Refining Fake News Classification with Sentiment Analysis

MICHAEL PRAPPAS, Southern Methodist University

The identification of fake news is a pressing issue for media outlets and Internet platforms. Existing methods for identifying fake news target notorious sources and use basic word counts on the articles. These classifiers are more effective than expected, but they are insufficient for the certainty under which an identifying entity must operate. A random forest classifier using sentiment analysis greatly improves upon existing solutions.

Key Words and Phrases: Sentiment analysis, Naïve Bayes classifier, random forest classifier

1 INTRODUCTION

The world of 2017 is reeling from the year dubbed by the Oxford Dictionary as the year of "post-truth" [1]. The United States presidential election of 2016 brought to light the ugly side of online advertising, social media, and universal Internet use. Digital natives and their baby boomer parents wielded their global platforms to propagate news that was more than biased – headlines that were completely false flew around the Internet, propelled by users whose opinions the articles supported. An army of fact checkers and professional journalists failed to stop the public from believing the content that they were digesting. The gatekeepers of the Internet, most notably Google, Facebook, and Twitter, attempted to stem the flow from notorious websites; not only were their efforts complicated and haphazard, but they also received accusations of censorship. Educators and journalists are still working to educate the digitally naïve, though their attention may be too late for the millions of Americans with hardened biases and online habits. Meanwhile, fake news entrepreneurs are bringing in a small fortune in advertising revenues from the hits to their articles.

At the end of 2016, a dataset of fake news was published on Kaggle [2]. The Google Chrome extension B.S. Detector developed by Daniel Sieradski crawled the internet for articles and classified them on a scale of bias based on their source [3]. From there, George McIntire compiled a corresponding dataset of real news and built a Naïve Bayes classifier using word and bigram features [5]. McIntire achieved a cross-validated accuracy of almost 92%. From there, sentiment analysis can be used to extract more contextual information from the articles. Two leading sentiment analyzers give information on the positivity, negativity, neutrality, polarity, and subjectivity of any subset of an article. A random forest classifier on selected features results in a cross-validated accuracy exceeding 96%, which improves upon McIntire's already outstanding solution. The most helpful features for classification and the articles that are difficult to identify as fake news reveal interesting information for the future of judging fake news.

2 PREEXISTING DATA AND CLASSIFIERS

2.1 Collecting Fake News

The problem of identifying fake news is politically charged and highly uncertain. Even the highest quality news sources are known to have biases in their reporting. Opinion pieces and attention-grabbing writing helps news publishers stay in business. News outlets are rewarded for being among the first to report on breaking news. Especially in the cases of disasters and major events, it is common for news organizations to publish whatever information they have and correct themselves later. In the age of social media, this problem has only been amplified. On channels like Facebook and Twitter, writers have immediate access to hearsay and jumbled pieces of eyewitness coverage. For this reason and many others, including political discussions and misrepresentation of facts and figures, it is impossible to amass a single source of "news truth." This ideal

Publication date: April 2017.

would allow for perfect fact-checking. While expert fact-checking is in fact possible, it is a very difficult problem to have computers reproduce their analytic efforts in real time.

Despite these challenges, the public demands a way to take on the specter of fake news. In November of 2016, Megan Risdal posted a dataset of fake news on Kaggle in order to have data scientists take on this problem [2]. The dataset contains articles, metadata, and a bias label from the Chrome extension B.S. Detector [3]. Daniel Sieradski created the tool as a "rejoinder to Mark Zuckerberg's dubious claims that Facebook is unable to substantively address the proliferation of fake news on its platform." The B.S. Detector references a list of unreliable news sources curated by expert analysts at OpenSources [4]. The list has specialized categories of bias, including satire, conspiracy theories, repressive state news, and hate groups. Such a list is subject to manipulation by its curators, but the public's easy access to the tool in their web browser proves that such a tool can inform individuals' decisions on the news they consume.

George McIntire, a contributing data science writer to Open Data Science, published his work using the Kaggle dataset in March of 2017 [5]. The corresponding problem to identifying fake news – the selection of "real" news – proved to be difficult for McIntire. McIntire began at AllSides, a website that collects articles from all political angles [6]. The articles he scraped came from leading publishers including CNN, the Wall Street Journal, and the New York Times. McIntire combined this data with a simplified fake news dataset. The resulting dataset has four attributes: a unique identifier, the title of the article, the text of the article, and a fake or real stamp. Along with his analysis, McIntire published his dataset for public use. That dataset is the starting point for this paper.

2.2 The McIntire Classifier

McIntire built and tuned a standard text classifier to decide if an article was real or fake. He did not expect great results, but he intended to establish a baseline for fake news classification. McIntire tested two standard feature creation methods for spam detection: a count vectorizer, which tallies word appearances, and a tfidf matrix, which tallies words relative to how often they appear in the corpus. McIntire implemented a Naïve Bayes classifier, the standard for text mining. A Naïve Bayes classifier assumes that all the features are independent; while certain words or pairs of words may appear frequently with one another in articles, it is not unreasonable to assume that one usage is independent from another. McIntire's testing showed that the best model trained on the whole articles rather than just the titles and used a count vectorizer rather than a tfidf matrix. In addition, McIntire determined the best parameters for processing the articles for the count vectorizer. The model's cross-validated accuracy was 91.7%, and the area under the ROC curve is 0.95. These results are outstanding, even surprising McIntire. With a no-information rate of about 50%, McIntire expected that the best results would be in the range of 70%.

2.3 Takeaways from the McIntire Classifier

McIntire published a great deal of helpful information, but he did not publish his final model. In order to grasp the details of the model and drill down into the feature creation step, McIntire's work was reproduced in Python [7]. First, the count vectorizer from the Python machine learning package *scikit-learn* was implemented [8]. The parameters for the count vectorizer are as follows:

CODE SEGMENT 1: Count Vectorizer (*scikit-learn*) Implementation

```
vectorizer = CountVectorizer( input = just_articles, decode_error = 'replace',
                              strip_accents = 'ascii', analyzer = 'word',
                              ngram_range = (2, 2), stop_words = 'english',
                              lowercase = True, min_df = 3 )
```

Publication date: April 2017.

The articles are stripped of capitalization, punctuation, accents and stop words, and only words that appear more than three times in the corpus are kept for analysis. Numbers in the articles are replaced with 0, so that a 0 represents any time that a number appeared. Bigrams, or pairs of adjacent words, are extracted from the articles. The 6,335 articles that McIntire published online give 122,930 unique bigrams. 3,164 or 49.9% of the articles are fake, and the other half are real articles.

As is common in text mining, the 6,335 x 122,930 feature matrix is incredibly sparse. McIntire did not indicate how many features he used in his classifier, so feature selection was used to identify the most salient features. A chi-squared test was tested for different numbers of features, k. A 10-fold cross-validated Naïve Bayes classifier was created for each k, and the accuracy and area under the ROC curve for each k can be found in Table 1. The specific implementation of the classifier is a Bernoulli Naïve Bayes classifier, as the class label is binary (fake or not).

Table 1. Naïve Bayes Classifier Results for Number of Top Chi-Squared Features, *k*

k	Accuracy	Area under the ROC curve
100	0.7907159	0.8819126
500	0.8317498	0.9355802
1000	0.8478489	0.9488339
5000	0.8975711	0.9748751
10000	0.9092491	0.9823536
50000	0.9520178	0.9927263

The value of k that gives the results closest to McIntire's is k = 500. While this value of k does not give the best accuracy or area under the ROC curve, it is the best tradeoff; 500 features dramatically reduces the number of features (0.4% of the original number), helps to eliminate overfitting, and offers room for improvement in accuracy and area under the ROC curve. Reproducing McIntire's results exactly would require more detailed information on the features selected and the parameters of his Naïve Bayes classifier.

3 SENTIMENT ANALYSIS

3.1 Feature Extraction

3.1.1 Sentiment Analysis Tools. Sentiment analysis is the NLP task of extracting information about the sentiment of natural text. The sentiment is a result of word choice, modifiers, and relationships between words. Two sentiment analysis tools were used on this dataset. The first, VADER (Valence Aware Dictionary and sEntiment Reasoner), is a rule-based tool for sentiment analysis in Python that was originally developed to parse text from social media [9]. The social media application may prove helpful for the sensational and attention-grabbing nature of headlines and news articles. VADER is publicly available through the distribution of the Natural Language Toolkit (NLTK) [10]. The second sentiment analysis tool leverages the TextBlob library for commonly-used NLP operations [11]. This more standard sentiment analysis tool will offset any information lost by the specific application of VADER. In addition, VADER and TextBlob encapsulate the sentiment of a text selection with features that have different meanings.

VADER returns four scores when it analyzes a selection of text. A compound score is a single measure of the sentiment between -1 (most negative) and +1 (most positive). The other three scores reflect the portions of the selection that are positive, neutral, and negative. The three scores, which add up to 1, are a multidimensional representation of the sentiment of the selection.

TextBlob's sentiment analyzer returns two scores for a given piece of text. The polarity is a score between -1 (most negative) and +1 (most positive), similar to the compound score from VADER. The subjectivity measure reflects if the text is very objective (0), very subjective (+1), or in between. Table 2 reviews the sentiment analysis metrics that will be used.

Table 2. Sentiment Analysis Metrics from VADER and TextBlob

Metric	Package	Range
Compound	VADER	[-1, 1]
Positive	VADER	
Neutral	VADER	[0,1] Sum = 1
Negative	VADER	
Polarity	TextBlob	[-1, 1]
Subjectivity	TextBlob	[0, 1]

3.1.2 Feature Creation. Text of any length can be passed to VADER and TextBlob for sentiment analysis. As a result, it is possible to look at any subset of an article. The most natural subset for a news article is a sentence, as decent quality news is generally written such that a sentence encapsulates a single idea. Using NLTK's sentence tokenizer, each article was broken up into sentences [10]. The six sentiment analysis metrics in Table 2 were collected for the title, the first sentence of the article, and the last sentence of the article. In addition, the most positive, neutral, negative, and subjective scores were recorded for each article. Furthermore, the highest and lowest polarity scores were collected. Finally, each of the six metrics was averaged by sentence. The sentence average was used rather than analyzing the whole article in order to minimize the impact of especially long quotes and ideas on the sentiment of the article as a whole.

3.2 Feature Reduction

As was done with the McIntire classifier, a chi-squared test was performed to determine the most important sentiment features for determining if an article is fake or real. Tables 3.1 and 3.2 show the results of the chi-squared test.

Table 3.1. Chi-Squared Test Results on Sentiment Features, Unimportant Attributes

Sentence(s)	Metric	Sentence	Metric
Title	Compound	First	Compound
Title	Negative	First	Negative
Title	Neutral	First	Positive
Title	Positive	First	Polarity
Title	Polarity	First	Subjectivity
Title	Subjectivity	Last	Compound

Note: Unimportant attributes have an attribute importance score of 0.

Table 3.2. Chi-Squared Test Results on Sentiment Features, Important Attributes

Sentence(s)	Metric	Attribute Importance
All	Max Positive	0.21321541
All	Max Subjectivity	0.20296909
All	Avg Compound	0.19837011
All	Min Polarity	0.19497042
Last	Neutral	0.19253449
All	Avg Negative	0.16982564
All	Avg Neutral	0.16706355
All	Max Polarity	0.16446193
All	Max Negative	0.14360528
Last	Positive	0.14156118
All	Avg Polarity	0.14137369
All	Max Neutral	0.13824885
All	Avg Positive	0.13775082
Last	Subjectivity	0.13166259
Last	Polarity	0.13009030
All	Avg Subjectivity	0.12324180
First	Neutral	0.09195362
Last	Negative	0.06512418

Notably, all of the title features and all but one of the features of the first sentence are relatively unimportant. The last sentence, however, provides many helpful features in determining if an article is real or fake. This may be due to the nature of attention-grabbing leading sentences; both reputable and fake news organizations use them to catch a reader's eye. Both the extreme and average features are important, and there is no trend to suggest that one of the six metrics is particularly more helpful than another.

3.3 Feature Inclusion

Given 12 unimportant features, 18 of the original 30 sentiment features should be used for classification. However, the features are highly correlated, and perhaps not using all the features would yield a better classifier than using them all. As a starting point, recursive partitioning trees were used to determine the best number of features to include. Table 4 includes the 10-fold cross-validated results of recursive partitioning trees using different numbers of features.

Table 4. Recursive Partitioning Results for Different Numbers of Top Features, k

k	10-Fold Cross-Validated Accuracy
4	0.6000000
6	0.6120063
8	0.6328594
10	0.6372828
12	0.6369668
14	0.6406003
16	0.6406003
18 (all)	0.6406003

After k = 14, accuracy is not improved by including more features. Notably, this untuned classifier has an accuracy of about 64%, which is in line with McIntire's expectations going into his study. Therefore, the top 14 features from Table 3.2 will be used for classification going forward.

4 RESULTS AND ANALYSIS

4.1 Classifier Comparison

With the classification dataset complete, the next task is selecting the best classifier. A Naïve Bayes classifier like McIntire used will no longer be useful since the assumption of feature independence is clearly violated. To facilitate the classification task, the R package caret [12, 13]. Table 5 includes the many classifiers that were tested and their corresponding cross-validated accuracy and kappa measures.

Table 5. Cross-Validated Results of Classifiers on Final Sentiment Feature Dataset

Classifier	Accuracy	Kappa
Recursive Paritioning (CART)	0.6397823	0.2794821
Conditional Inference Tree	0.7452250	0.4904563
C4.5	0.7136430	0.4272483
Rule-Based (PART)	0.6786100	0.3571448
Linear Support Vector Machine	0.6069461	0.2137527
Artificial Neural Network	0.6735694	0.3470616
Random Forest	0.9695329	0.9390653

The random forest classifier has the best classification results by far, by both the accuracy and kappa metrics. The ensemble method works well given a small set of features that are relatively good individually. Other ensemble classifiers were attempted, but no solution was reached given time and computational constraints.

4.2 Incorrect and Correct Identifications

The 96.9% accuracy of the random forest classifier is exceptional, but its misidentified articles may hold more information for improvement. Twenty-five of the 26 articles are fake news that were incorrectly identified as fake, with one real article incorrectly identified as fake. Their labels from McIntire's original dataset are 2623, 5445, 6084, 6268, 6338, 6474, 6647, 6747, 6978, 7046, 7300, 7620, 8253, 8751, 8825, 8826, 8889, 9368, 9713, 9799, 9961, 9981, 10329, 10380, 10452, and 10526. Figure 1 contains the misidentified real article and one of the misidentified fake article.

Figure 1.1 Misidentified Real Article

<table>
<tr><td colspan="5">X = 2623
"7 Times Obama Failed to Support Israel"
Article contents: "Should the U.S. Continue to Support Israel?"</td></tr>
<tr><td><u>Title Scores</u></td><td><u>First Sentence Scores</u></td><td><u>Last Sentence Scores</u></td><td><u>Sentence Avg. Scores</u></td><td><u>Sentence Max. Scores</u></td></tr>
<tr><td>Compound = -0.1531
Negative = 0.33
Neutral = 0.4
Positive = 0.27
Polarity = -0.5
Subjectivity = 0.3</td><td>Compound = 0.4019
Negative = 0
Neutral = 0.69
Positive = 0.31
Polarity = 0
Subjectivity = 0</td><td>Compound = 0.4019
Negative = 0
Neutral = 0.69
Positive = 0.31
Polarity = 0
Subjectivity = 0</td><td>Compound = 0.4019
Negative = 0
Neutral = 0.69
Positive = 0.31
Polarity = 0
Subjectivity = 0</td><td>Negative = 0
Neutral = 0.69
Positive = 0.31
High Polarity = 0
Low Polarity = 0
Subjectivity = 0</td></tr>
</table>

Figure 1.2 Misidentified Fake Article

X = 7620 "Russia Demands Explanation After US Hacks Entire Russian Infrastructure" Article contents: "Ever since Wikileaks and hacking groups began releasing incriminating evidence against the Democrat National Committee and their presidential nominee Hillary Clinton, the US establishment has…" (ends with ellipsis)				
Title Scores	First Sentence Scores	Last Sentence Scores	Sentence Avg. Scores	Sentence Max. Scores
Compound = 0	Compound = 0	Compound = 0	Compound = 0	Negative = 0
Negative = 0	Negative = 0	Negative = 0	Negative = 0	Neutral = 1
Neutral = 1	Neutral = 1	Neutral = 1	Neutral = 1	Positive = 0
Positive = 0	Positive = 0	Positive = 0	Positive = 0	High Polarity = 0
Polarity = 0	Polarity = 0	Polarity = 0	Polarity = 0	Low Polarity = 0
Subjectivity = 0.3125	Subjectivity = 0	Subjectivity = 0	Subjectivity = 0	Subjectivity = 0

Though these articles are not particularly interesting, it is a good sign because the classifier had to make a decision on an arbitrarily short article. However, the classifier's performance on these two articles is still worth noting. The title of the misidentified real article sounds fake, but calling something a "failure" is a matter of opinion rather than an undisputable fact. The title of the misidentified fake article sounds real, but an informed reader would be able to deduce that hacking an entire nation's infrastructure is not feasible. Since these articles do not reveal much information on the power of the classifier, Figure 2 is included below.

Figure 2.1 Correctly Identified Fake Article

X = 9143 "Israel: Ancient Papyrus Proves Jerusalem Belongs to Israel" Fragment of Old Tax Bill Meant to Undercut Muslims' Claim to Important Mosque by Jason Ditz, October 26, 2016 Share This While the UNESCO resolution which recognized the al-Aqsa Mosque in Jerusalem as a "Muslim holy site of worship" was barely reported around the world, and considered fairly non-controversial, Israeli officials have been expressing fury over the matter for two solid weeks. And the Muslims may have a huge, ancient mosque that has been a key part of Islam for 1,300 years, but Israel has a small strip of papyrus they found in a cave, which they're pretty sure is a far more conclusive document, since it mentioned the word Jerusalem and was written in Hebrew. Israeli officials have claimed that the UNESCO resolution, in recognizing the mosque as important to Islam, was tantamount to denying Israel's absolute and eternal control over the entire city of Jerusalem. Israeli Culture Minister Miri Regev said the papyrus strip proved Jerusalem "was and will remain the eternal capital of the Jewish people." The al-Aqsa mosque was built on a site which is believed to have previously housed an important Jewish temple, and some Israelis advocate the eventual destruction of the mosque and the construction of a new temple, though the details of such a construction would be hugely religiously complicated, and since the destruction of the mosque would undoubtedly start a massive war, it is considered unlikely. Still, the far-right government wants to ensure that they have some international precedent for their claim to the territory. Last 5 posts by Jason Ditz				
Title Scores	First Sentence Scores	Last Sentence Scores	Sentence Avg. Scores	Sentence Max. Scores
Compound = 0	Compound = 0	Compound = 0	Compound = 0.02937	Negative = 0.123
Negative = 0	Negative = 0.056	Negative = 0	Negative = 0.0418333	Neutral = 1
Neutral = 1	Neutral = 0.843	Neutral = 1	Neutral = 0.881	Positive = 0.162
Positive = 0	Positive = 0.101	Positive = 0	Positive = 0.07716667	High Polarity = 0.25
Polarity = 0.3	Polarity = 0.25	Polarity = 0	Polarity = 0.1157087	Low Polarity = - 0.07
Subjectivity = 0	Subjectivity = 0.46	Subjectivity = 0.06667	Subjectivity = 0.45742	Subjectivity = 0.84167

Figure 2.1 Correctly Identified Real Article

X = 626
"Donald Trump Is Changing His Campaign Slogan to Prove He's Not Racist"

After a week of nonstop criticism from Democrats and Republicans alike for comments many condemned as racially charged, Donald Trump claims to be altering his campaign to be a little more inclusive. While the presumptive G.O.P. has long promised to make America great again, Trump now says he's adding two words to slogan to illustrate just how non-racist he really is. "You know, I have the theme 'make America great again,' and I've added a couple of things," Trump announced to supporters at a campaign rally in Richmond, Virginia, on Friday night. "Right now I'm adding make America great again,' adding 'for everyone,' because it's really going to be for everyone. It's not going to be for a group of people, it's going to be for everyone. It's true." The allegedly amended slogan, which has yet to appear on any official signage or Trump merchandise, comes after the presidential candidate spent the first half of June repeatedly denouncing Gonzalo Curiel, the federal judge of Mexican heritage presiding over the Trump University class action lawsuit, as inherently biased against him. (Curiel was born in Indiana.) His comments were widely condemned by the Washington political establishment, including Senate Minority Leader Mitch McConnell, who suggested he may be an idiot, and House Speaker Paul Ryan, who called Trump's statement the "textbook definition of a racist comment." Trump, who hasn't apologized or taken back any of his comments, indicated on Friday that he realized his words have had a negative effect on his campaign and declared he is not a racist. "I am the least racist person. The least racist person that you've ever seen. I mean give me a break," he said at the rally. "I am the least racist person that you've ever looked at, believe me."

Title Scores	First Sentence Scores	Last Sentence Scores	Sentence Avg. Scores	Sentence Max. Scores
Compound = -0.6124	Compound = -0.765	Compound = 0.4973	Compound = 0.11539	Negative = 0.22
Negative = 0.267	Negative = 0.22	Negative = 0	Negative = 0.039	Neutral = 1
Neutral = 0.733	Neutral = 0.78	Neutral = 0.788	Neutral = 0.8404615	Positive = 0.392
Positive = 0	Positive = 0	Positive = 0.212	Positive = 0.1205385	High Polarity = 0.8
Polarity = 0	Polarity = 0.2708333	Polarity = -0.3	Polarity = 0.03163004	Low Polarity = -0.313
Subjectivity = 0	Subjectivity = 0.5	Subjectivity = 0.4	Subjectivity = 0.35614	Subjectivity = 0.75

The fake article in Figure 2.1 is significant because it is not particularly extreme. It contains sentences that are completely neutral (neutral score = 1), and no sentence gets very negative or polar (scores ≤ 0.25). However, the high max subjectivity of the article stands out. This score may reflect the many extreme words in the article, including "fury," "far more conclusive," "hugely religiously complicated." As a side note, the article contains two links to share the article; the sentiment classifier would not pick up on this, but it is a common sight in fake news articles as views keep the publishers generating advertising revenues.

The real article in Figure 2.2 sounds politically charged, but a great deal of the article is quoting presidential candidate Donald Trump. The article contains both very polar and subjective sentences, as reflected by phrases including "condemned as racially charged," "widely condemned," and "he may be an idiot." The classifier is able to parse opinionated quotes without throwing off the article classification.

5 CONCLUSIONS AND FUTURE RESEARCH

Beginning with the standard text mining approach to the fake news problem, the effectiveness of McIntire's classifier was proven and refined. Two different sentiment analyzers were chosen to build a sentiment feature dataset for the given articles. A holistic sentiment analyzer is needed, as chi-squared testing showed that different features from VADER and TextBlob were effective when used together for classification. An accurate sentiment classifier showed that fake news can be identified with a subset of less than 15 sentiment features. The random forest ensemble method emerged as the most capable classification method given the time and computational constraints of this project.

Despite the accuracy of the model, there is still room for improvement. The articles used for training and testing this classifier come from a relatively short period of time. In that span of time, a collection of the same names, events, etc. are mentioned by many articles. This limitation is more of a problem for the McIntire classifier, but the sentiment analyzer is also subject to changes in content and style over time. In addition, an ensemble classifier could be created that uses both CountVectorizer and sentiment data. Finally, in order to roll out a more capable fake news identification tool than the B.S. Detector, outside experts must help train the model. Specifically, there is are high costs associated with identifying fake news as real and vice-versa, and an agreement must be reached on the tradeoff. Going forward, however, sentiment analysis will be an indispensable tool for parsing and classifying fake news.

REFERENCES

[1] Oxford Dictionaries. "Word of the Year 2016 is…" https://en.oxforddictionaries.com/word-of-the-year/word-of-the-year-2016
[2] Risdal, Megan. "Getting Real about Fake News." *Kaggle*. November 2016. https://www.kaggle.com/mrisdal/fake-news
[3] Sieradski, Daniel. "B.S. Detector." *The Self Agency, LLC*. http://bsdetector.tech
[4] OpenSources. "OpenSources: Professionally curated lists of online sources, available free for public use." http://www.opensources.co
[5] McIntire, George. "On Building a 'Fake News' Classification Model.'" *Open Data Science*. 22 March 2017. https://opendatascience.com/blog/how-to-build-a-fake-news-classification-model/
[6] AllSides. https://www.allsides.com
[7] Python Software Foundation. Python Language Reference, version 2.7. Available at http://www.python.org.
[8] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, Édouard Duchesnay. Scikit-learn: Machine Learning in Python, *Journal of Machine Learning Research*, 12, 2825-2830 (2011). http://scikit-learn.org/stable/
[9] Hutto, C.J. & Gilbert, E.E. (2014). "VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text." Eighth International Conference on Weblogs and Social Media (ICWSM-14). Ann Arbor, MI, June 2014. (VADER on GitHub: https://github.com/cjhutto/vaderSentiment)
[10] Bird, Steven, Edward Loper and Ewan Klein (2009). *Natural Language Processing with Python*. O'Reilly Media Inc.
[11] Loria, Steven. TextBlob, version 0.12.0. http://textblob.readthedocs.io/en/dev/index.html
[12] R Core Team (2016). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.
[13] Max Kuhn. Contributions from Jed Wing, Steve Weston, Andre Williams, Chris Keefer, Allan Engelhardt, Tony Cooper, Zachary Mayer, Brenton Kenkel, the R Core Team, Michael Benesty, Reynald Lescarbeau, Andrew Ziem, Luca Scrucca, Yuan Tang, Can Candan and Tyler Hunt. (2016). caret: Classification and Regression Training. R package version 6.0-73. https://CRAN.R-project.org/package=caret

Publication date: April 2017.