Analysing Language used in Some Online News Headlines with Python

Jasper Kearney

https://github.com/jasperjkearney/headline-language-analysis

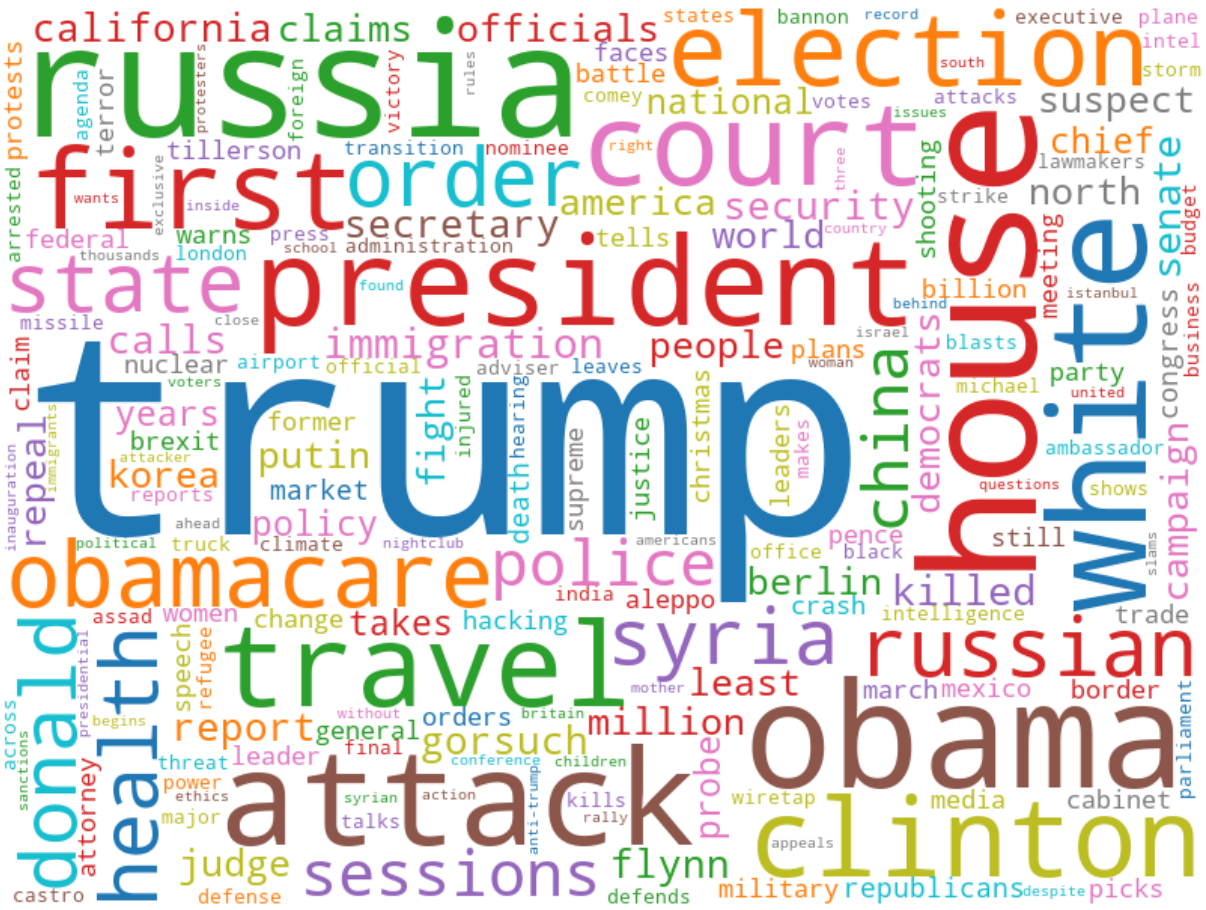

Figure 1: Word cloud generated from words > 4 letters in headlines collected 06/11/2016 - 14/02/2017. Word sizes are proportional to their frequency of occurrence. Created with `https://github.com/amueller/word_cloud`.

1 Introduction

This was a small project to done to improve knowledge of Python. Over the course of several months, headlines were collected hourly from several major news sites, and some shallow analysis and charts produced for this report.

2 Method

Headlines, along with their associated metadata, were retrieved and stored in a comma-separated values (CSV) file. The file was then used to create a Python class, meaning the data could be evaluated in a variety of ways. The headlines collected and analysed were the singular largest and most prominent headlines on the front page of 14 of the most-read[1] news websites (Table 1).

Table 1: Headline sources.

News Organisation	**Website**
BBC	`http://bbc.com/news`
Daily Mail	`http://www.dailymail.co.uk/news/index.html`
NBC News	`http://www.nbcnews.com/`
The Washington Post	`https://www.washingtonpost.com/`
New York Times	`http://www.nytimes.com/`
The Huffington Post	`http://www.huffingtonpost.com/`
Fox News	`http://www.foxnews.com/`
The Guardian	`https://www.theguardian.com/international`
The Wall Street Journal	`http://www.wsj.com/`
USA Today	`http://www.usatoday.com/`
LA Times	`http://www.latimes.com/`
India Times	`http://www.indiatimes.com/`
Bloomberg	`http://www.bloomberg.com/`
Yahoo	`https://www.yahoo.com/news/`

2.1 Collecting Headlines

The Python script `store_current_headlines.py` retrieves headlines from the news sources in Table 1, and writes them - along with metadata - to a CSV file (Figure 2).
It utilises the `urllib.request`[1] module from the Python standard library to fetch HTML. The headline is then retrieved from the HTML using the Beautiful Soup library[2] (which in turn uses `html.parser`[3] to parse the html).
This script was run every hour for 100 days from 06/11/2016 to 15/04/2017, making for a total of 51,618 stored headlines, although not all are unique, due to headlines changing less frequently than the polling period.

[1] `https://docs.python.org/3/library/urllib.request.html`
[2] `https://www.crummy.com/software/BeautifulSoup/`
[3] `https://docs.python.org/3/library/html.parser.html`

	Source 1	Source 2	...
2016-11-06 12:00:04	*headline*	*headline*	...
2016-11-06 13:00:04	*headline*	*headline*	...
2016-11-06 14:00:04	*headline*	*headline*	...
2016-11-06 15:00:04	*headline*	*headline*	...
⋮	⋮	⋮	

Figure 2: The structure of the CSV file that stored retrieved headlines.

2.2 Moving the data into Python

The file `HeadlineData.py` defines the two Python classes that are used to store and manipulate the data. The `Headline` class stores a single headline with metadata (data and time of publication, source).

```
>>> headline1
Headline('Trump beats Clinton to take White House', \
         'BBC', datetime.datetime(2016, 11, 9, 10, 0))
```

The `HeadlineData` class is comprised of `Headline` objects and stores all data that was collected, it has methods to allow for sorting of the headlines by source, date etc., and includes the classmethod `from_file`, which creates a `HeadlineData` object by importing the data from the CSV file.

2.3 Evaluating Headline Sentiments

In order to evaluate the sentiment of each headline, the VADER Sentiment Analysis[4] tool was used, it is a is a lexicon and rule-based sentiment analysis tool[2], and enables each headline to be scored on its positivity or negativity. Each headline string was given sentiment polarity scores using the tool, and these were stored in an attribute of the Headline objects.

3 Results

In total, 51,618 `Headline` objects were created by collecting headlines every hour from 06/11/2016 to 15/04/2017. The total number of unique headlines was 12,926.
Some organisations updated their main headline more frequently than others (Figure 3), so had a larger contribution to the total number of unique headlines.

[4] `https://github.com/cjhutto/vaderSentiment`

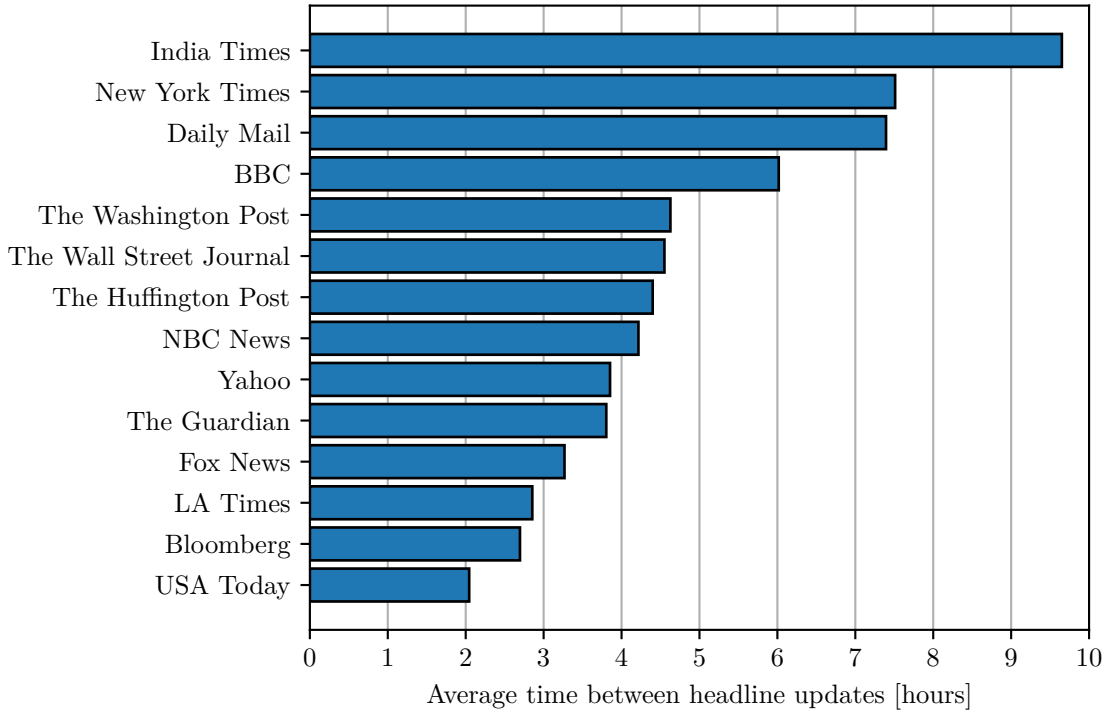

Figure 3: The average lifetime of headlines from different organisations.

3.1 Vocabulary

Table 2 shows the most common words from all the unique headlines that were collected. To investigate if key dates in the American presidential election process affected the number of usages of 'Trump', a plot of its usage against time was generated (Figure 4).

Table 2: The 10 most commonly occurring words > 4 letters long.

Rank	**Word**	**Proportion of unique headlines containing**
1.	trump	28.918%
2.	trump's	7.233%
3.	after	6.916%
4.	house	4.255%
5.	russia	4.023%
6.	obama	3.288%
7.	white	3.009%
8.	attack	2.963%
9.	president	2.329%
10.	travel	2.274%

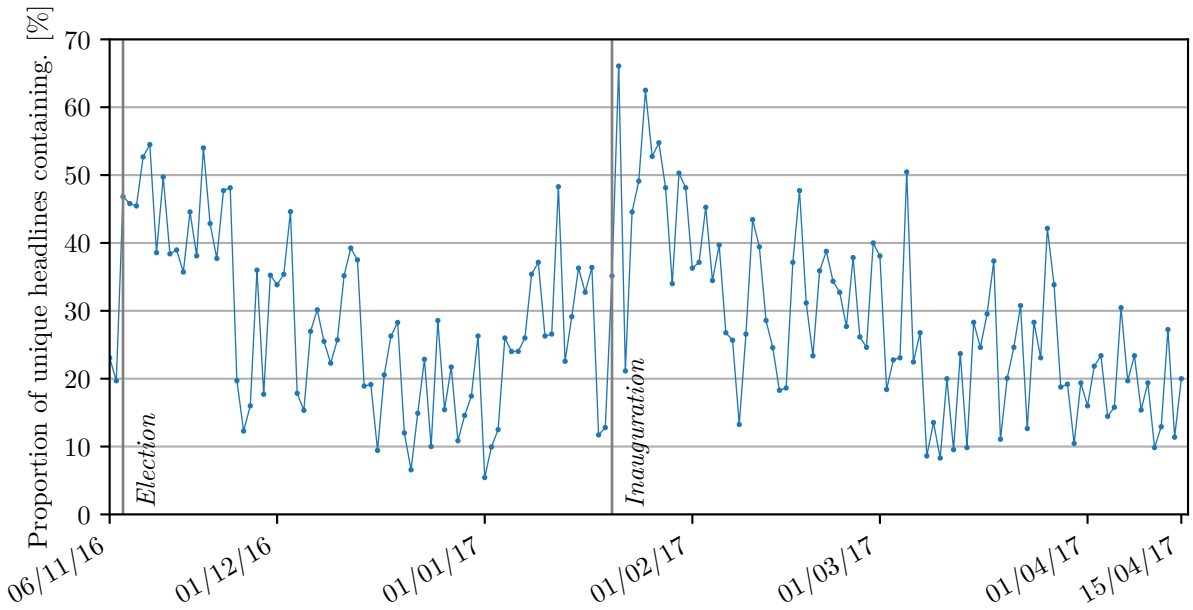

Figure 4: Proportion of headlines per day containing the string 'trump'.

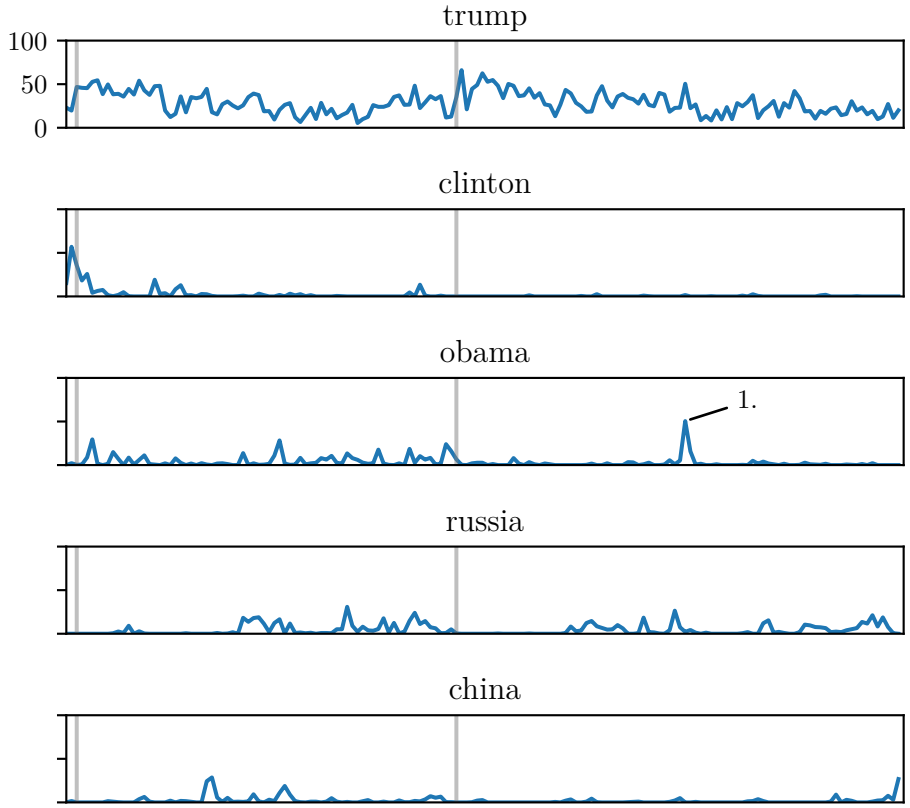

Figure 5: Changes in the usage of some other proper nouns.
1. Trump accuses Obama administration of wiretapping. (5/3/2017)

3.2 VADER Sentiment Scores

Figure 6 shows the score distribution for all unique headlines.

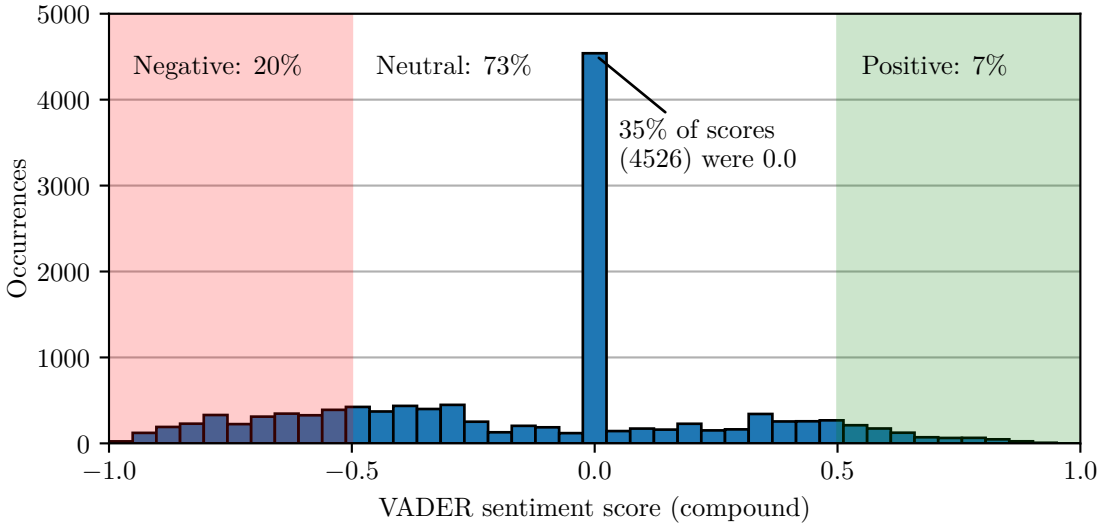

Figure 6: The distribution created by scoring all headlines using the VADER sentiment analyser. Scores above 0.5 show a positive sentiment, below -0.5 a negative one.

The headlines with the most positive, and the most negative sentiment can be found:

```
>>> # Import the pickled HeadlineData object from file
>>> import pickle
>>> headline_data = pickle.load(open('headline_data.p', 'rb'))
>>> headlines = list(headline_data.unique_headlines())

>>> # Print the most positive and most negative headlines:
>>> print(max(headlines, key=lambda x: x.vader_sentiment_scores['compound']))
"""
'We're going to be like gypsies - except our cars are insured': Clarkson
takes full advantage of his new freedom from the BBC as petrolheads
declare his £160m Grand Tour BETTER than Top Gear
 -Daily Mail
2016-11-18
"""
>>> print(min(headlines, key=lambda x: x.vader_sentiment_scores['compound']))
"""
Turkish gunman who killed British boy is shot dead at his own wedding
six days after being released from prison as the father of the dead
two-year-old says he 'doesn't take any joy' from the killing
 -Daily Mail
2017-03-08
"""
```

To see what factors affected the VADER sentiment score of a headline, plots vs. the source of the headline and its length were produced. (Figures 7 & 8)

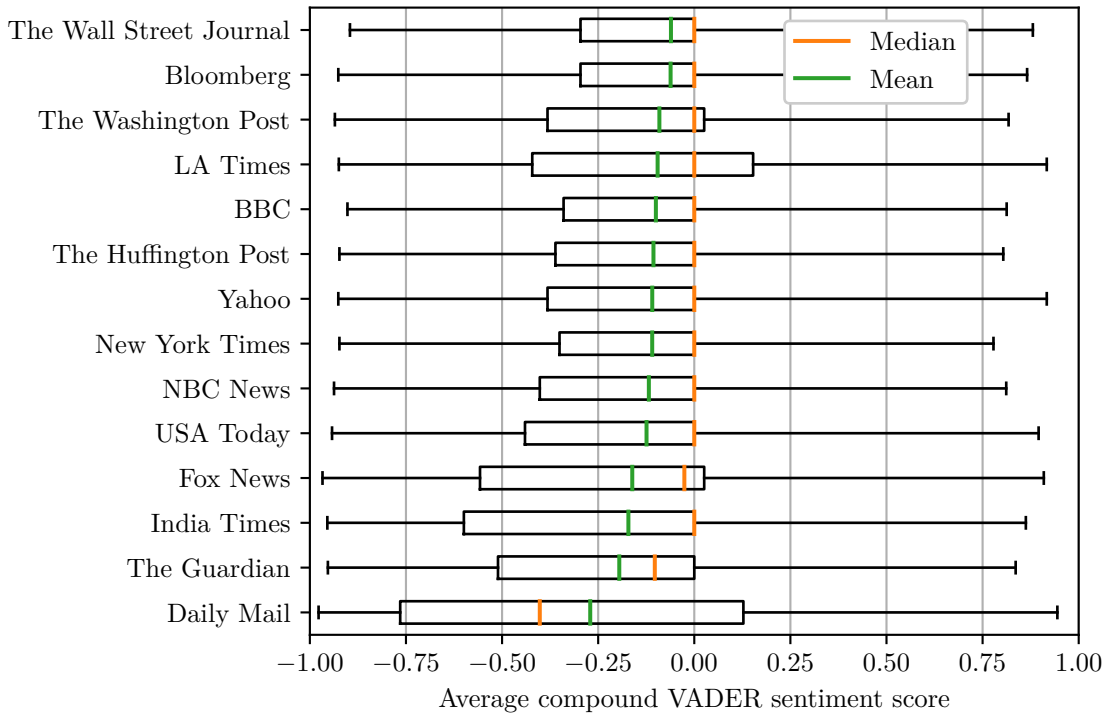

Figure 7: A plot of the VADER Sentiment score distributions for each news organisation.

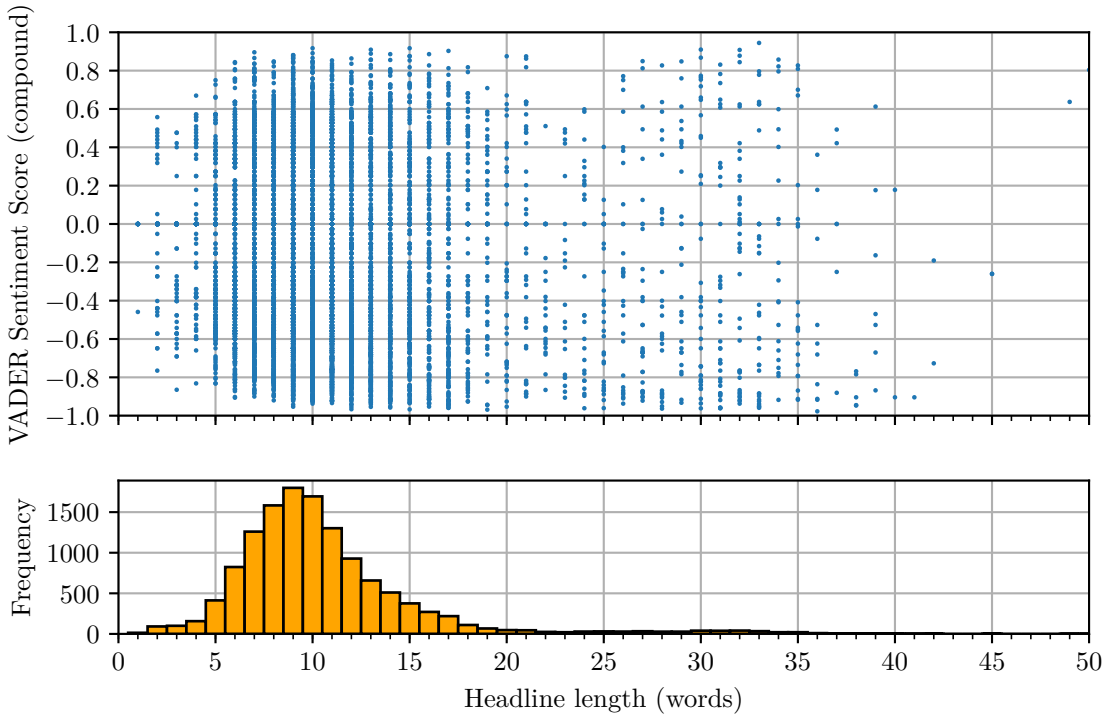

Figure 8: Plot showing distribution of headline lengths, as well as the sentiment scores received for headlines of each length.

Finally, in order to investigate if certain words were generally associated with an overall positive or negative sentiment, each word was given a weighted score based on the VADER sentiment scores of all the headlines which it appeared in, for more detail see `word_stigma.py`. Table 3 shows the results, so for example, headlines containing the word 'suicide', had a VADER sentiment score of -0.861 on average.

Table 3: Words with the largest absolute average sentiment score for all headlines that they appear in.

Negative		**Positive**	
suicide	-0.861	best	0.679
milan	-0.846	awards	0.609
feared	-0.820	confidence	0.603
rape	-0.814	optimism	0.603
horror	-0.813	wins	0.575
terrorist	-0.808	gain	0.574
refusing	-0.785	super	0.558
injured	-0.785	win	0.521
killed	-0.779	freedom	0.511
tragedy	-0.773	commitment	0.500

References

[1] Alexa. *Top Sites by Category: News*. Available: `http://www.alexa.com/topsites/category/Top/News`. (Accessed: 05/12/2016).

[2] E.E Hutto C.J. & Gilbert. ''VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text''. In: *Eighth International Conference on Weblogs and Social Media (ICWSM-14). Ann Arbor, MI, June 2014* (2014).