Luke Kvamme, Phillip Garcia, Marc D'Avanzo

CSC-272

12-2-2022

Sentiment Analysis of News Sites' Home Pages

Introduction

In today's climate, many people feel that the news is overwhelmingly negative. Every day there is a new crisis affecting the world, like a new conflict happening between Ukraine and Russia or a local tragedy such as a fatal car crash. In this vein of thinking, this project's central question was to examine if there was a correlation between the day of the week, news source, or the political leaning of the source on the sentiment of the news given, either positive or negative. To accomplish this, we downloaded the HTML of various home pages from different news sources from January 2021 to November 2022 and three sentiment Lexica to help us analyze the HTML files.

We found inconclusive results from the analysis of our data. It was determined that the day of the week contained no correlation with its polarity, as there is a fairly equal distribution of positive and negative data instances per day. Each day maintained the same ratio of positive to negative instances. However, there were notable differences in the polarity of positive and negative stories between "Left" and "Right" leaning political sources, as well as a large variance in the polarity ratio of individual sources.

Road Map

Our report is in 7 sections, laid out in this manner:

- Section 1 discusses the data set in detail Page 2
- Section 2 discusses our data preparation in detail Page 7
- Section 3 discusses our data analysis in detail Page 10
- Section 4 discusses our results Page 13
- Section 5 discusses our overall results Page 33
- Section 6 is our conclusion Page 34
- Section 7 is our appendix Page 36

1. Dataset Description

We have 5 different data sets for this project. One is a testing set and the other 4 are training data sets. There is also a 6th data set used to help with visualization. All data sets have the same basic format: each instance is a specific HTML file of a news source's home page on their website. For that instance we would track the year it was from, the day of the week, the source, its political leaning, and its sentiment. The below table Further explains the data sets.

Attribute	Possible	Description	Data
	Values		Туре
Year	2021	The year in which the HTML file was published	Nominal
	2022		
Day of	Monday	The day of the week the file was published	Nominal
Week	Tuesday		
	Wednesday		
	Thursday		
	Friday		
	Saturday		
	Sunday		
Company	ABC	The names of the sources we captured HTML files	Nominal
Name	Breitbart	from	
	Buzzfeed		
	CBS		
	Daily Kos		
	Daily wire		
	Fox		
	HuffPo		
	MSNBC		

	National		
	Review		
	Slate		
	The blaze		
	Vox		
	Wall Street		
	Journal		
	Washington		
	post		
PolLeaning	Left	The general leaning of the news source. The political	Nominal
	Right	leaning of a news source can vary by subject and	
		author, so this designation is generalized. The	
		following is how each news source is classified:	
		ABC – > Left	
		Breitbart – > Right	
		Buzzfeed -> Left	
		CBS – > Left	
		Daily Kos – > Left	
		Daily wire – > Right	
		Fox – > Right	
		HuffPo – > Left	

		MSNBC -> Right	
		National Review – > Right	
		Slate – > Left	
		The blaze – > Right	
		Vox -> Left	
		Wall Street Journal – > Left	
		Washington post – > Left	
Sentiment	Negative	The class attribute of the data set is determined by	Nominal
	Positive	counting the number of positive and negative words	
		in the file. The category with a higher count denotes	
		its classification.	

The reason there are four different training data sets is that each represents one of three respective sentiment Lexica we found, and the fourth is a combined Lexicon of all three. We gave each Lexicon a simple numerical indicator of 1, 2, or 3 and for the combined we called it Lexicon "C" or "Combined". We wanted to assess the different Lexica to see if we could determine if one was more accurate than the others.

Lexicon 1 is called the MPQA Subjectivity Lexicon from the University of Pittsburgh Computer Science Department (http://mpqa.cs.pitt.edu/#subj_Lexicon). It contains 8,222 different words. Lexicon 2 has no specific name, but was created by Dr. Bing Liu and Dr. Minqing Hu of the University of Illinois Chicago (https://www.cs.uic.edu/~liub/FBS/sentimentanalysis.html#datasets). Their Lexicon contains 6,800 words. Lexicon 3 was found from a website called SenticNet, which describes itself as helping machines learn, leverage, and love (http://sentic.net/downloads/). Their Lexicon contains 150,000 total entries making it magnitudes larger than the other two entries. SenticNet's Lexicon contained not only single words like the other Lexica, but also emojis and up to 4-word n-grams. The emojis and ngrams needed to be removed to bring Lexicon 3 to equal footing with the other two Lexica. After removing the phrases and emojis, Lexicon 3 contains 39,500 words. When all three Lexica were combined together for the fourth Lexicon, all of their unique words total 43,732.

The test data set contains 120 files taken out of the training data set. Instead of having these HTML files be classified using the four Lexica, we manually divided the files between the three of us and parsed them by hand. We kept track of the number of positive and negative words to determine the sentiment of the file once we finished reading it. While classifying the HTML files we recreate the automated algorithm to the best of our abilities. For each of the test files, we went word by word and ignored the context of what the sentence was conveying. For example, if a headline stated "kind and innocent found murdered", there would be 2 positive words, "kind" and "innocent", and one negative word, "murdered". This tedious process took a very long time because some of the sources have up to 10,000 words in their HTML file.

The final data set we created was a conglomeration of the results of Lexica 1 through 3 and C. This file contained all of the counts of positive and negative words for all of the Lexica as

well as the sentiment differences (count of positive - count of negative) used to determine the class attribute. This file was used to compare differences between how the Lexica classified the HTML files. As a class attribute for the file we included an average sentiment, where whatever the majority sentiment from Lexica 1 through 3 was, it became the class attribute/sentiment.

Lexicon	Number of	Positive	Negative	Political Leaning
	Words	Classifications	Classifications	Ratio (L:R)
Lexicon 1	8222	6653	2246	5607:3292
Lexicon 2	6800	1756	7143	5607:3292
Lexicon 3	39500	8670	229	5607:3292
Lexicon	43732	8618	281	5607:3292
Combined				
Test Data	Human	10	110	76:44
	Intuition			

2. Data Preparation

In order to answer if specific news sources, their political leanings, or different days of the week affected the polarity of news, we needed files containing each news sources' website headlines and a way to parse each file and discern the polarity of each word. One thing we were concerned about was getting enough data. Because the project was started part way through the semester, we realized in order to obtain a sufficiently large data set we could not just track news sources from the project commencement to the project conclusion. This led us to want to gather data from the past. In order to gather news from the past, we created a web scraper to systematically crawl through the internet archives of Archive.org and download all of the articles on the home page of each news source for every day from January 1, 2021 to November 4, 2022. To create the web scraper, we had to use Eclipse to create a Maven project so that we could download all of the dependencies necessary for making a web connection and then parsing through all of the HTML, CSS, and JavaScript to grab the data we wanted so that we could then save the website to an output file. For the web connection we used the "JSOUP" package and for the disabling of CSS and JavaScript so we could parse just the words on the website we used "htmlunit". Throughout the web scraping process we encountered many problems that culminated in us shifting away from grabbing just the articles off the home page to instead downloading the entire home page..

Archive.org has a strict policy on web scraping, which took us a long time to get around. It uses JavaScript to detect if the user is actually a human or a bot, in which case it will refuse connection to the request. This resulted in a long process of debugging our code in order to avoid getting our connection refused. At a high level, we had to create functionality to change the user agent every once in a while so as to not flag its bot detection system. We also had to use an insecure SSL connection, otherwise disabling CSS and JavaScript would not function properly. Finally, we made the program sleep for 5 seconds after completing our request.

Archive.org gave us further issues due to having a processing period of around 10-30 seconds per request in order to run its anti-bot check and fetch the past web page from a massive

database. This 10-30 second span of time coupled with the 5 seconds of sleeping per request and the multitude of errors thrown throughout reduced the amount of data we originally desired from the past 5 years to the past 2 years. On one of the mental health days we were given, we even went into the Mac lab at 8am and tried using 15 computers to each run a different source at the same time. This did not work however, and instead resulted in us being blocked by Archive.org for a couple of days.When we were able to access a home page, there was a non significant chance that the file would be unreadable by our web scraper due it having a horrible HTML setup, giving us useless, bad files. The combination of all these issues reduced our time frame of data from January 1, 2021 to November 4th 2022, giving us 8,899 total HTML files of around 100 MB in size.

After we had collected our HTML files we needed to create another program that would scan through the HTML files to calculate the sentiment of the file. This was done with a Python program, which was used to create the four training data sets. The Python file would read through every collected HTML file and then output the data into a new file. The Python program added the file's name, total count of positive and negative words, and the difference between the two counts in addition to the needed attributes. These unneeded attributes were removed manually afterwards to remove any highly correlated values. The program was also used to reformat the different Lexica into CSV formatted files with two attributes: Word and Sentiment.

The program was essentially one large nested for-loop. The program would search through every HTML. When reading an HTML file, the program would go line by line reading every word. If the program saw a word for the first time it would add it to a dictionary, its key being the new word and the value being 1, if the word was already present then the key with the same word would have its value incremented by 1. After every line in the file was read, the dictionary would be iterated through searching for matching words in one of the four Lexica. After the frequency list dictionary was iterated through, the results would be added to a Pandas data frame. The Python program went through several iterations to make it as efficient as possible. In its first version, the program made no attempt to improve efficiency. This was problematic as the amount of iteration through every file and every line caused significant slow downs. In the final version of the program, two dictionaries were used for O(1) searching and accessing. There was one dictionary for the frequency of words found in HTML files and one dictionary for the words in a Lexicon. For this dictionary the key would be a word and its value would be its polarity. After these optimizations were applied to the algorithm, the program went from classifying 500 files in 50 minutes to classifying all 8,899 files in less than 2 minutes.

3. Data Analysis

The focus of this project was on the classification of HTML files into "Positive" or "Negative". This restricted us to only using classification algorithms.

3.1 ZeroR

This is the simplest classification algorithm. Given a supplied training data set, ZeroR will calculate the majority value of the class attribute. It will then predict this majority class attribute for any new test cases. ZeroR ignores all other attributes but the class attribute, and because of this is often used as a baseline for a data set.

For our project, ZeroR will be used to determine which polarity either positive or negative is predicted more in Lexica 1 through 3, C, and the test file.

3.2 OneR

Like ZeroR, 1R is a simple classification algorithm often used as a baseline. Unlike ZeroR, 1R will analyze other attributes of a data set to determine which is the single most predictive of the class attribute. It will then output a set of rules for the most predictive attribute and the class attribute. There will be a rule for each value of the most predictive attribute that best predicts the class attribute. If there are more class attribute values than values the most productive attribute has, then some of the class attribute's values will not have a rule associated with them.

For our project 1R will be used as a baseline and to see if either year, day of week, company name, or political leaning are very predictive of sentiment.

3.3 J48 Decision Tree

For our project we also ran the J48 algorithm in Weka to analyze our data and what attributes it chose. The J48 algorithm results in a tree in which each node is an attribute and the branches will be values of the attribute it is connected to. The reason for this is that when an attribute is chosen by the algorithm, a split occurs where the attribute's criteria determines the path to be taken in the tree. The final level of the tree is the resulting class attribute for an instance. Sometimes not all attributes will be present in a J48 tree and this is due to certain attributes being unpredictive. The algorithm works by choosing attributes in which the best possible split occurs and tries to handle as many instances in a split. The algorithm uses accuracy and information gain to choose the best possible splits. The result is a tree that effectively classifies new instances in the data set. The J48 algorithm was run with and without a cost sensitive classifier for our purposes. The reason we performed this algorithm was to see which attributes the algorithm would select and the certain splits that occurred at each node.

3.4 Support Vector Machine

The support vector machine is an algorithm used to minimize the error when classifying test data. The algorithm works by finding the best line on a graph that splits instances by their class value. The hope of this algorithm is to minimize error and incorrect classifications with the test data set. We used this algorithm to maximize our accuracy and to give insight onto misclassified instances and analyze those instances further. The algorithm was run with and without a cost sensitive classifier. In order to use this algorithm, we had to use the package manager to install the LibSVM package.

3.5 Random Forest

Random Forest is an algorithm that relies on decision trees and ensemble learning to create a model. In simple terms, the random forest works by creating a decision tree of decision trees that are not correlated. The "random" portion of "random" forest comes from the fact that every decision tree is created with a random subset of the data and a random selection of features. The random subset of data is called the bootstrap sample. A specific ensemble method called bagging is used to randomly choose features which ensure low correlation between

individual decision trees within the forest. This algorithm was run with and without a cost sensitive classifier.

4. **Results**

4.1 ZeroR

Lexicon	Accuracy	Rules
Lexicon 1	74.76%	Predicts Positive
Lexicon 2	80.27%	Predicts Negative
Lexicon 3	97.42%	Predicts Positive
Lexicon	96.84%	Predicts Positive
Combined		

ZeroR showed us that the majority of our Lexica predict a positive outcome. Lexicon 1 and 3 both overwhelmingly classify HTML files as positive, while surprisingly Lexicon 2 classified files as overwhelmingly negative.

4.2 OneR

Lexicon	Rules	Accuracy
Lexicon 1	ABC – > Positive	27.5%
	Breitbart – > Negative	
	Buzzfeed – > Positive	
	CBS – > Positive	
	Daily Kos – > Positive	
	Daily Wire – >Negative	
	Fox – >Positive	
	Huffington Post – > Positive	
	MSNBC – >Positive	
	National review – > Positive	
	Slate – >Positive	
	The Blaze – >Negative	
	Vox – >Positive	
	Wall Street Journal – > Positive	
	Washington Post ->Positive	

Lexicon 2	ABC –> Negative	83.33%
	Breitbart – > Negative	
	Buzzfeed – > Positive	
	CBS – > Negative	
	Daily Kos – > Negative	
	Daily Wire – > Negative	
	Fox – > Negative	
	Huffington Post – > Negative	
	MSNBC -> Negative	
	National review – > Negative	
	Slate – > Negative	
	The Blaze – > Negative	
	Vox – > Positive	
	Wall Street Journal – > Positive	
	Washington Post – > Negative	
Lexicon 3	2021 -> Positive	8.33%
	2022 – > Positive	
Lexicon Combined	2021 – > Positive	8.33%

	2022 – > Positive	

Depending on the Lexicon, 1R predicted one of two variables, either company name or year. For Lexicon 1 1R output ruled that company name was most predictive, with most sources predicting a positive classification. The only sources that were predicted to be negative were Breitbart, Daily Wire, and the Blaze. For Lexicon 2, 1R predicted again that source of the HTML file is the most predictive attribute. It was opposite of Lexicon 1, as most companies predicted a negative sentiment, except for Buzzfeed, Vox, Wall Street Journal. These two rules show that Lexicon 1 and Lexicon 2 are more balanced in how they classified the HTML files as there was a mix of both sentiments.

On the other hand for Lexicon 3 and Lexicon combined, 1R predicted that that the year of the article was most predictive. The low accuracy is caused by the large imbalance of positive files that the Lexica classified with the large amount of negative HTML files that we manually classified. Lexicon 3 seems to dominate over the other Lexica for words found due to its increased word count. This leads to the results of Lexicon 3 being over represented in the combined Lexicon.

4.3 J48

4.3.1 Lexicon 1

Base Accuracy: 22.69%

a	b	← Classified As
9	0	a = Positive
92	18	b = negative

Cost Sensitive Evaluator Accuracy: 78.99%

a	b	← Classified As
7	2	a = positive
23	87	b = negative

Lexicon 1's base accuracy of 22.69% is alright on its own, but after performing a cost sensitive analysis of J48 with weights of 20 on guessing false positives and 1 on false negatives we were able to get the accuracy to increase to 78.99%.

4.3.2 Lexicon 2

Base Accuracy: 84.87%

a	b	← Classified As
7	2	a = Positive
16	94	b = negative

Cost Sensitive Evaluator Accuracy: 98.32%

a	b	← Classified As
7	2	a = positive
0	110	b = negative

Lexicon 2 had an amazing base accuracy of 84.87% with J48, and increased even further when it was run again with weights of 20 on false positives and 1 on false negatives using the cost sensitive filter on J48.

4.3.3 Lexicon 3

Base Accuracy: 7.56%

a	b	← Classified As
9	0	a = Positive
110	0	b = negative

Cost Sensitive Evaluator Accuracy: 11.76%

a	b	← Classified As
9	0	a = positive
105	5	b = negative

The base accuracy for Lexicon 3 was only 7.56% so we also applied weights of 20 on the false positives and 1 on the false negatives on it with a cost sensitive J48 and were able to increase the accuracy to 11.76%. Because Lexicon 3 leans so heavily towards the positive side, regardless of what weights we applied against false positives the maximum accuracy we could achieve was 11.76%.

4.3.4 Lexicon C

Base	Accuracy:	7.56%
------	-----------	-------

a	b	← Classified As
9	0	a = Positive
110	0	b = negative

Cost Sensitive Evaluator Accuracy: 11.76%

a	b	← Classified As
9	0	a = positive
105	5	b = negative

Because Lexicon 1 and 3 are so similar in their functionality, the combined Lexicon

results in the same base accuracy of 7.56%. The highest accuracy we could attain with the cost sensitive filter was 11.76% by using a weight of 20 on the false positives.

4.4 Support Vector Machine

4.4.1 Lexicon 1

Base Accuracy: 27.73%

a	b	← Classified As
9	0	a = Positive
86	24	b = negative

Cost Sensitive Evaluator Accuracy: 78.99%

а	b	← Classified As
7	2	a = positive
23	87	b = negative

The base accuracy of SVM was 27.73%. After applying weights of 10 for the false positives and 1 for the false negatives by using the cost sensitive filter, we were able to increase the accuracy to 78.99%.

4.4.2 Lexicon 2

Base Accuracy: 84.87%

a	b	← Classified As
7	2	a = Positive
16	94	b = negative

Cost Sensitive Evaluator Accuracy: 98.34%

a	b	← Classified As
7	2	a = positive
0	110	b = negative

The base accuracy of Lexicon 2 was 84.87% and, as opposed to Lexicon 1 and 3 which overwhelmingly predict positive, we were able to increase the accuracy to 98.34% by using weights of 20 for false positives and 1 for false negatives. This combination allowed us to bring over every instance of negatives that were being predicted positive, but did not improve the false negatives.

4.4.3 Lexicon 3

Base Accuracy: 7.56%

a	b	← Classified As
9	0	a = Positive
110	0	b = negative

Cost Sensitive Evaluator Accuracy: 11.76%

a	b	← Classified As
9	0	a = positive
105	5	b = negative

Cost Sensitive Evaluator Accuracy: 92.44%

a	b	← Classified As
0	9	a = positive
0	110	b = negative

Lexicon 3 did not perform as well as Lexicon 1 or 2, with a base accuracy of 7.56%. By using a weight of 10 on the false positives and 0 on the false negatives, we were able to increase the accuracy to 11.76%. We flew too close to the sun however, and added more weight to the false positives and counterweights to the false negatives in an effort to increase the predictability of the negative values. This seemed like an amazing increase at first glance, 11.76% to 92.44%, however it now predicts all 9 positive values incorrectly. The only reason why the new model's accuracy is 92.44% is because of the overwhelming amount of negatives in proportion to the positives. Therefore, the 11.76% model is arguably more accurate and preferable because it predicts all of the positives correctly and is able to predict some of the negatives properly as well.

4.4.4 Lexicon C

Base Accuracy: 7.56%

a	b	← Classified As
9	0	a = Positive
110	0	b = negative

Cost Sensitive Evaluator Accuracy: 11.76%

a	b	← Classified As
9	0	a = positive
105	5	b = negative

The combined Lexicon gets overly influenced by Lexicon 3 and mirrors its exact behavior, with a 7.56% baseline accuracy and an 11.76% accuracy when a weight of 10 is added to the false positives.

4.5 Random Forest

4.5.1 Lexicon 1

Base Accuracy: 23.33%

a	b	← Classified As
10	0	a = Positive
92	18	b= Negative

Cost Sensitive Evaluator Accuracy: 91.67%

a	b	← Classified As
0	10	a = Positive
0	110	b= Negative

The base accuracy of the random forest on Lexicon 1 is only 23.33% but is arguably better then the 91.67% of the cost sensitive evaluator random forest. While the base accuracy was extremely low it was able to correctly predict all of the manually evaluated positive HTML files while at the same time predict a few of the negative files. Using the cost sensitive evaluator allowed the model to predict all of the negative stories correctly at the cost of all the correct positive stories. No amount of different weight combinations with the evaluator was able to improve on the base random forest. All attempts at changing the costs caused the model to either predict everything as positive or negative.

4.5.2 Lexicon 2

Base Accuracy: 84.17%

a	b	← Classified As
7	3	a = Positive
16	94	b = negative

Cost Sensitive Evaluator Accuracy: 91.67%

a	b	← Classified As
0	10	a = positive
0	110	b = negative

Cost Sensitive Evaluator Accuracy: 41.67%

a	b	← Classified As

8	2	a = positive
68	42	b = negative

Due to the high amount of negative classifications in Lexicon 2 the random forest performed exceptionally well in its base configuration. With an accuracy of 84%, the model predicted a good ratio of both positive classifications and negative classifications correctly. With the cost sensitive evaluator we could not generate an improvement. We were able to achieve an accuracy of 91.67% by putting a cost of 15 on false positives and a cost of 1 on false negatives. This however caused the model to only predict all instances as negative.

Our second attempt that was somewhat close to the base accuracy was by applying a weight of 5 to false positives and 150 to false negatives. This got an accuracy of 41.67%, it was slightly better at predicting positive instances, at 8 instead of 7. However, this came at the cost of 40% accuracy.

4.5.3 Lexicon 3

Base Accuracy: 9.80%

a	b	← Classified As
10	0	A = positive
110	0	$\mathbf{B} = negative$

Cost Sensitive Evaluator Accuracy: 8.33%

a	b	← Classified As
10	0	A = positive
110	0	B = negative

4.5.4 Lexicon C

Base Accuracy: 9.17%

a	b	← Classified As
10	0	A = positive
109	1	B = negative

Cost Sensitive Evaluator Accuracy: 8.33%

a	b	← Classified As
10	0	A = positive
110	0	$\mathbf{B} = \mathbf{negative}$

Both Lexicon 3 and Lexicon Combined have the same issue as Lexicon 1. When using the evaluator on these Lexica, we were unable to find a combination that did not cause the model to predict only one type of classification. We believe that the Lexicon 2 was the only real

successful model because its training data contained a significant number of negative classifications compared to the other three Lexica.

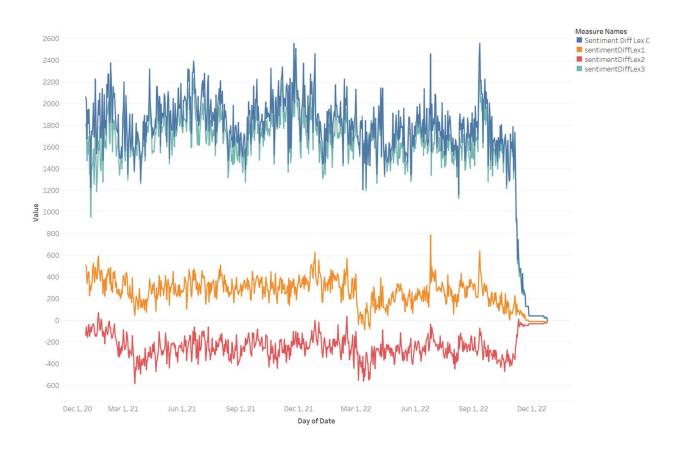


Figure 1: The total sentiment difference per day for Lexicon 1,2,3, and C

Figure 1 shows the total sentiment difference per day for each Lexicon; it is important to note that due to Archive.org's implementation, some HTML files were encoded with the wrong "Publish" date. As mentioned earlier, sentiment difference is calculated by the total count of positive words for a file subtracted from the total number of negative words for a file. This chart allows for an easy way to compare how positive or negative each Lexicon thought a day was. It is important to note that even if a Lexicon has positive or negative sentiment difference sum, there could still be files classified as the opposite value. It just indicates what the majority of files

were classified as. As expected, Lexicon 2 is the only Lexicon that has a consistent negative total sentiment, and this corresponds with its overwhelming negative classification. The chart also confirms our suspicion that Lexicon C is dominated by the results of Lexicon 3 as both Lexica have almost identical total sentiment differences per day.

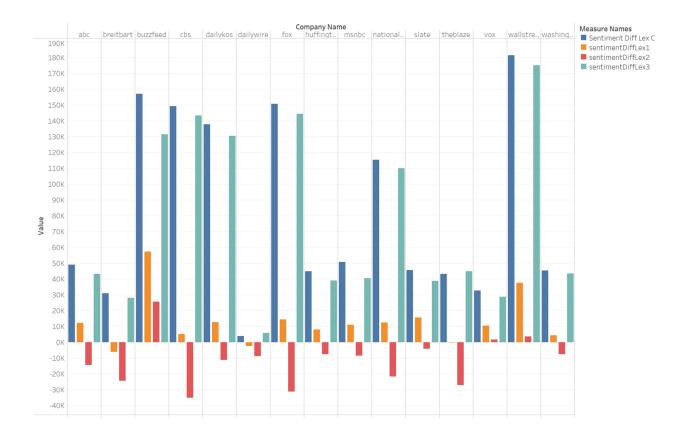


Figure 2: The total sentiment difference by company for Lexicon 1,2,3, and C

Figure 2 shows the total sentiment difference for each of the new sources for each Lexicon. The chart confirms what was shown in Figure 1, that being Lexicon 3 and Combined have the largest sentiment difference values, Lexicon 1 has a modestly positive sentiment difference sum, and Lexicon 2 has a negative sentiment difference. However, we also see that each news source can make a significant difference in the sentiment difference sum. The Wall Street Journal, Vox, and Buzzfeed, seem to contain a significant number of positive words for each Lexicon as they are the only three sources with all the Lexica having a positive sentiment difference sum.

Another insight this visualization brings forward is the lack of correlation between the Lexica. Buzzfeed, Fox and the Wall Street Journal according to Lexicon 3 and Lexicon Combined are extremely positive. They have some of the largest positive sentiment difference sums; however, Lexicon 2 had dramatically different sentiment sums. For Lexicon 2, Buzzfeed compared to all other sources was extremely positive and Fox was one of the most negative, even though Lexicon 3 and Lexicon Combined had similar sentiment differences between Fox and Buzzfeed. Compared to Buzzfeed and Fox, the Wall Street Journal has a larger sentiment difference sum for Lexicon 3 and Lexicon Combined, yet Lexicon 2 only has a marginally positive sentiment difference sum when compared to the Buzzfeed total.

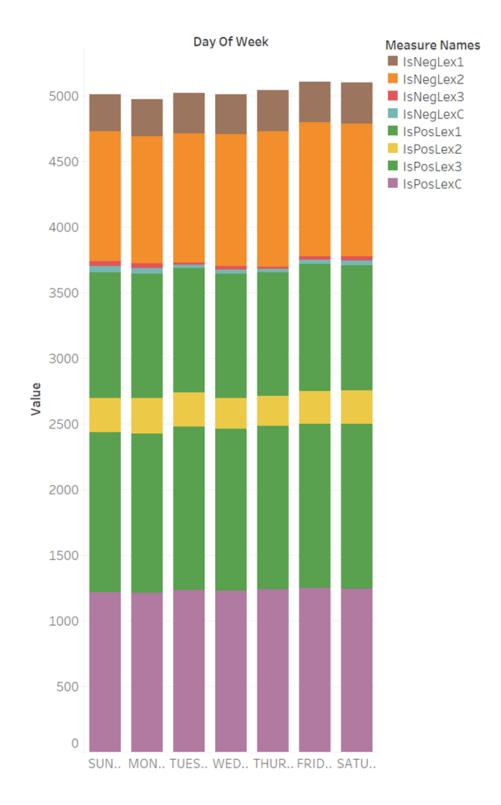


Figure 3: The total count of positive and negative HTML files per day of the week for each Lexicon

Figure 3 shows the count of positive and negative HTML files for every Lexicon for each day of the week. As has been shown in the previous figures, Lexicon 2 is the only Lexicon with a significant number of negative classifications. The visualization also demonstrates that the day of the week makes no impact with the count of positive or negative HTML files, as each day has no significant classification differences.

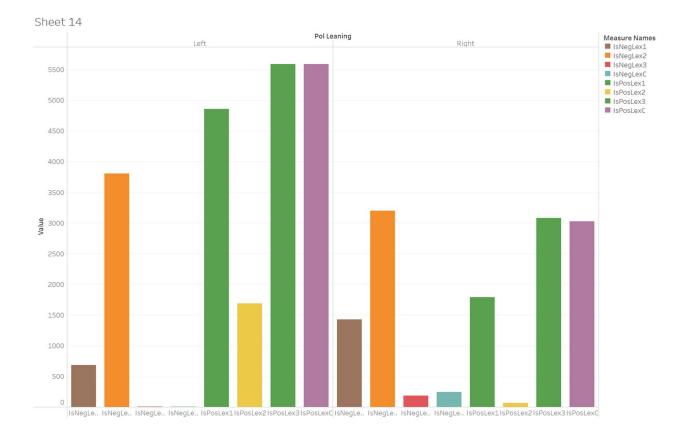


Figure 4: The count of negative and positive classified HTML files by their political leaning, Left or Right

Figure 4 shows that the political leaning of a news source/HTML file makes an impact on the classification of either positive or negative. While the Lexica classify fewer right leaning HTML files, Lexica 1, 3, and Combined have much larger counts of negative files in right leaning files. Due to the underrepresentation of right leaning files in the training data sets, their higher counts may indicate right leaning sources trend more negatively. On the other hand, Lexicon 2 has a larger count of negative files in left leaning files, which would indicate no real difference between the left or right sources because in the training data set there are more left leaning files present.

5. Overall Results

The results of the various learning algorithms and the visualizations demonstrate that the files published year and day of the week make no impact on the classification of the news site's polarity; however, our results gave credence that company name and political leaning may have some predictive power in home page classification.

While OneR with some Lexica used year to predict the classification, those rules produced very low accuracy ratings when used on the testing data set. Furthermore, Figure 3 showed that there were no significant differences between the day of the week and the count of negative and positive files for all four Lexica.

The more advanced machine learning algorithms of J48, SVM, and Random Forest were able to generate models with high amounts of accuracy, normally close to 80% or more. However, these high accuracies were at the cost of classifying all of the test HTML files as negative, which was the dominant classification. This made these high accuracies less desirable. We attempted to create models with lower accuracy, but were able to predict a mix of both positive and negative files. This was done using a cost sensitive evaluator. However, even with the cost sensitive evaluator Lexica 1, 3, and Combined found limited success due to their skewed inclination toward positive classification. Lexicon 2 we found to be the most successful, as even though it would not have the highest accuracy, it was able to predict the best mix of positive and negative files.

6. Conclusion

After running our tests, we feel that we cannot draw any conclusions about our results and apply it to news sources. We believe this is due to a multitude of reasons. Firstly our results were extremely skewed by the massive difference between the human classified testing data set and how the 4 Lexica classified the HTML files. Secondly, the limitations of Achrive.org forces us to look at a new source's home page instead of individual stories.

Most of the human evaluated sources were classified as negative. When the rest of the training data was run through the sentiment program, most turned out to be positive. This means there must be some discrepancy or issue in how the "hand-done" analyses were done. If any work on this project were to continue a more suitable Lexicon would be needed. Our results indicated that Lexicon 2 produces the results most similar to how we classified HTML files. However, it was made of the smallest list of words, indicating that there may be positive words we missed along with Lexicon 2 when classifying the files. Alternatively the extra words present in the other Lexica could have caused the skewed positive classification, which could be inaccurate at least in the context of news headlines.

Archive.org forced us into an abstraction of our initial idea. By looking at only the home page of the news source, we lost the ability to look at an article's author, the genre of the article,

and made the experiment more susceptible to click bait titles. By not being able to capture the author or genre, we could not see if specific news categories or authors disproportionately affect the positive or negative classification. Furthermore, by not looking at the text of the article, clickbait titles may be skewing our data. Click bait titles are often used by authors to try and quickly grab a reader's attention by using an exaggerated title. These titles could heavily skew our data and in the experiment's current iteration we had no way to account for these titles.

Ultimately, while this experiment was unable to produce any definitive results and conclusions, for future projects continuing in this vein, the lessons learned here will be useful in producing better results.

7. Appendix

Java code to: create connection, gather home page text and output to file, and each source's url

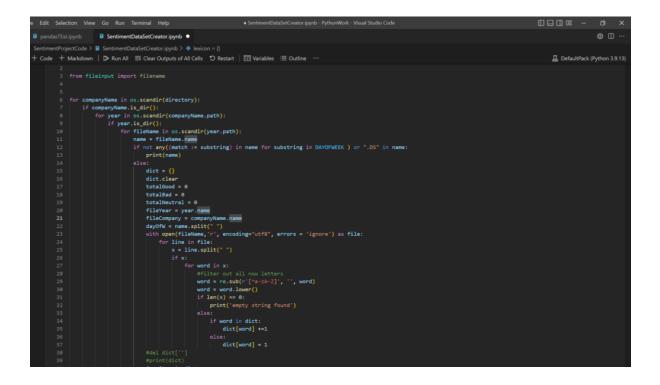
```
public class WebScraper {
    public WebScraper() {
    }
    * gets the html from the input url, sleeps for 5 seconds upon making connection
     * @param url
     * @return
    public HtmlPage getDocument(String url) {
        HtmlPage page = null;
        try {
try (final WebClient webClient = new WebClient(BrowserVersion.FIREFOX)) {
            webClient.getOptions().setUseInsecureSSL(true);
            webClient.getOptions().setCssEnabled(false);
            webClient.getOptions().setJavaScriptEnabled(false);
            try {
                Thread.sleep(5000);
            } catch (InterruptedException e) {
                e.printStackTrace();
            }
            page = webClient.getPage(url);
        } catch (IOException e) {
            e.printStackTrace();
        }
        return page;
        } catch(com.gargoylesoftware.htmlunit.FailingHttpStatusCodeException e) {
        }
        return null;
    }
```

```
public void fetchHomePages() throws IOException {
   DayOfWeek day = LocalDate.parse(this.startDate.toString()).getDayOfWeek();
String[] dateArr;
   String htmlText;
   String incrementedURL;
File file = new File("null file instantiated");
   for (int i = 0; i < 365; i++) {</pre>
        dateArr = getDate(i, day);
file = createNextFile(dateArr, file); // used with file.delete()
        incrementedURL = incrementUrl(dateArr);
        try {
        htmlText = WS.getDocument(incrementedURL).asNormalizedText();
        } catch(NullPointerException n) {
            file.delete();
       file.delete();
        }
        clipHtml(htmlText, dateArr[0]);
        // testing purposes : bw.write(htmlText);
        System.out.println((i + 1) + " files have been created, most recent file: " + dateArr[0]);
    badFiles.close();
```

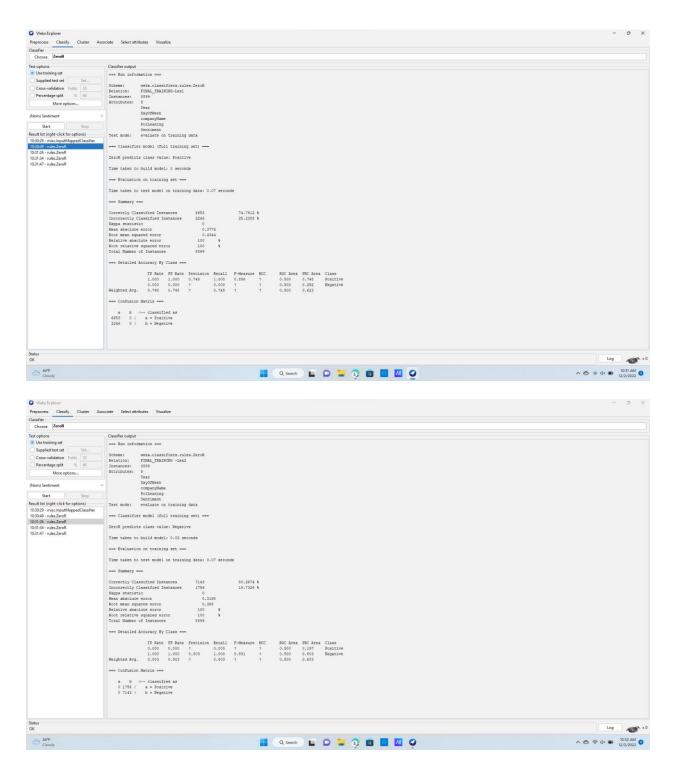
```
public class Abc extends Source {
    // https://web.archive.org/web/20221110033948/https://abcnews.go.com/
    public Abc() {
        super("https://web.archive.org/web/", "033948/https://abcnews.go.com/");
    }
    public Abc(String urlFirstHalf, String urlSecondHalf) {
        super(urlFirstHalf, urlSecondHalf);
    }
    public void fetchHomePages() throws IOException {
        super.fetchHomePages();
    }
}
```

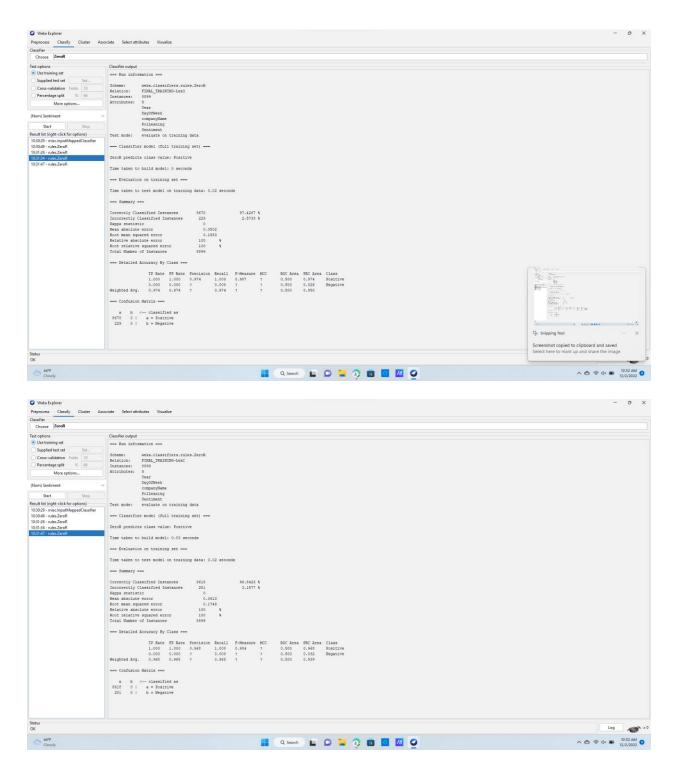
Python Code to classify HTML files as Positive/Negative

	1 import <u>pandas</u> as pd 2 import <u>pumpy</u> as np 3 import os 4 import os 5 import re 6 import re 6 import sys 7 import lo 8 from <u>soupsieve</u> import match	Python
	<pre>1 directory = 'C:/Users/Marc/Desktop/CSC-272/SentimentProject/Data' 2 DAYOPWEEK = {'MONDAY', 'TUESDAY', 'WEDWESDAY', 'THURSDAY', 'FRIDAY', 'SATURDAY', 'SUNDAY'} 3 sentimentDifone = pd:read_esv("C:/Users/Marc/Desktop/CSC-272/SentimentProject/Lexicon/LexiconCombined.csv", encoding='utf-8') 4 SentimentDictone = {} 5 for index in range(0,len(sentimentdfOne)): 6 SentimentDictone[sentimentdfOne['Word'][index]] = sentimentdfOne['Polarity'][index] 7 </pre>	Python
	<pre>1 df = pd.DataFrame({'FileName': pd.Series(dtype = 'str'), 2</pre>	
		Python
	1 The following 4 blocks are used to create the updaded lexcion dataset. The new data set only keeps the word, the type, and the polar	ity of the orginal data set Markdown
	1 The following 4 blocks are used to create the updaded lexcion dataset. The new data set only keeps the word, the type, and the polar	Markdown
D v		
	1 lexicon = []	Markdown Dy Dy 🖯 … 📦
	<pre>1 lexicon = [] 2 lexicondf = pd.DataFrame({'Word' : pd.Series(dtype= 'str'), 'Type' : pd.Series(dtype = 'str'), 'Polarity' : pd.Series(dtype = 'str')</pre>	Markdown № [>+ > ₄ 日 … 音
	<pre>1 lexicon = [] 2 lexicondf = pd.DataFrame({'Word' : pd.Series(dtype= 'str'), 'Type' : pd.Series(dtype = 'str'), 'Polarity' : pd.Series(dtype = 'str')</pre>	Markdown ™⊡ ▷ ₁ ▷ ₂ ⊟ … @
	<pre>1 lexicon = [] 2 lexicondf = pd.DataFrame({'Word' : pd.Series(dtype= 'str'), 'Type' : pd.Series(dtype = 'str'), 'Polarity' : pd.Series(dtype = 'str')</pre>	Markdown № [>+ > ₄ 日 … 音
	<pre>1 lexicon = [] 2 lexicondf = pd.DataFrame(('Word' : pd.Series(dtype= 'str'), 'Type' : pd.Series(dtype = 'str'), 'Polarity' : pd.Series(dtype = 'str') 1 2 with open('C:/Users/Marc/Desktop/CSC-272/SentimentProject/Lexicon/subjclueslen1-HLTEMWLP05.tff') as f: 3 for line in f: 4 lexicon.append(line) 5 print(len(lexicon))</pre>	Markdown № D _k D _k D ··· 8
	<pre>1 lexicon = [] 2 lexicondf = pd.DataFrame(('Word' : pd.Series(dtype= 'str'), 'Type' : pd.Series(dtype = 'str'), 'Polarity' : pd.Series(dtype = 'str') 1 2 with open('C:/Users/Marc/Desktop/CSC-272/SentimentProject/Lexicon/subjclueslen1-HLTEMWLP05.tff') as f: 3 for line in f: 4 lexicon.append(line) 5 print(len(lexicon))</pre>	Markdown № [>+ > ₆ 日 … 會
	<pre>1 lexicon = [] 2 lexicondf = pd.DataFrame(('Word' : pd.Series(dtype= 'str'), 'Type' : pd.Series(dtype = 'str'), 'Polarity' : pd.Series(dtype = 'str') 1 2 with open('C:/Users/Marc/Desktop/CSC-272/SentimentProject/Lexicon/subjclueslen1-HLTEMWLP05.tff') as f: 3 for line in f: 4 lexicon.append(line) 5 print(len(lexicon))</pre>	Markdown № D _k D _k D ··· 8 }})
	<pre>1 lexicon = [] 2 lexicondf = pd.Oat#Frame(('Word' : pd.Series(dtype= 'str'), 'Type' : pd.Series(dtype = 'str'), 'Polarity' : pd.Series(dtype = 'str') 1 2 with open('C:/Users/Marc/Desktop/CSC-272/SentimentProject/Lexicon/subjclueslen1-HLTEMMLP05.tff') as f: 1 4 5 for line in f: 4 5 print(len(lexicon)) 8222 1 for line in lexicon: 2</pre>	Markdown № D _k D _k D ··· 8
	<pre>1 lexicon = [] 2 lexicondf = pd.DataFrame(('Nord' : pd.Series(dtype= 'str'), 'Type' : pd.Series(dtype = 'str'), 'Polarity' : pd.Series(dtype = 'str') 1 2 with open('C:/Users/Marc/Desktop/CSC-272/SentimentProject/Lexicon/subjclueslen1-HLTEMULPOS.tff') as f: 3</pre>	Python
	<pre>1 lexicon = [] 2 lexicondf = pd.DataFrame(('Nord' : pd.Series(dtype= 'str'), 'Type' : pd.Series(dtype = 'str'), 'Polarity' : pd.Series(dtype = 'str') 1 2 with open('C:/Users/Marc/Desktop/CSC-272/SentimentProject/Lexicon/subjclueslen1-HLTEMULPOS.tff') as f: 3</pre>	Markdown № D ₄ D ₄ D ··· }}

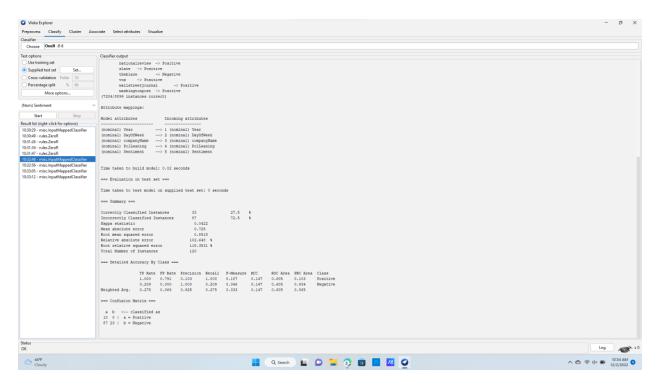


le Edit Selection View	Go Run Termi	nal Help • SentimentDataSetCreator./pynb - PythonWork - Vasual Studio Code		o x
pandasTEst.ipynb	SentimentDataS	etCreator.jpynb 🔹		
SentimentProjectCode >	SentimentDataSel			
+ Code + Markdown	D Run All 🗮	Clear Outputs of All Cells 🙁 Restart / 🖂 Variables 🗮 Outline 🚥	🚊 DefaultPack ((Python 3.9.13)
T Code T Markdown 38 39 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 59		<pre>ClacrOupdot of AlCels 3 Destart [E2] Variables = Outline dict[word] = 1 dict[word] = 1 dict[word] = 1 for item in dict: for item in SentimentDictOne.get(item) if oplarity = Nontileve': totalBad+= dict.get(item) ellf polarity = "positive': totalBad+= dict.get(item) ellf polarity = "positive': totalBad+= dict.get(item) ellf polarity = "positive': totalBad+= dict.get(item) ellf polarity = "both:" totalBad + dict.get(item) elle: totalBad + dict.get(item) sentimentBasult = "Negrive' if totalBad < totalBood: sentimentBasult = "Negrive' sentimentBasult = "Positive' sentimentBasult = "Positive' sentimentBasult = "Positive' sentimentBasult = "Positive' sentimentBasult = "Positive' sentimentBasult = "Fosol' sentimentBasult = "Fos</pre>	E DefaultPack (Python 3.9.13)
60 61 62 63 [1] 1 df.to_csv [31] 2 for key,		<pre>df.loc[len(df.index)] = [name,fileYear,dayof#[0], fileCompany, totalGood,totalBad,totalNeutral,sentimentDiff,sentimentResul print('I finished an itteration', companyName, '', year, '', fileName) //Desktop/CSC-272/SentimentProject/Output/OutputDate.csv*, sep=',', index= False) /tees():</pre>		Python Python
4 E 1				Python

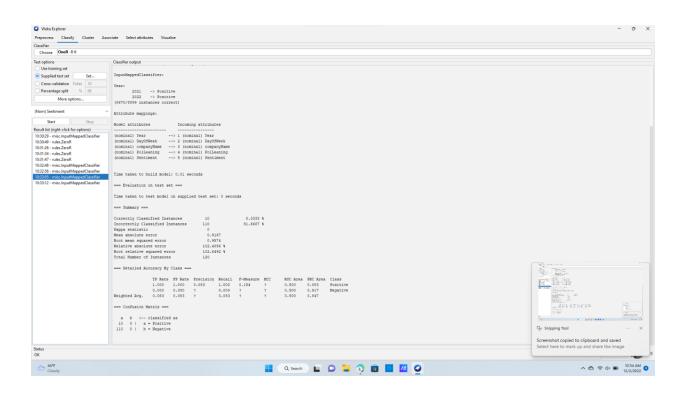




OneR - Lexicon 1 - LexiconC



Weka Explorer		- 0
reprocess Classify Cluster Ass	ciate Select attributes Visualize	
ssifier		
Choose OneR -B 6		
t options	Classifier output	
Use training set	nationalreview -> Negative	
Supplied test set Set	slate -> Hegative	
Cross-validation Folds 10	theblaze -> Negative	
Percentage split % 66	vox -> Positive wellsteetournal -> Positive	
	walabitecjoulial -> Posture washingtonpost -> Negative	
More options	(8113/8899 instances correct)	
iom) Sentiment	Attribute mappings:	
	Notifude mappings.	
Start Stop	Hodel attributes Incoming attributes	
sult list (right-click for options)	(nominal) Year> 1 (nominal) Year	
h30:29 - misc.InputMappedClassifier h30:49 - rules.ZeroR	(nominal) Tear	
h31:26 - rules.ZeroR	(nominal) companyName> 3 (nominal) companyName	
131:34 - rules.ZeroR	(nominal) Polleaning> 4 (nominal) Polleaning	
:31:47 - rules-ZeroR	(nominal) Sentiment> 5 (nominal) Sentiment	
32:48 - misc.InputMappedClassifier		
32:56 - misc.InputMappedClassifier 33:05 - misc.InputMappedClassifier	Time taken to build model: 0.03 seconds	
33:12 - misc.inputMappedClassifier		
	=== Evaluation on test set ===	
	Time taken to test model on supplied test set: 0 seconds	
	ass Summary ass	
	Correctly Classified Instances 100 83.3333 %	
	Incorrectly Classified Instances 20 16.6667 %	
	Kappa statistic 0.333 Wean absolute error 0.1667	
	Nean absolute error 0.1067 Root mean squared error 0.4082	
	Relative absolute error 67.251 %	
	Root relative squared error 136.5397 %	
	Total Number of Instances 120	
	Detailed Accuracy By Class	and the second s
		The second secon
	TP Rate FF Rate Precision Recall F-Measure MCC ROC Area FRC Area Class	
	0.700 0.155 0.292 0.700 0.412 0.977 0.773 0.229 Positive 0.845 0.300 0.969 0.845 0.903 0.377 0.773 0.961 Megative	
	Weighted Avg. 0.853 0.286 0.912 0.833 0.862 0.837 0.773 0.900	to a final second se
		Rangel (2)
	=== Confusion Matrix ===	E 1 3 10 1 2 10 4
	a b < classified as	41 M3-
	7 3 a = Postire	
	17 93 b = Hegative	Snipping Tool ····
		Constant of the debugger of the
		Screenshot copied to clipboard and saved Select here to mark up and share the image
tus		select here to mark up and share the image
Cloudy	📑 🔍 Search 🔛 💭 🧮 🕄 🔟 🚺 🧭	^ @ \$ ¢ ■ ;



Weka Explorer		- 0 X
		0 7
Preprocess Classify Cluster Ass	scate Select attributes Visualize	
Classifier		
Choose OneR -B 6		
Test options	Classifier output	
 Use training set 		
Supplied test set Set	InputMappedClassifier:	
Cross-validation Folds 10		
Percentage split % 66	Year: 2021 -> Positive	
More options	2022 -> Positive	
more options	(9618/8899 instances correct)	
(Nom) Sentiment	Attribute mappings:	
Start Stop	Model attributes Incoming attributes	
Result list (right-click for options)	(nominal) Year> 1 (nominal) Year	
10:30:29 - misc.InputMappedClassifier 10:30:49 - rules.ZeroR	(nominal) Year> 1 (nominal) Year (nominal) Deyöfferek> 2 (nominal) Deyöfferek	
10:31:26 - rules.ZeroR	(nominal) companyName> 3 (nominal) companyName	
10:31:34 - rules.ZeroR	(nominal) PolLeaning> 4 (nominal) PolLeaning	
10:31:47 - rules.ZeroR	(nominal) Sentiment> 5 (nominal) Sentiment	
10:32:48 - misc.InputMappedClassifier		
10:32:56 - misc.InputMappedClassifier 10:33:05 - misc.InputMappedClassifier	Time taken to build model: 0.02 seconds	
10:33:12 - misc.InputMappedClassifier	Evaluation on test pet	
	Time taken to test model on supplied test set: 0 seconds	
	Sumary	
	Correctly Classified Instances 10 8.3333 % Informetry Classified Instances 10 \$1.6667 % Maps statistic 0 \$1.6667 % Mass absolute error 0.5167 Point maximum error 0.5474 Point maximum error 102.566 % Roots relative supared error 102.666 % Tools Instance on Instances 120	
	Detailed Accuracy By Class	2.4
	Best Place Place	
Status		Select here to mark up and share the image
ОК		
Cloudy	📕 (0, Saveh) 🖬 💭 🐂 🧛 🖬 🔟 💋 🥥	∧

J48 Lexicon 1 - Base

=== Summary ===									
Correctly Classi	ances	27		22.6891 %					
Incorrectly Classified Instances			92		77.3109				
Kappa statistic			0.02	87					
Mean absolute er	Mean absolute error			19					
Root mean square			0.74						
Relative absolut			98.84						
Root relative so			103.75	75 %					
Total Number of			119						
Ignored Class Ur	IKNOWN INS	tances		1					
=== Detailed Acc	curacy By	Class ===	:						
	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1.000	0.836	0.089	1.000	0.164	0.121	0.698	0.118	Positive
	0.164	0.000	1.000	0.164	0.281	0.121	0.700	0.950	Negative
Weighted Avg.	0.227	0.063	0.931	0.227	0.272	0.121	0.700	0.887	
=== Confusion Ma	atrix ===								
a b < classified as 9 0 a = Positive 92 18 b = Negative									

J48 Lexicon 1 - Cost Sensitive Evaluator

=== Summary ===									
Correctly Classified Instances			94		78,9916	%			
Incorrectly Clas			25		21.0084	%			
Kappa statistic			0.27	46					
Mean absolute er	ror		0.29	32					
Root mean square			0.44						
Relative absolut			41.29						
Root relative sq			61.06	1 %					
Total Number of			119	1					
Ignored Class Un	KNOWN INS	tances		1					
=== Detailed Acc	uracy By	Class ===							
	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.778	0.209	0.233	0.778	0.359	0.346	0.842	0.450	Positive
	0.791	0.222	0.978	0.791	0.874	0.346	0.840	0.979	Negative
Weighted Avg.	0.790	0.221	0.921	0.790	0.835	0.346	0.841	0.939	
=== Confusion Ma	trix ===								
a b < classified as 7 2 a = Positive 23 87 b = Negative									

J48 Lexicon 2 - Base

=== Summary ===									
Correctly Classified Incorrectly Classifi Kappa statistic Mean absolute error Root mean squared er Relative absolute er Root relative square Total Number of Inst Ignored Class Unknow	101 18 0.3689 0.1634 0.2922 67.2053 % 100.3947 % 119 1		84.8739 % 15.1261 %						
=== Detailed Accurac						DOC I	DD C I	61	
0.7 0.8		0.304 0.979	0.778 0.855	F-Measure 0.438 0.913	MCC 0.423 0.423	ROC Area 0.873 0.869	PRC Area 0.803 0.983	Class Positive Negative	
Weighted Avg. 0.8	849 0.216	0.928	0.849	0.877	0.423	0.869	0.969		
	=== Confusion Matrix ===								
a b < classif 7 2 a = Positi 16 94 b = Negati	ive								

J48 Lexicon 2 - Cost Sensitive Evaluator

=== Summary ===									
Correctly Class: Incorrectly Class Kappa statistic Mean absolute en Root mean square Relative absolut Root relative so Total Number of Ignored Class Un	117 2 0.8661 0.0409 0.1383 16.8066 % 47.499 % 119 1		98.3193 % 1.6807 %						
Weighted Avg. === Confusion Ma a b < c 7 2 a =	TP Rate 0.778 1.000 0.983	FP Rate 0.000 0.222 0.205	Precision 1.000 0.982 0.983	Recall 0.778 1.000 0.983	F-Measure 0.875 0.991 0.982	MCC 0.874 0.874 0.874	ROC Area 0.873 0.869 0.869	PRC Area 0.803 0.983 0.969	Class Positive Negative

J48 Lexicon 3 - Base

-

=== Summary ===									
Correctly Classi	ified Inst	ances	9		7.563	%			
Incorrectly Clas	ssified In	stances	110		92.437	%			
Kappa statistic			0						
Mean absolute error			0.90	25					
Root mean squared error			0.93	67					
Relative absolute error			100.01	%					
Root relative squared error			100.01	09 %					
Total Number of	•		119						
Ignored Class Ur	nknown Ins	tances		1					
J · · · · · · · ·									
=== Detailed Acc	curacy By	Class ===							
	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1.000	1.000	0.076	1.000	0.141	?	0.500	0.075	Positive
	0.000	0.000	?	0.000	?	?	0.500	0.917	Negative
Weighted Avg.	0.076	0.076	?	0.076	?	?	0.500	0.853	5
5									
=== Confusion Ma	atrix ===								

a b <-- classified as 9 0 | a = Positive 110 0 | b = Negative

J48 Lexicon 3 - Cost Sensitive Evaluator

=== Summary ===					
Correctly Classified Instances Incorrectly Classified Instance Kappa statistic Mean absolute error Root mean squared error Relative absolute error Root relative squared error Total Number of Instances Ignored Class Unknown Instances === Detailed Accuracy By Class	0.0072 0.8205 0.8675 90.9206 % 92.624 % 119 1	11.7647 % 88.2353 %			
TP Rate FP Ra 1.000 0.955 0.045 0.000 Weighted Avg. 0.118 0.072 === Confusion Matrix ===	te Precision Recall	F-Measure MCC 0.146 0.060 0.087 0.060 0.091 0.060	ROC Area 0.518 0.518 0.518 0.518	PRC Area 0.094 0.924 0.861	Class Positive Negative
a b < classified as 9 0 a = Positive 105 5 b = Negative					

J48 Lexicon Combined - Base

=== Summary ===									
Correctly Classi	Correctly Classified Instances				7.563	%			
Incorrectly Classified Instances			110		92.437	%			
Kappa statistic			0						
Mean absolute error			0.89	76					
Root mean square	d error		0.93	11					
Relative absolute error			100.01	%					
Root relative sq	uared err	or	100.01	.08 %					
Total Number of	Instances		119						
Ignored Class Un	known Ins	tances		1					
=== Detailed Acc	uracy By	Class ===							
	TP Rate	FP Rate	Precision	Recall	F-Measure	мсс	ROC Area	PRC Area	Class
	1.000	1.000	0.076	1.000	0.141	?	0.500	0.075	Positive
	0.000	0.000	?	0.000	?	?	0.500	0.917	Negative
Weighted Avg.	0.076	0.076	?	0.076	?	?	0.500	0.853	5
=== Confusion Ma	trix ===								

a b <-- classified as 9 0 | a = Positive 110 0 | b = Negative

J48 Lexicon Combined - Cost Sensitive Evaluator

=== Summary ===									
Correctly Classified Instances Incorrectly Classified Instances Kappa statistic Mean absolute error Root mean squared error Relative absolute error Root relative squared error Total Number of Instances Ignored Class Unknown Instances			14 105 0.00 0.82 0.87 91.74 93.54 119	34 1 71 %	11.7647 88.2353				
=== Detailed Accu	iracy By (Class ===							
	TP Rate 1.000 0.045 0.118	FP Rate 0.955 0.000 0.072	Precision 0.079 1.000 0.930	Recall 1.000 0.045 0.118	F-Measure 0.146 0.087 0.091	MCC 0.060 0.060 0.060	ROC Area 0.518 0.518 0.518	PRC Area 0.094 0.924 0.861	Class Positive Negative
=== Confusion Mat	rix ===								
9 0 a =	lassified Positive Negative	as							

SVM Lexicon 1 - Base

=== Summary ===									
Correctly Classi Incorrectly Clas Kappa statistic			33 86 0.04	.05	27.7311 72.2689				
Mean absolute er	ror		0.72						
Root mean square	d error		0.85	01					
Relative absolut	e error		101.77	13 %					
Root relative sq			117.73	14 %					
Total Number of			119						
Ignored Class Un	known Ins	tances		1					
=== Detailed Acc	uracy By	Class ===							
	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1.000	0.782	0.095	1.000	0.173	0.144	0.608	0.094	Positive
	0.218	0.000	1.000	0.218	0.358	0.144	0.609	0.935	Negative
Weighted Avg.	0.277	0.059	0.932	0.277	0.344	0.144	0.609	0.871	
=== Confusion Ma	trix ===								
a b < cla	ssified a	s							
9 $0 \mid a = Po$		-							

9 0 | a = Positive 86 24 | b = Negative

SVM Lexicon 1 - Cost Sensitive Evaluator

=== Summary ===								
Correctly Classified Incorrectly Classifie Kappa statistic Mean absolute error Root mean squared err Relative absolute err Root relative squared Total Number of Insta Ignored Class Unknown === Detailed Accuracy	Instances r r error ces Instances	94 25 0.27 0.21 29.58 63.47 119	101 583 347 %	78.9916 21.0084				
TP R 0.77 0.79 Weighted Avg. 0.79 === Confusion Matrix a b < classifi 7 2 a = Positiv 23 87 b = Negativ	0.209 0.222 0.221 == d as	Precision 0.233 0.978 0.921	Recall 0.778 0.791 0.790	F-Measure 0.359 0.874 0.835	MCC 0.346 0.346 0.346	ROC Area 0.781 0.795 0.794	PRC Area 0.192 0.965 0.906	Class Positive Negative

SVM Lexicon 2 - Base

=== Summary ===									
Correctly Classi	ified Inst	ances	101		84.8739	%			
Incorrectly Clas	sified Ir	stances	18		15.1261	%			
Kappa statistic			0.36	89					
Mean absolute er	ror		0.15	13					
Root mean square	ed error		0.38	89					
Relative absolut	e error		62.20	46 %					
Root relative so	quared err	or	133.60	64 %					
Total Number of	•		119						
Ignored Class Ur	nknown Ins	tances		1					
=== Detailed Acc	curacy By	Class ===	:						
	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.855	0.222	0.979	0.855	0.913	0.423	0.777	0.961	Negative
	0.778	0.145	0.304	0.778	0.438	0.423	0.817	0.253	Positive
Weighted Avg.	0.849	0.216	0.928	0.849	0.877	0.423	0.780	0.908	
=== Confusion Ma									
a h < cla	assified a	IS							

a b <--- classified as 94 16 | a = Negative 2 7 | b = Positive

SVM Lexicon 2 - Cost Sensitive Evaluator

=== Summary ===									
Correctly Classi	fied Inst	ances	117		98.3193	%			
Incorrectly Clas	sified In	stances	2		1.6807	%			
Kappa statistic			0.86	61					
Mean absolute er	ror		0.01	.68					
Root mean square	d error		0.12	96					
Relative absolut	e error		6.91	16 %					
Root relative so	uared err	or	44.53	55 %					
Total Number of			119						
Ignored Class Un	known Ins	tances		1					
-									
=== Detailed Acc	uracy By	Class ===	:						
	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0.778	0.000	1.000	0.778	0.875	0.874	0.889	0.794	Positive
	1.000	0.222	0.982	1.000	0.991	0.874	0.850	0.973	Negative
Weighted Avg.	0.983	0.205	0.983	0.983	0.982	0.874	0.853	0.960	-
=== Confusion Ma	trix ===								
a b < c	lassified	as							
7 2 3 -	- Docitivo								

7 2 | a = Positive 0 110 | b = Negative

SVM Lexicon 3 - Base

=== Summary ===									
Correctly Classi	ified Inst	ances	9		7,563	%			
Incorrectly Clas	sified In	stances	110		92.437	%			
Kappa statistic			0						
Mean absolute er	ror		0.92	44					
Root mean square	ed error		0.96	514					
Relative absolut	e error		102.43	02 %					
Root relative so	quared err	or	102.64	96 %					
Total Number of	Instances	;	119						
Ignored Class Ur	nknown Ins	tances		1					
=== Detailed Acc	curacy By	Class ===							
	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	1.000	1.000	0.076	1.000	0.141	?	0.500	0.075	Positive
	0.000	0.000	?	0.000	?	?	0.500	0.917	Negative
Weighted Avg.	0.076	0.076	?	0.076	?	?	0.500	0.853	5
5 5									
=== Confusion Ma	atrix ===								

a b <-- classified as 9 0 | a = Positive 110 0 | b = Negative

=== Summary ===									
Correctly Classi Incorrectly Class Kappa statistic Mean absolute er	sified In		110 9 0 0.07	56	92.437 7.563				
Root mean square Relative absolut Root relative so	ed error ce error quared err		0.27 8.38 29.36	5 07 %					
Total Number of Ignored Class Ur === Detailed Acc	nknown Ins	tances	119	1					
	0.000	FP Rate 0.000 1.000	Precision ? 0.924	Recall 0.000 1.000	F-Measure ? 0.961	MCC ? ?	ROC Area 0.500 0.500	PRC Area 0.075 0.917	Class Positive Negative
Weighted Avg.	0.924	0.924	?	0.924	?	?	0.500	0.853	Negative
=== Confusion Ma	atrix ===								
09 a=	lassified Positive Negative	1							

SVM Lexicon 3 - Cost Sensitive Evaluator No False Negative Weights

SVM Lexicon 3 - Cost Sensitive Evaluator

9 0 | 105 5 |

a = Positive
b = Negative

=== Summary ===									
Correctly Classi	fied Inst	ances	14		11.7647	%			
Incorrectly Clas	sified In	stances	105		88.2353	%			
Kappa statistic			0.00	72					
Mean absolute er	ror		0.88	24					
Root mean square	ed error		0.93	93					
Relative absolut	e error		97.77	43 %					
Root relative sq	uared err	or	100.28	95 %					
Total Number of			119						
Ignored Class Un	iknown Ins	tances		1					
=== Detailed Acc	curacy By	Class ===	:						
	TP Rate	FP Rate	Precision	Recall	F-Measure	мсс	ROC Area	PRC Area	Class
	1.000	0.955	0.079	1.000	0.146	0.060	0.523	0.078	Positive
	0.045	0.000	1.000	0.045	0.087	0.060	0.523	0.920	Negative
Weighted Avg.	0.118	0.072	0.930	0.118	0.091	0.060	0.523	0.857	5
=== Confusion Ma	ntrix ===								
a b < c	lassified	as							

SVM Lexicon Combined - Base

=== Summary ===							
Correctly Classified Insta Incorrectly Classified Ins Kappa statistic Mean absolute error Root mean squared error Relative absolute error Root relative squared erro Total Number of Instances Ignored Class Unknown Inst === Detailed Accuracy By C	tances r ances	9 110 0.9244 0.9614 102.9961 % 103.2673 % 119 1	7.563 92.437				
1.000 0.000	1.000 0.07 0.000 ? 0.076 ?	ision Recall 6 1.000 0.000 0.076	F-Measure 0.141 ? ?	MCC ? ?	ROC Area 0.500 0.500 0.500	PRC Area 0.075 0.917 0.853	Class Positive Negative

SVM Lexicon Combined - Cost Sensitive Evaluator

=== Summary ===									
Correctly Classi Incorrectly Class Kappa statistic Mean absolute er Root mean square Relative absolut Root relative sq Total Number of	sified In ror d error e error uared err Instances	stances	14 105 0.00 0.88 0.93 98.31 100.89 119	24 93 45 %	11.7647 88.2353				
Ignored Class Un				1					
				De ee 11	F Maaaaaaa	MCC			<u>()</u>
Weighted Avg.	TP Rate 1.000 0.045 0.118	FP Rate 0.955 0.000 0.072	Precision 0.079 1.000 0.930	Recall 1.000 0.045 0.118	F-Measure 0.146 0.087 0.091	MCC 0.060 0.060 0.060	ROC Area 0.523 0.523 0.523	PRC Area 0.078 0.920 0.857	Class Positive Negative
=== Confusion Ma		0.072	0.950	0.110	0.091	0.000	0.323	0.057	
a b < c	lassified	as							

9 0 | a = Positive 105 5 | b = Negative

reprocess Classify Cluster Asso assifier	odate Select attributes Visualize
	-num-slots1-K0-M1.0-V0.001-S1
est options	Classifier output
Use training set	Bagging with 100 iterations and base learner
Supplied test set Set	pagging with 100 lefations and base learner
Cross-validation Folds 10	weka.classifiers.trees.RandomTree -K 0 -M 1.0 -V 0.001 -S 1 -do-not-check-capabilities
Percentage split % 66	Attribute mappings:
More options	
More options	Model attributes Incoming attributes
Nom) Sentiment	
ion, senanen	(nominal) DayOfWeek> 2 (nominal) DayOfWeek
Start Stop	(nominal) companyName> 3 (nominal) companyName
esult list (right-click for options)	(nominal) PolLeaning> 4 (nominal) PolLeaning
0:21:13 - misc.InputMappedClassifier	(nominal) Sentiment> 5 (nominal) Sentiment
0:22:30 - misc.InputMappedClassifier	
0:27:55 - misc.InputMappedClassifier	Time taken to build model: 0.15 seconds
0:29:10 - misc.InputMappedClassifier	
0:31:46 - misc.InputMappedClassifier	=== Evaluation on test set ===
0:39:55 - misc.InputMappedClassifier	Time taken to test model on supplied test set: 0.01 seconds
	Time taken to test model on supplied test set: 0.01 seconds
	=== Summary ===
	Correctly Classified Instances 28 23.3333 9
	Incorrectly Classified Instances 92 76.6667 % Kappa statistic 0.0316
	Mean absolute error 0.6698
	Root mean squared error 0.7486
	Relative absolute error 97.6637 %
	Root relative squared error 104.0516 %
	Total Number of Instances 120
	=== Detailed Accuracy By Class ===
	TF Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class
	1.000 0.836 0.098 1.000 0.179 0.127 0.821 0.326 Positive
	Weighted Avg. 0.164 0.000 0.1064 0.281 0.127 0.821 0.972 Negative
	Weighted Avg. 0.233 0.070 0.923 0.233 0.273 0.127 0.021 0.910
	=== Confusion Matrix ===
	a b < classified as 10 0 a = Positive
	$10 \ 0 \ \ a = \text{rostive}$ $52 \ 18 \ \ b = \text{Negative}$
	be to i the mediante
atus	
K	Log

Random Forest Lexicon 1 - Base

Preprocess Classify Cluster Assoc Classifier	iate Select attributes Visualize	
Choose RandomForest -P 100 -I 100 -	num-slots 1 - K 0 - M 1.0 - V 0.001 - S 1	
Test options Use training set Supplied test set Ocross-validation Percentage split % 66 More options	Cassifer output 0 3 30 0 Attribute mappings: Model attributes Incoming attributes (nominal) Year > 1 (nominal) Year (nominal) You O'Greek -> 2 (nominal) DayO'Breek	
Start Stop Result list (right-click for options) 10:21:13 - misc.lnputMappedClassifier 10:22:30 - misc.lnputMappedClassifier 10:27:55 - misc.lnputMappedClassifier 10:29:10 - misc.lnputMappedClassifier	<pre>(nominal) companyName> 3 (nominal) companyName (nominal) Polleaning> 4 (nominal) Polleaning (nominal) Sentiment> 5 (nominal) Sentiment Time taken to build model: 0.01 seconds</pre>	
10:31:46 - misc.InputMappedClassifier 10:39:55 - misc.InputMappedClassifier	<pre>**** Evaluation on test set **** Time taken to test model on supplied test set: 0 seconds ***** Summary **** Correctly Classified Instances 110 \$1.6667 % Incorrectly Classified Instances 10 \$.3333 % Kappa statistic 0 Mean absolute error 0.2738 Root relative absolute error 30.7684 % Root relative squared error 43.3994 % Total Number of Instances 120 **** Detailed Accuracy By Class ****</pre>	
	TP Fate FP Fate Precision Recall F-Measure NCC ROC Area PCL Area CLass 0.000 0.000 ? 0.000 ? 0.000 0.003 Positive 1.000 0.917 1.000 0.957 ? 0.500 0.083 Positive Weighted Avg. 0.917 0.917 ? 0.917 ? 0.500 0.847 ==== Confluion Matrix === > Negative a b < classified as	
Status OK		Log x0
	📕 Q 🔎 💿 🦁 🐂 📕 🖼 📾 🔗 🤗 💿 🧟 🛤	^

Random Forest Lexicon 1 - Cost Sensitive Evaluator

Classifier Choose RandomForest -P 100 -I 100	-num-slots 1 - K0 - M 1.0 - V 0.001 - S 1
est options	(Classifier output
Use training set	
Supplied test set Set	Bagging with 100 iterations and base learner
Cross-validation Folds 10	weka.classifiers.trees.RandomTree -K 0 -M 1.0 -V 0.001 -S 1 -do-not-check-capabilities
	Attribute mappings:
· ·	
More options	Model attributes Incoming attributes
	(nominal) Year> 1 (nominal) Year
Nom) Sentiment	(nominal) real 1 (nominal) real (nominal) DayOfWeek
Start Stop	(nominal) companyName> 1 (ioninal) companyName
Result list (right-click for options)	(nominal) PolLeaning> 4 (nominal) PolLeaning
10:21:13 - misc.InputMappedClassifier	(nominal) Sentiment> 5 (nominal) Sentiment
10:22:30 - misc.inputMappedClassifier	
10:27:55 - misc.InputMappedClassifier	Time taken to build model: 0.13 seconds
10:29:10 - misc.InputMappedClassifier	Time taken to build model: 0.13 seconds
10:31:46 - misc.InputMappedClassifier	=== Evaluation on test set ===
10:39:55 - misc.InputMappedClassifier	
	Time taken to test model on supplied test set: 0.01 seconds
	=== Summary ===
	Correctly Classified Instances 101 84.1667 %
	Incorrectly Classified Instances 19 15.8333 %
	Kappa statistic 0.3496
	Mean absolute error 0.1665
	Root mean squared error 0.3003
	Relative absolute error 67.1604 % Root relative squared error 100.4237 %
	Not relative squared error 100.4237 * Total Number of Instances 120
	=== Detailed Accuracy By Class ===
	TP Rate FP Rate Precision Recall F-Measure MCC ROC Area FRC Area Class
	0.700 0.145 0.304 0.700 0.424 0.389 0.811 0.742 Positive 0.855 0.300 0.969 0.855 0.500 0.389 0.811 0.963 Negative
	0.635 0.300 0.767 0.534 0.570 0.553 0.700 0.359 0.611 0.765 megative Weighted Avg. 0.842 0.287 0.914 0.842 0.866 0.389 0.811 0.945
	=== Confusion Matrix ===
	a b < classified as 7 3 a = Positive
	/ 3 a = FOSILIVE 16 94 b = Negative
	AV 21 D = regulate
itatus	
)K	Log

Random Forest Lexicon 2 - Base

Classifier Choose CostSensitiveClassifier -cos	st-matrix "(0.0 1.0; 15.0 0.0)" - S1 - W weka dassifiers rules ZeroR	
Test options	Classifier output	
O Use training set	0 1	
Supplied test set Set	15 0	
Cross-validation Folds 10	Attribute mappings:	
Percentage split % 66		
More options	Model attributes Incoming attributes	
	(nominal) Year> 1 (nominal) Year	
(Nom) Sentiment	(nominal) DayOfweek> 2 (nominal) DayOfweek	
Start Stop	(nominal) companyName> 3 (nominal) companyName	
Result list (right-click for options)	(nominal) PolLeaning> 4 (nominal) PolLeaning (nominal) Sentiment> 5 (nominal) Sentiment	
10:21:13 - misc.InputMappedClassifier	(nominal) Sentiment> 3 (nominal) Sentiment	
10:22:30 - misc.InputMappedClassifier		
10:27:55 - misc.InputMappedClassifier	Time taken to build model: 0 seconds	
10:29:10 - misc.InputMappedClassifier	=== Evaluation on test set ===	
10:31:46 - misc.InputMappedClassifier 10:39:55 - misc.InputMappedClassifier		
10:33:32 - misc.inputWappedClassifier	Time taken to test model on supplied test set: 0 seconds	
10:45:01 - misc.InputMappedClassifier	Sumary	
	Correctly Classified Instances 110 91.6667 %	
	Incorrectly Classified Instances 10 8.3333 % Kappa statistic 0	
	Kappa statistic 0 Mean absolute error 0.0969	
	Root mean squared error 0.2844	
	Relative absolute error 39.0841 %	
	Root relative squared error 95.123 %	
	Total Number of Instances 120	
	=== Detailed Accuracy By Class ===	
	TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class 0.000 0.000 2 0.000 ? 0.000 ? 2 0.500 0.083 Positive	
	1.000 1.000 r 0.000 r r 0.500 0.003 POSITIVE	
	Weighted Avg. 0.917 0.917 7 0.917 7 7 0.500 0.847	
	=== Confusion Matrix ===	
	a b < classified as	
	0 10 a = Positive	
	0 110 (b = Negative	
Status		
OK		Log x0

Random Forest Lexicon 2 - Cost Sensitive Evaluator

	C:\\Program Hiles\\Weka-3-	8-6" -S 1 -W wek	.dassifiers.trees.Ra	andomForest ·	P 100 -I 100 -	num-slots 1	-K 0 -M 1.0 -V	√0.001 -S 1	
ons	Classifier output								
aining set ied test set Set	0 150								
s-validation Folds 10	Attribute mapping	16.							
tage split % 66			coming opposite						
More options	Model attributes		coming attrib						
ntiment ~	(nominal) Year (nominal) DayOfWe	eek> 2	(nominal) Yea (nominal) Day	ofWeek					
tart Stop	(nominal) company (nominal) PolLear		(nominal) com (nominal) Pol						
(right-click for options) - meta.CostSensitiveClassifier	(nominal) Sentime		(nominal) Ser						
3 - misc.InputMappedClassifier	Time taken to bui	14							
51 - misc.InputMappedClassifier 18 - misc.InputMappedClassifier									
0 - misc.InputMappedClassifier 2 - misc.InputMappedClassifier	=== Evaluation on test set ===								
 misc.InputMappedClassifier misc.InputMappedClassifier 	Time taken to tes	st model on su	upplied test s	set: 0.01 s	seconds				
 misc.InputMappedClassifier 	=== Summary ===								
2 - misc.InputMappedClassifier	Correctly Classif Incorrectly Class				35.8333 64.1667				
	Kappa statistic		0.	0274	04.1007	•			
	Mean absolute err Root mean squared	d error	0.	.607 .6832					
	Relative absolute Root relative squ	ared error	228.	.9459 % .4851 %					
	Total Number of I	Instances	120						
	=== Detailed Accu	aracy By Class							
			Rate Precisio					PRC Area	
		0.800 0.60		0.800		0.071	0.803	0.741	Positive Negative
		0.358 0.24		0.358		0.071	0.803	0.942	
	=== Confusion Mat	trix ===							
	a b < clas								
	8 2 a = Pos 75 35 b = Neg								
Partly cloudy	•	Q 🗖	<u> </u>		-	*	8	<u>e</u> (2 2
Partly cloudy ka Explorer cess <u>Classify</u> Cluster Asso	ociate Select attributes		D			₽	9	<u>e</u> (2 2
artly cloudy a Explorer sss <u>Classify</u> Cluster Asso	ociate Select attributes	Visualize			•		•••	-	2
Partly cloudy ka Explorer ess <u>Classify</u> Cluster Asso recommended the classifier -N "G ions	cciate Select attributes C\\Program Files\\Weka-3- Classifier output	Visualize			•		•••	-	2
Rattly cloudy ka Explorer r sses <u>Classify</u> Cluster Asso r sses <u>CostSensitiveClassifier -N 'G</u> ions raining set	Classifier output	Visualize			•		•••	-	2
Rantly cloudy ka Explorer cess Classify Cluster Asse r cse CostSensitiveClassifier -N "6 tions training set uplied test set <u>Set.</u>	Clubrogram Files/Weka-3- Classifier output 0 150 5 0	Visualize 8-6° -S 1 -W wekt			•		•••	-	2
Rattly cloudy Rattly cloudy Ratsly cluster Ress Cassify Cluster Asso r training set plied test set Set Plied test set Set 10	Classifier output	Visualize 8-6° -S 1 -W wekt			•		•••	-	
Partly clously ka Explorer cess Cluster Asso r cos CostSensitiveClassifier -N " tions training set oplied test set Set svelidation Folds 10	Clubrogram Files/Weka-3- Classifier output 0 150 5 0	Visualize 8-6° -S 1 -W weki 76 :		andomForest -	•		•••	-	2
Classify Cluster Asso ier costSensitiveClassifier -N 'G ptions se training set ptions set raining set projend test set Set	Clubrogram Files/\Weka-3- Clubrogram Files/\Weka-3- Clubrogram Files/\Weka-3- 0 150 5 0 Attribute mapping Model attributes (nominal) Year	Visualize 8-6° -S 1 -W wekt 15: 	.dassifiers trees.Ra accoming attrib (nominal) Yee	andomForest -	•		•••	-	
Partly cloudy Reaction of the second secon	Close Select attributes Close Select attributes Close Select Attributes 0 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Year	Visualize &-6° - S 1 -W wekt ps: 	.dassifiers.trees.Re ncoming attrib (nominal) Yea (nominal) Com	andomForest 	•		•••	-	
Partly cloudy Peka Explorer Classify Cluster Asso Rer Coces CatSensitiveClassifier -N ' ptions te training set training set Coces-validation Folds 10 Coces-validation Folds 10 Coces-validation Folds 10 Coces-validation Set training text Stop Start Stop	CliProgram Files/Weka-3- CliProgram Files/Weka-3- Cliprogram Files/Weka-3- Cliprogram Files/Weka-3- 0 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Year	Visualize 8-6" - S 1 -W week 75 : 	.dassifiers trees Ra ncoming attrib (nominal) Yea (nominal) Day	andomForest - putes us yoffwek spanyName Leaning	•		•••	-	
Party cloudy Weka Explorer vrocess Classify Cluster Asse iffer hoose CostSensitiveClassifier -N ''Co options Jese training set Supplied test set Set_ Cross-validation Folds 10 Percentage split % 66 More options n) Sentiment	Closifier output Closifier output 0 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Follear (nominal) Sentime	Visualize 8-5' - 5 1 - W weld 25 : 	. dessifiers trees Ra nooming attrib (nominal) Yee (nominal) Day (nominal) Pol (nominal) Sen	andomForest - putes us yoffwek spanyName Leaning	•		•••	-	
	CliProgram Files/\Weka-3- CliProgram Files/\Weka-3- CliProgram Files/\Weka-3- CliProgram Files/\Weka-3- CliProgram Files/\Weka-3- CliProgram Files/\Weka-3- S 0 S 0 Attribute mapping Model attributes (nominal) DevofWe (nominal) DevofWe (nominal) DevofWe	Visualize 8-5' - 5 1 - W weld 25 : 	. dessifiers trees Ra nooming attrib (nominal) Yee (nominal) Day (nominal) Pol (nominal) Sen	andomForest - putes us yoffwek spanyName Leaning	•		•••	-	2
Rettly cloudy Rettly cloudy Retsly cloudy Ress Cassify Cluster Asso r r css CastSensitiveClassifier -N 'r tions r training set polet test set Set servalidation Folds 10 G6 More options Start Stop at trjiht-tdick for options) Rettly Classifier 1 - mici.ngutMappedClassifier 8 - mici.ngutMappedClassifier 8 - mici.ngutMappedClassifier 8 - mici.ngutMappedClassifier	Closifier output Closifier output 0 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Follear (nominal) Sentime	Visualize 8-5" - 5 1 - W weiki 25: > 1 rek> 2 Name> 3 ning> 4 nnt> 5 L1d model: 0.2		andomForest - putes us yoffwek spanyName Leaning	•		•••	-	2
Antity cloudy An	Classifier output Classifier output 0 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Sentime (nominal) Sentime Time taken to buj	Visualize 8-5' - 5 1 - W weke 25: > 1 rek> 2 Name> 3 ning> 4 nnt> 5 L1 model: 0	coming attrib (nominal) Yee (nominal) Day (nominal) Pol (nominal) Ser seconds	andomForest 			•••	-	2
Partly cloudy Partly cloudy Las Explorer ess Cassify Cluster Asse ses CostSensitiveClassifier -N "C fors training set plied test set Set s-validation Folds 10 entage split % 66 More options entiment Start Stop t (right-click for options) - misc.inputMappedClassifier - mi	Classifier output Classifier output 0 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Sentime (nominal) Sentime Time taken to bui === Evaluation or	Visualize 8-5' - 5 1 - W weke 25: > 1 rek> 2 Name> 3 ning> 4 nnt> 5 L1 model: 0	coming attrib (nominal) Yee (nominal) Day (nominal) Pol (nominal) Ser seconds	andomForest 			•••	-	2
Partly cloudy Weka Explorer Torooss Calssify Cluster Asso filer Torooss CostSensitiveClassifier -N 'r vptions te training set upplied test set Set. Toss-validation Folds 10 Wore options. Wore options. Wore options. Start Stop tit (right-filek for options) EG8 - macLogNampedClassifier 218 - misc.InputMappedClassifier 219 - misc.InputMappedClassifier 210 - misc.InputMappedClassifier 210 - misc.InputMappedClassifier 212 - misc.InputMappedClassifier 223 - misc.InputMappedClassifier 224 - misc.InputMappedClassifier 225 - misc.InputMappedClassifier 226 - misc.InputMappedClassifier 227 - misc.InputMappedClassifier 227 - misc.InputMappedClassifier 228 - misc.InputMappedClassifier 229 - misc.InputMappedClassifier 230 - misc.InputMappedClassifier 231 - misc.InputMappedClassifier 231 - misc.InputMappedClassifier 232 - misc.InputMappedClassifier 233 - misc.InputMappedClassifier 234 - misc.InputMappedClassifier 235 - misc.InputMappedClassifier 236 - misc.InputMappedClassifier 237 - misc.InputMappedClassifier 238 - misc.InputMappedClassifier 239 - misc.InputMappedClassifier 230 - misc.InputMappedClassifier 230 - misc.InputMappedClassifier 231 - mis	Clustifier output O 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Year (nominal) Sentime Time taken to buil === Evaluation or Time taken to tes === Summary === Correctly Classifi	Visualize 25: 75: 75: 75: 75: 75: 75: 75: 7	coming attrib nooming attrib (nominal) Yea (nominal) Pol (nominal) Pol (nominal) Ser s seconds s 50	andomForest a butes 		num-slots 1	•••	-	
Partly cloudy Weka Explorer rocoss Classify Cluster Asse filter rocose CostSensitiveClassifier - N 'r opptions se training set upplied test set Set_ rocos-validation Folds 10 wreentage split % 66 More options Nome options N Sentiment Statt Stop t lat (right-cluck for options)	Classifier output Classifier output Classifier output 0 150 0 150 Attribute mapping Model attributes (nominal) Year (nominal) Sentime Time taken to bui === Evaluation or Time taken to tes === Summary ===	Visualize 8-5'-51-W wek 75: II Teck> 2 Vitame> 3 ling> 4 nnt> 5 lild model: 0.1 a test set	coming attrib nooming attrib (nominal) Yea (nominal) Pol (nominal) Pol (nominal) Ser s seconds s 50	andomForest a butes 		num-slots 1	•••	-	
Partly cloudy Veka Explorer Classify Cluster Asso ier CostSensitiveClassifier -N * ptions te training set ppilot est set Set. Oss-validation Folds Cost-validation Cost-	Classifier output Classifier output Classifier output 0 150 0 150 Attribute mapping Model attributes (nominal) rear (nominal) company (nominal) company (nominal) sentime Time taken to buil === Evaluation or Time taken to tes === Summary === Correctly Classifi Kappa statistic	Visualize 8-5'-51-W weke 75: II Teck> 2 Vitame> 3 ling> 4 nnt> 5 lild model: 0.1 a test set ==== at model on su field Instance; ifield Instance; ifie		andomForest - 		num-slots 1	•••	-	
Partly cloudy Veka Explorer Classify Cluster Asso ier CostSensitiveClassifier -N * ptions te training set ppilot est set Set. Oss-validation Folds Cost-validation Cost-	Classifier output Classifier output Classifier output 0 150 0 150 Attribute mapping Model attributes (nominal) rear (nominal) sentime Time taken to buil === Evaluation or Time taken to tes === Summary === Correctly Classifi Kappa statistic Kappa statistic Real time absolute err Root mean squaree	Visualize 8-6'-S1-W weak 75: II Teck -> 2 Vitama -> 3 Vitama ->		andomForest- nuttes 		num-slots 1	•••	-	
Perthy clously ka Explorer cess Classify Cluster Asse of costSensitiveClassifier -N ** tions training set polied test set Set_ss-validation Folds 10 Ge6 Germape split % 66 Germanuel CostSensitiveClassifier 38 - mic.inputMappedClassifier 38 - mic.inputMappedClassifier 39 - mic.inputMappedClassifier 39 - mic.inputMappedClassifier 30 - mic.inputMappedClassifier 31 - mic.inputMappedClassifier 32 - mic.inputMappedClassifier 33 - mic.inputMappedClassifier 34 - mic.inputMappedClassifier 35 - mic.inputMappedClassifier 35 - mic.inputMappedClassifier 35 - mic.inputMappedClassifier 36 - mic.inputMappedClassifier 37 - mic.inputMappedClassifier 38 - mic.inputMappedClassifier 38 - mic.inputMappedClassifier 39 - mic.inputMappedClassifier 30 - mic.inputMappedClassifier 31 - mic.in	Clastifer output O 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Year (nominal) company (nominal) rolLear (nominal) Sentime Time taken to bui === Evaluation or Time taken to tes === Summary === Correctly Classif Incorrectly Classif Encorrectisticic Mean absolute er: Root mean squarec	Visualize 8-6'-5'-V week 25: 15: > 1 16: 17: 18: 19: 19: 19: 10: 19: 10: 10: 10: 10: 10: 10: 10: 10		outes 		num-slots 1	•••	-	
Rattly cloudy Rattly cloudy Ratsly cluster Ress Cassify Cluster Asso CostSensitiveClassifier -N Cost CostSensitiveClassifier -N Cost CostSensitiveClassifier -N CostSensitiveClassifier -N CostSensitiveClassifier - CostSensitive	Clastifer output O 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Year (nominal) company (nominal) rollear (nominal) Sentime Time taken to bui === Evaluation or Time taken to test === Summary === Correctly Classif Incorrectly Classif Incorrectly Classif Excorrectisticic Mean absolute eri Relative absolute Relative absolute solutives	Visualize 8-6'-51-W week 15: > 1 tek> 2 hing> 4 hint> 5 lid model: 0.1 a test set ===== tified Instance: ified Instance: ified Instance: aread error instende error	coming attribution of the second seco	outes 		num-slots 1	•••	-	
Partly cloudy eka Explorer Coss Classify Cluster Asso er coces CostSensitiveClassifier -N ** tions et raining set polied test set Set_ cost-validation Folds 10 contage spit % 66 5 contage spit % 5 to fight-click for optiona) 33 - mic.hgutAppedClassifier 51 - mic.hgutAppedClassifier 53 - mic.hgutAppedClassifier 54 - mic.hgutAppedClassifier 55 - mic.hgutAppedClass	Clare Select attributes CLOProgram Files\\Wekes-3- Classifier output 0 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Year (nominal) Company (nominal) Sentime Time taken to bui === Evaluation or Time taken to test === Summary ==== Correctly Classifi Incorrectly Classifi Incorrectly Classifi Correctly Classifi Report mean squaree Relative absolute of 1 === Detailed Acc	Visualize 8-6'-51-W week 15: > 1 tek> 2 hing> 4 hing> 4 hint> 5 hid model: 0.13 h test set ===== tified Instance: field Instance: field Instance: areade error fared error	coming attribution of the second seco	andomForest of butes 		num-slots 1	-K0-M1.0-W	V 0.001 - S 1	
Rently cloudy Reactly	Clare Select attributes CLAProgram Files/\Wekes-2- Classifier output 0 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Year (nominal) Year (nominal) Year (nominal) Sentime Time taken to bui Evaluation or Time taken to test 	Visualize 8-6'-51-W weiko 3-5'-51-W weiko > 1 > 1 > 1 > 1 > 1 > 1 > 2 > 1 > 2 > 1 > 2 > 2 > 2 > 4 > 5 > 4 > 5 > 4 > 5 > 4 > 5 > 4 > 5 > 4 > 5 > 4 > 4 > 5 > 4 > 5 >> 5 > 4 > 5 > 4 > 5 > 4 >> 5 >>>>>>>>>>	coming attribution of the second seco	andomForest of butes 	onds 41.6667 58.3333	num-slots 1	-K0-M1.0-V	V0.001-S 1	Class Positive
Rently cloudy Reactly	Classifier output Classifier output 0 150 0 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Pollear (nominal) Sentiss Time taken to buil === Evaluation or Time taken to tes === Summary === Correctly Classif Kappa statistic Kappa statistic Kappa statistic Root relative solute err Root mean squares Robitute err Root relative solute err Root	Visualize 8-6'-51-W weiko 3-5'-51-W weiko > 1 > 1 > 1 > 1 > 1 > 1 > 2 > 1 > 2 > 1 > 2 > 2 > 2 > 4 > 5 > 4 > 5 > 4 > 5 > 4 > 5 > 4 > 5 > 4 > 5 > 4 > 4 > 5 > 4 > 5 >> 5 > 4 > 5 > 4 > 5 > 4 >> 5 >>>>>>>>>>	coming attribution of the second seco	andomForest Duttes Pattes Pattes panyliame Leaning tLiment set: 0 seco .0455 .5393 .6263 .5993 .4595	onds 41.6667 58.3333	num-slots 1	-K0-M10-V	70.001-5 1	Class
Veka Explorer Ve	Clare Select attributes CLAProgram Files/\Wekes-2- Classifier output 0 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Year (nominal) Year (nominal) Year (nominal) Sentime Time taken to bui Evaluation or Time taken to test 	Visualize 8-6'-51 -W week standard and a standard and and a stan	coming attribution of the second seco	andomForest of butes 	onds 41.6667 58.3333	num-slots 1	-K0-M1.0-V	PRC Area 0.741 0.961	Class Positive
Veka Explorer Ve	Classifier output Classifier output Classifier output 0 150 Attributes mapping Model attributes Model attributes Model attributes Inominall year Inominall Pollear Inominall Pollear Inominall Pollear Time taken to bui Time taken to tes Summary === Correctly Classif Kapp statistic Kapp statistic Robits absolute Robits absolute	Visualize 8-6'-S1-W week rs: II II II II II II II II II I	coming attribution of the second seco	andomForest of butes 	onds 41.6667 58.3333	num-slots 1	-K0-M1.0-V	PRC Area 0.741 0.961	Class Positive
Rattly cloudy Rattly cloudy Ratsly cluster Ress Cassify Cluster Asso CostSensitiveClassifier -N Cost Cost CostSensitiveClassifier -N Cost CostSensitiveClassifier -N Cost CostSensitiveClassifier - Cost	Classifier output Classifier output Classifier output 0 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Pollear (nominal) John Time taken to buil Time taken to buil Time taken to tes Summary === Correctly Classif Kapp statistic Kapp statistic Relative absolute err Root relative solute err R	Visualize 8-6'-S1-W week 1 1 1 1 1 2 1 1 2 1 1 2 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1	coming attribution of the second seco	andomForest of butes 	onds 41.6667 58.3333	num-slots 1	-K0-M1.0-V	PRC Area 0.741 0.961	Class Positive
	Clastific output Clastific output Clastific output 0 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Year (nominal) Company (nominal) Sentime Time taken to bui == Evaluation or Time taken to bui == Evaluation or Time taken to test == Summary === Correctly Classifi Honoractly Classifi Reop statistic Reot relative apport Total Number of 1 == Detailed Accor Weighted Avg. == Confusion Mat a b < clast	Visualize 8-6'-51-W week 1 1 1 1 1 2 1 2 1 2 2 1 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2	coming attribution of the second seco	andomForest of butes 	onds 41.6667 58.3333	num-slots 1	-K0-M1.0-V	PRC Area 0.741 0.961	Class Positive
tty cloudy Explorer CassSify Cluster Asse CassSify Set CassSify Set CassSify Set CassSify Set CassSify Set CassSify Set CassSify Cluster CassSify	Classifier output Classifier output Classifier output 0 150 5 0 Attribute mapping Model attributes (nominal) Year (nominal) Pollear (nominal) John Time taken to buil Time taken to buil Time taken to tes Summary === Correctly Classif Kapp statistic Kapp statistic Relative absolute err Root relative solute err R	Visualize 8-6'-51-W week 1 1 1 1 1 2 1 2 1 2 2 1 2 2 1 2 2 2 2 2 2 2 2 2 2 2 2 2	coming attribution of the second seco	andomForest of butes 	onds 41.6667 58.3333	num-slots 1	-K0-M1.0-V	PRC Area 0.741 0.961	Class Positive

Classifier Choose CostSensitiveClassifier -cost	matrix (10.0 10.0 0.0; 10.0 0.0; 0.0 1.0 0.0]" - S1 - W weka dassifiers rules ZeroR
Test options	Classifier output
Use training set	
Supplied test set Set	Bagging with 100 iterations and base learner
Cross-validation Folds 10	
Percentage split % 66	weka.classifiers.trees.RandomTree -K 0 -M 1.0 -V 0.001 -S 1 -do-not-check-capabilities Attribute mappings:
	Attribute mappings:
More options	Model attributes Incoming attributes
(Nom) Sentiment ~	
(Nom) Sentiment ~	(nominal) Year> 1 (nominal) Year
Start Stop	(nominal) DayofWeek> 2 (nominal) DayofWeek
Result list (right-click for options)	(nominal) companyName> 3 (nominal) companyName (nominal) Polleaning> 4 (nominal) Polleaning
10:21:13 - misc.InputMappedClassifier	(nominal) Folleening> 4 (nominal) Folleening (nominal) Sentiment> 5 (nominal) Sentiment
10:22:30 - misc.InputMappedClassifier	
10:27:55 - misc.InputMappedClassifier	
10:29:10 - misc.InputMappedClassifier	Time taken to build model: 0.11 seconds
10:31:46 - misc.InputMappedClassifier	=== Evaluation on test set ===
10:39:55 - misc.InputMappedClassifier	=== Evaluation on test set ===
10:43:32 - misc.InputMappedClassifier 10:45:01 - misc.InputMappedClassifier	Time taken to test model on supplied test set: 0 seconds
10:46:31 - misc.inputMappedClassifier	
10:47:51 - misc.InputMappedClassifier	=== Summary ===
10:48:36 - misc.InputMappedClassifier	Correctly Classified Instances 10 0.3333 %
10:50:12 - misc.InputMappedClassifier	Correctly Classified Instances 10 8.3333 % Incorrectly Classified Instances 110 91.6667 %
10:52:38 - misc.InputMappedClassifier	Kappa statistic 0
10:55:37 - misc.InputMappedClassifier	Mean absolute error 0.9048
20:16:06 - misc.InputMappedClassifier	Root mean squared error 0.9459
20:16:36 - misc.InputMappedClassifier	Relative absolute error 101.0782 %
20:17:05 - misc.InputMappedClassifier 20:17:07 - misc.InputMappedClassifier	Not relative squared error 101.4140 % Total Number of Instances 120
20:17:07 - misc.inputwappedclassiner	TOTAL NUMBER OF INSTANCES 120
	=== Detailed Accuracy By Class ===
	TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class
	1.000 1.000 0.083 1.000 0.154 ? 0.709 0.159 Positive
	0.000 0.000 7 0.000 7 7 0.709 0.952 Negative
	=== Confusion Matrix ===
	a b < classified as
	10 0 a = Positive
	110 0 b = Negative
Status	
Ж	

Random Forest Lexicon 3 - Base

Preprocess Classify Cluster Associ Classifier	ciate Select attributes Visualize	
Choose CostSensitiveClassifier -cost-n	-matrix "(0.0 1.0.0 .0; 1.0 0.0; 0.0 1.0 0.0)" - S 1 - W weka dassifiers.rules ZeroR	
Test options	Classifier output	
 Use training set 	0 1	
Supplied test set Set	10 0	
Cross-validation Folds 10	Attribute mappings:	
Percentage split % 66	Received mappings.	
More options	Model attributes Incoming attributes	
More options		
(Nom) Sentiment	(nominal) Year> 1 (nominal) Year (nominal) Dayoffweek> 2 (nominal) Dayoffweek	
	(nominal) bayUsweek 2 (nominal) bayUsweek (nominal) companyName 2 (nominal) companyName	
Start Stop	(nominal) Polleaning> 4 (nominal) Polleaning	
Result list (right-click for options)	(nominal) Sentiment> 5 (nominal) Sentiment	
10:21:13 - misc.InputMappedClassifier		
10:22:30 - misc.InputMappedClassifier	Time taken to build model: 0.01 seconds	
10:27:55 - misc.InputMappedClassifier	Time taken to build model: 0.01 seconds	
10:29:10 - misc.InputMappedClassifier 10:31:46 - misc.InputMappedClassifier	=== Evaluation on test set ===	
10:39:55 - misc.inputMappedClassifier		
10:43:32 - misc.InputMappedClassifier	Time taken to test model on supplied test set: 0 seconds	
10:45:01 - misc.InputMappedClassifier	mms Summary mmm	
10:46:31 - misc.InputMappedClassifier	Sumary	
10:47:51 - misc.InputMappedClassifier	Correctly Classified Instances 10 8.3333 %	
10:48:36 - misc.InputMappedClassifier	Incorrectly Classified Instances 110 91.6667 %	
10:50:12 - misc.InputMappedClassifier 10:52:38 - misc.InputMappedClassifier	Kappa statistic 0	
10:52:38 - misc.inputMappedClassifier	Mean absolute error 0.7425 Root mean squared error 0.7597	
20:16:06 - misc.InputMappedClassifier	Relative absolute error 82,9479 %	
20:16:36 - misc.InputMappedClassifier	Root relative squared error 81.4521 %	
20:17:05 - misc.InputMappedClassifier	Total Number of Instances 120	
20:17:07 - misc.InputMappedClassifier		
	=== Detailed Accuracy By Class ===	
	TP Rate FF Rate Precision Recall F-Measure MCC ROC Area PRC Area Class	
	1.000 1.000 0.083 1.000 0.154 ? 0.500 0.083 Positive	
	0.000 0.000 ? 0.000 ? ? 0.500 0.917 Negative	
	Weighted Avg. 0.083 0.083 ? 0.083 ? 0.500 0.847	
	Confusion Matrix	
	a b < classified as	
	10 0 a = Positive	
	110 0 b = Negative	
Status		
OK		Log 💉 🕬

Random Forest Lexicon 3 - Cost Sensitive Evaluator

Choose RandomForest -P 100 -I 100 -	-num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1
est options Use training set Supplied test set Cross-validation Folds Cross-validation Folds Cross-validation Folds More options (Norn) Sentiment Start Stop Result Bit (right-tick for options) 102133 - mic.chrutMappedClassifier 102236 - mic.ehrutMappedClassifier 102256 - mic.ehrutMappedClassifier 102256 - mic.ehrutMappedClassifier 102256 - mic.ehrutMappedClassifier 102256 - mic.ehrutMappedClassifier	Classifier output Bagging with 100 iterations and base learner weka.classifiers.trees.RandomTree -K 0 -M 1.0 -V 0.001 -S 1 -do-not-check-capabilities Attribute mappings: Model.attributes Incoming attributes incominall Year
1022-0 - misc.inputMapped.classifier 102755 - misc.inputMappedClassifier 1029:10 - misc.inputMappedClassifier 1039:55 - misc.inputMappedClassifier 104352 - misc.inputMappedClassifier 1044751 - misc.inputMappedClassifier 104631 - misc.inputMappedClassifier 1046361 - misc.inputMappedClassifier 105012 - misc.inputMappedClassifier	Time taken to build model: 0.1 seconds === Evaluation on test set === Time taken to test model on supplied test set: 0 seconds === Summary === Correctly Classified Instances 11 9.1667 % Incorrectly Classified Instances 109 90.8333 % Rappa statistic 0.0015 Mean absolute error 0.9022 Root mean suparted error 0.9438 Relative absolute error 101.443 % Root relative squared error 101.4011 % Total Number of Instances 120
	<pre> Detailed Accuracy By Class TP Rate FP Rate Precision Recall F-Measure MCC ROC Area FRC Area Class 1.000 0.991 0.084 1.000 0.155 0.028 0.661 0.148 Positive 0.009 0.000 1.000 0.009 0.018 0.029 0.029 0.029 0.029 Regative Weighted Avg. 0.092 0.083 0.924 0.092 0.029 0.028 0.661 0.862</pre>

Random Forest Lexicon Combined - Base

Test options Class	x "[0.0.10.0.0.0; 1.0.0.0; 0.0.1.0.0.0]" -5.1 -W weka dassifiers.rules.ZeroR		
Test options Class			
	silier output		
Supplied test set Set	<pre>0 1 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</pre>		
Wei 11 Status	TP Rate FP Rate Precision Recall F-Measure MOC ROC ROC Area Class 1.000 1.000 0.003 1.000 0.154 7 0.500 0.003 positive ighted Avg. 0.083 0.083 7 0.500 0.917 Megative = Confusion Matrix === 4 6 < classified as 0 0 1 0 1 b = Nositive 10 0 b = Nositive </th <th>100</th> <th></th>	100	
ОК		Log	🕼 . × (

Random Forest Lexicon Combined - Cost Sensitive Evaluator

References

What is Random Forest? | IBM

Support Vector Regression (SVR) - One of the Most Flexible Yet Robust Prediction

Algorithms| by Saul Dobilas | Towards Data Science

Random forest - Wikipedia

Bootstrap aggregating - Wikipedia

MPQA University of Pittsburgh Lexicon

Bing Liu and Minqing Hu Sentiment Lexicon

SenticNet Lexicon

More visualizations can be found in the attached Viz.pptx file