton', 'sanders', 'trump', 'cruz']

         for c in CANDIDATES:
             all_df[all_df['candidate'] == c]['org'].value_counts().plot(kind="bar", title="Publication
             matplotlib.pyplot.show()
```


Publication Breakdown for clinton
80
70
60
50
40
30
20
10
0
politico
huffpo
nyt
fox
cnn
wsj
ap
reuters
latimes
mcclatchy
npr
buzzfeed


Publication Breakdown for sanders
90
80
70
60
50
40
30
20
10
0
huffpo
politico
nyt
wsj
fox
cnn
ap
reuters
latimes
mcclatchy
npr
buzzfeed


Publication Breakdown for trump
180
160
140
120
100
80
60
40
20
0
huffpo
politico
nyt
cnn
fox
ap
reuters
wsj
latimes
mcclatchy
buzzfeed
npr



```
In [47]: #all_df[(all_df['candidate']=='trump') & (all_df['top_topic'] == 'Veterans')]

In [ ]:
```