Nikhil Paonikar and Connor Pigott

Dr. Olfa Nasraoui

CECS 621-50 Web Mining

27 August 2018

SENTIMENT DETECTION IN NEWSPAPER HEADLINES

INTRODUCTION

While machines can perform many of the same functions that humans can do - in some cases much better than humans can - they can only do as they are instructed. Machines operate in concrete, structured, and rigid manners when humans do not always. This particular discrepancy is observed in human communication, where sentiment can be conveyed through tone, body language, or the manner in which words are spoken or written. Interpretation of sentiments from online text can be difficult for humans to perform but even more so for machines. This analysis aims to explore methods in which machines could potentially identify human sentiments in user-generated text, which may aid in identify overall user sentiments toward ideas, media, products, and other possibilities.

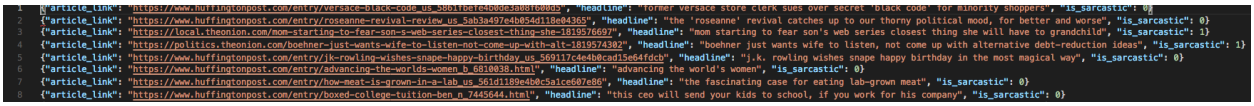
The objective will be to take a data set of compiled headlines from differing sources - a contemporary news outlet and a satirical news outlet - and identifying if a set classification models can reasonably detect the sarcasm present in each of the headlines and to assess how well each of the models performs. The models, Naïve Bayes, logistic regression, and support vector machines (SVM) are commonly used classifiers that can be applied to the data within the software Weka.

DATASET DESCRIPTION

The dataset consists of 26,709 instances of newspaper headlines classified as either "sarcastic" or "not sarcastic."[1] The 14,985 not sarcastic headlines were compiled from the Huffington Post while the 11,724 sarcastic headlines were compiled from The Onion, a satirical newspaper. In its raw form the data consisted of 3 attributes: the URL to the news article, the article headline, and a binary class attribute indicating whether or not the headline was sarcastic.

Attribute	**Description**
URL	Contains the URL to the article identified by the headline
Headline	Contains a string of words that indicate either a sarcastic or not sarcastic sentiment
Is_sarcastic	A binary attribute indicating the presence of sarcasm in the headline as evidenced by the publisher of the news article

Complete attribute label and descriptions



Raw Data in json format

former versace store clerk sues over secret black code for minority shoppers	0
the 'roseanne' revival catches up to our thorny political mood, for better and worse	0
j.k. rowling wishes snape happy birthday in the most magical way	0
advancing the world's women	0
the fascinating case for eating lab-grown meat	0
this ceo will send your kids to school, if you work for his company	0
friday's morning email: inside trump's presser for the ages	0
airline passengers tackle man who rushes cockpit in bomb threat	0
facebook reportedly working on healthcare features and apps	0
north korea praises trump and urges us voters to reject 'dull hillary'	0

Selection of Not Sarcastic Headlines

[1] https://www.kaggle.com/rmisra/news-headlines-dataset-for-sarcasm-detection/home

study: majority of americans not informed enough to stereotype chechens	1
nation's prospective college applicants go straight to princeton review's 'best college radio	1
touring raffi refuses to play 'shake my sillies out'	1
school surprised to learn student committed suicide over pressures of intro to communica	1
destruction of rainforest cafe clears room for new hooters	1
food network goes off air after every possible iteration of ingredient combinations complete	1
area man honored to have name in hat	1
new pepsi product specifically mentions target demographic in name	1
company to use internet to waste money, employees' time	1
convention-goer removes name tag, vanishes back into world of anonymous hilton orland	1

Selection of Sarcastic Headlines

DATA MINING GOAL

Classification: Our data mining goal is to identify the sentiment or authenticity of newspaper headlines produced by news agencies and comedic outlets. To identify the best model to meet this objective, Naïve Bayes, Support Vector Machines, and Simple Logistic Regression models will be applied to the data to classify these records. All applications of these models will be performed within the software Weka.

In addition to a simple determination of how these models perform against a set of data, this analysis aims to explore how much data is necessary to train these models for accurate classification. To perform this analysis, the three aforementioned models are run against two, additional sets of reduced data - sets containing 66% and 33% of the original headlines.

Percentage of Headlines	Total Headlines	Sarcastic Headlines	Not Sarcastic Headlines
100%	26,709	11,724	14,985
66%	17,628	7,183	10,445
33%	8,814	2,909	5,905

Breakdown of data sets used

Of the complete set of headlines, the instances ranged from 2 to 29 words. Median and average length were 10 words and 9.845 words respectively.

METHODOLOGY

Pre-processing: Before application of the algorithms, the data required a number of pre-processing measures. Initially, the raw data was converted into an arff format, where the URLs were removed given that they had no bearing on the sentiment of the headline. Within Weka, a String-to-Word filter was applied to the headlines attributes - resulting in the distinct words appearing in the set of headlines as individual, binary attributes. A stop word list was applied beforehand to eliminate common English words which may either skew the data or add no value to the algorithms. Words to keep was also adjust to 1000.

For each set of data, the data were then ranked by order of ability to indicate a given sentiment. Attributes with a score of over 0.0002 were kept while others were discarded.

Percentage of Headlines	Total Word Attributes Following String to Word Filter	Reduced Attributes After Ranking
100%	1,508	660
66%	1,500	585
33%	1,701	507

Table summarizing attribute reduction following InfoGain.

```
Ranked attributes:
 0.020618      623 man
 0.020058     1208 area
 0.010562      297 donald
 0.008951     1083 trump
 0.007502      851 report
 0.006909     1084 trumps
 0.005245      996 study
 0.005197      690 nation
```

Screenshot of the Highest-Scoring Attribute Outputs

INPUTS AND OUTPUTS

For each set of data, the following inputs will be provided for analysis:

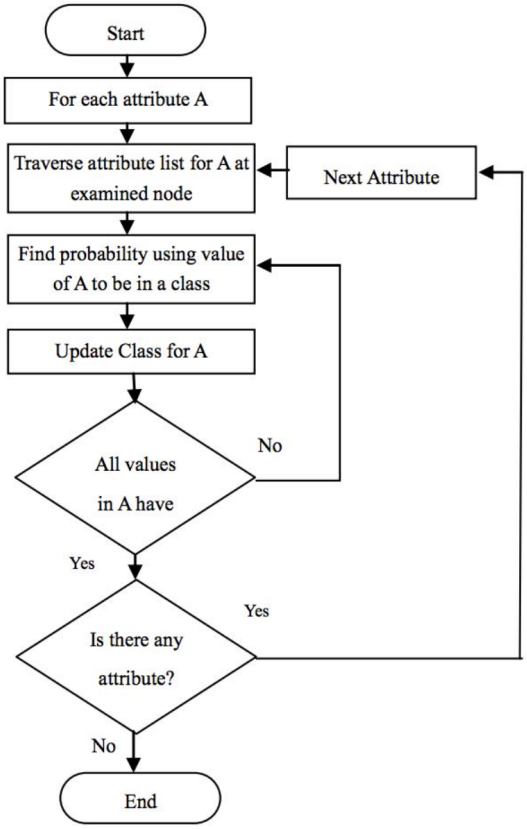
1. For each instance, a series of binary attributes indicating the presence of a word in that instance's headline

2. A binary indication as to whether the instance's headline is classified sarcastic or not

The models will provide predictions based on the word attributes as to whether each headline is sarcastic or not sarcastic. As mentioned above, the amount of data supplied to the models as training data is varied to assess the impact of training data completeness' effect on the models.

ALGORITHMS

Naïve Bayes: The Naïve Bayes is a classification technique based on Bayes' Theorem. It assumes independence among its predictors; i.e., a Naïve Bayes classifier assumes that the presence of a specific feature in a class is unrelated to the presence of other features. Even if these features depend on each other or upon the existence of the other features, all of these properties independently contribute to a resultant probability. This model is easy to build and particularly useful for very large data sets. It has been known to outperform even highly sophisticated classification methods.[2]

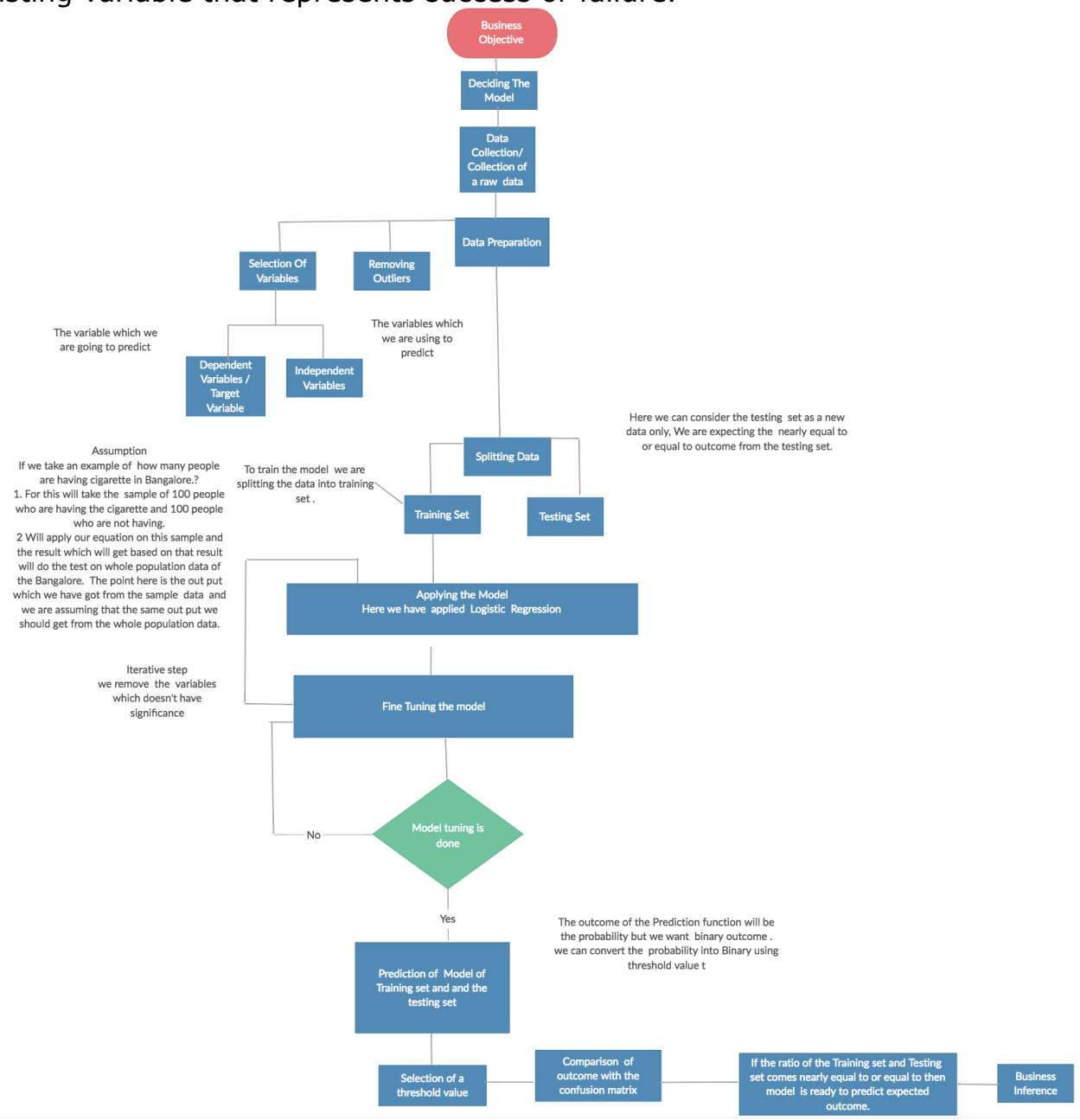


[2] https://file.scirp.org/pdf/JSEA_2013042913162682.pdf

Logistic Regression

Logistic regression is a binary classification algorithm that Modifies linear regression to fit logistic function. The output is the probability of a given class.

Logistic regression follows principles similar to those of linear regression. However, an exception to this is that the outcome is a contrasting variable that represents success or failure.[3]



[3] https://creately.com/diagram/example/iq36neav/Logistic%20Regression%20Flow%20Chart

Support Vector Machine

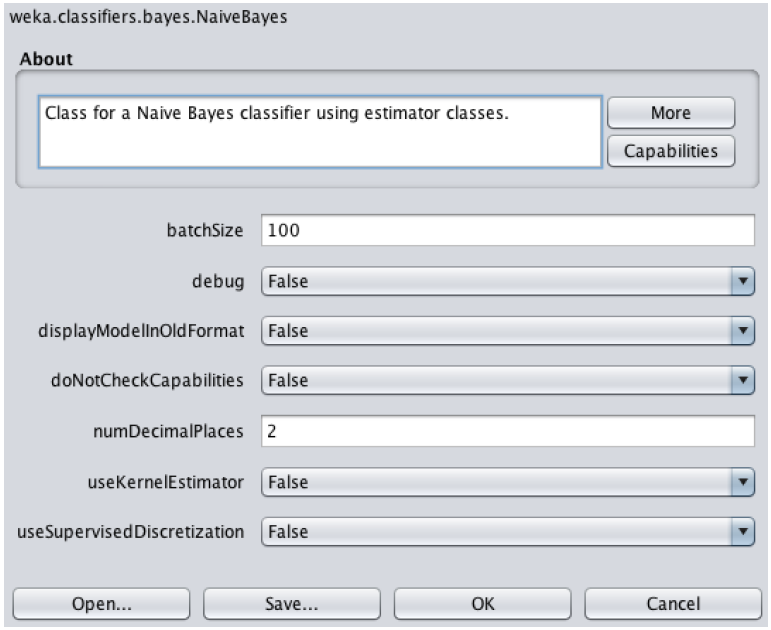
A Support Vector Machine (SVM) is a discriminative classifier. It is formally defined by a separating hyperplane; i.e., given labeled training data (supervised learning), this algorithm produces outputs in the form of an optimal hyperplane which categorizes new examples. In two dimensional space, this hyperplane is a line dividing a plane in two parts where in each class lay in either side. [4]

candidateSV = { closest pair from opposite classes }
while there are violating points **do**
  Find a violator
  candidateSV = candidateSV $\bigcup$ violator
  if any $\alpha_p < 0$ due to addition of c to S **then**
    candidateSV = candidateSV $\setminus p$
    repeat till all such points are pruned
  end if
end while

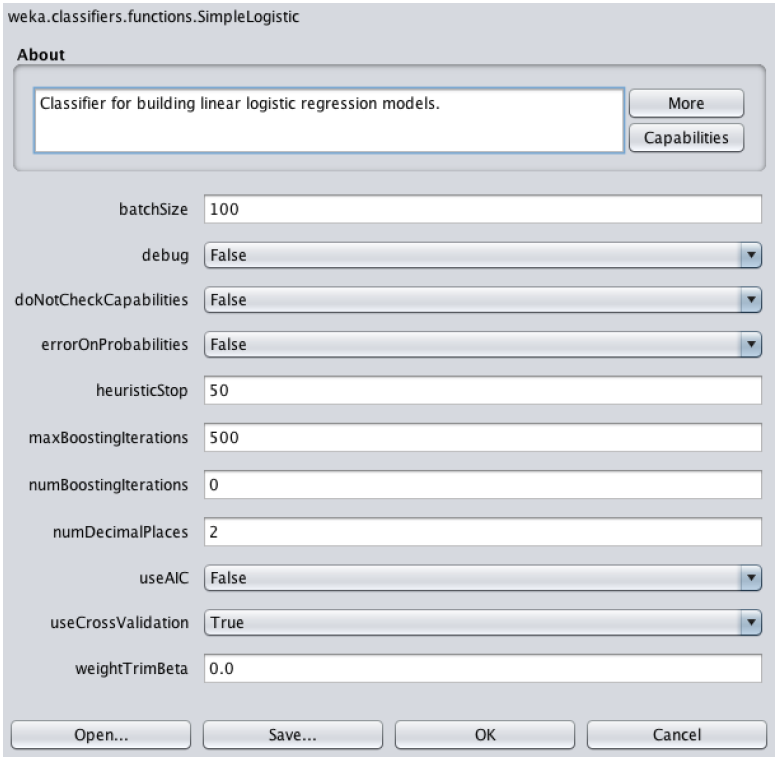
[4] https://file.scirp.org/pdf/JSEA_2013042913162682.pdf

Complexity	Naïve Bayes	Logistic Regression	SVM
Space	O(pqr), where p is the number of features, q is values for each feature, and r is alternative values for the class.	O(1)	$O(n^2)$
Time	O(Np), where N is the number of training samples and p is the number of features.	O(n)	$O(N^3)$

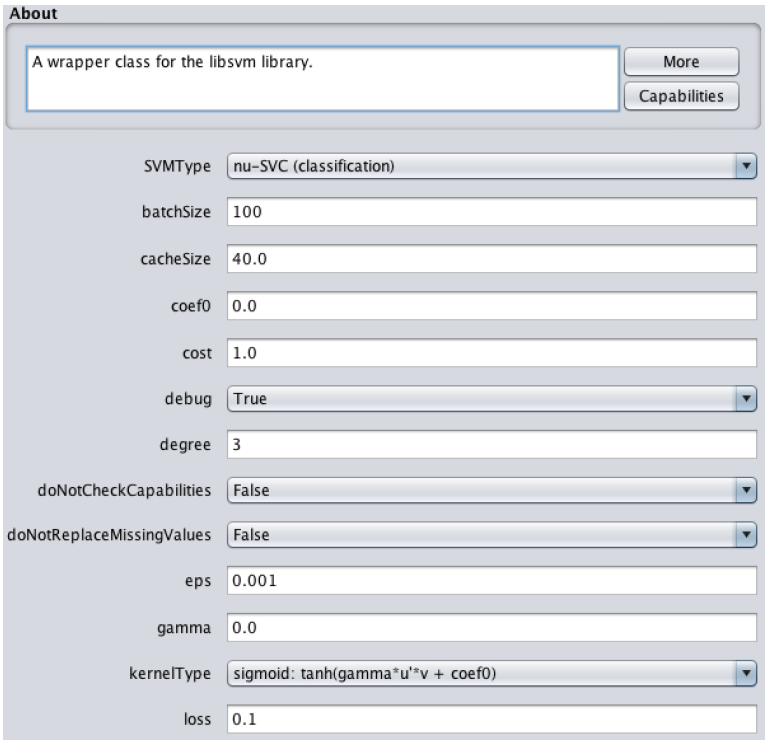
IMPLEMENTATION



Naïve Bayes Settings



Logistic Regression Settings



SVM Settings

TESTING

Tests with 100 Percent of Data

```
=== Summary ===

Correctly Classified Instances       16910               63.312  %
Incorrectly Classified Instances      9799               36.688  %
Kappa statistic                          0.1866
Mean absolute error                      0.4394
Root mean squared error                  0.4684
Relative absolute error                 89.2038 %
Root relative squared error             94.3878 %
Total Number of Instances            26709

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
               0.976     0.805     0.608      0.976     0.749      0.284    0.798     0.816     0
               0.195     0.024     0.863      0.195     0.318      0.284    0.798     0.749     1
Weighted Avg.  0.633     0.462     0.720      0.633     0.560      0.284    0.798     0.787

=== Confusion Matrix ===

     a     b   <-- classified as
 14622   363 |     a = 0
  9436  2288 |     b = 1
```

Naïve Bayes Summary

```
=== Summary ===

Correctly Classified Instances       20026               74.9785 %
Incorrectly Classified Instances      6683               25.0215 %
Kappa statistic                          0.4771
Mean absolute error                      0.3334
Root mean squared error                  0.4048
Relative absolute error                 67.6875 %
Root relative squared error             81.573  %
Total Number of Instances            26709

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
               0.880     0.416     0.730      0.880     0.798      0.492    0.827     0.854     0
               0.584     0.120     0.791      0.584     0.672      0.492    0.827     0.804     1
Weighted Avg.  0.750     0.286     0.757      0.750     0.743      0.492    0.827     0.832

=== Confusion Matrix ===

     a     b   <-- classified as
 13182  1803 |     a = 0
  4880  6844 |     b = 1
```

Logistic Regression Summary

```
=== Summary ===

Correctly Classified Instances       20186               75.5775 %
Incorrectly Classified Instances      6523               24.4225 %
Kappa statistic                          0.4914
Mean absolute error                      0.2442
Root mean squared error                  0.4942
Relative absolute error                 49.584  %
Root relative squared error             99.5832 %
Total Number of Instances            26709

=== Detailed Accuracy By Class ===

               TP Rate   FP Rate   Precision   Recall  F-Measure  MCC      ROC Area  PRC Area  Class
               0.873     0.394     0.739      0.873     0.800      0.503    0.739     0.716     0
               0.606     0.127     0.789      0.606     0.685      0.503    0.739     0.651     1
Weighted Avg.  0.756     0.277     0.761      0.756     0.750      0.503    0.739     0.688

=== Confusion Matrix ===

     a     b   <-- classified as
 13080  1905 |     a = 0
  4618  7106 |     b = 1
```

SVM Summary

Tests with 66 Percent of Data

```
=== Summary ===

Correctly Classified Instances       11380               64.5564 %
Incorrectly Classified Instances      6248               35.4436 %
Kappa statistic                          0.1597
Mean absolute error                      0.4313
Root mean squared error                  0.464
Relative absolute error                 89.3077 %
Root relative squared error             94.4366 %
Total Number of Instances            17628

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.980    0.840    0.629      0.980    0.766      0.257    0.800     0.835     0
               0.160    0.020    0.843      0.160    0.269      0.257    0.800     0.727     1
Weighted Avg.  0.646    0.506    0.716      0.646    0.563      0.257    0.800     0.791

=== Confusion Matrix ===

     a     b   <-- classified as
 10231   214 |     a = 0
  6034  1149 |     b = 1
```

Naïve Bayes Summary

```
=== Summary ===

Correctly Classified Instances       13338               75.6637 %
Incorrectly Classified Instances      4290               24.3363 %
Kappa statistic                          0.4721
Mean absolute error                      0.3275
Root mean squared error                  0.4022
Relative absolute error                 67.813  %
Root relative squared error             81.8442 %
Total Number of Instances            17628

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.894    0.444    0.746      0.894    0.813      0.488    0.826     0.867     0
               0.556    0.106    0.784      0.556    0.651      0.488    0.826     0.783     1
Weighted Avg.  0.757    0.306    0.761      0.757    0.747      0.488    0.826     0.833

=== Confusion Matrix ===

    a    b   <-- classified as
 9342 1103 |    a = 0
 3187 3996 |    b = 1
```

Logistic Regression Summary

```
=== Summary ===

Correctly Classified Instances       13378               75.8906 %
Incorrectly Classified Instances      4250               24.1094 %
Kappa statistic                          0.4777
Mean absolute error                      0.2411
Root mean squared error                  0.491
Relative absolute error                 49.9282 %
Root relative squared error             99.9284 %
Total Number of Instances            17628

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.893    0.437    0.748      0.893    0.815      0.493    0.728     0.732     0
               0.563    0.107    0.784      0.563    0.656      0.493    0.728     0.620     1
Weighted Avg.  0.759    0.302    0.763      0.759    0.750      0.493    0.728     0.686

=== Confusion Matrix ===

    a    b   <-- classified as
 9332 1113 |    a = 0
 3137 4046 |    b = 1
```

SVM Summary

Tests with 33 Percent of Data

```
=== Summary ===

Correctly Classified Instances        6184               70.1611 %
Incorrectly Classified Instances      2630               29.8389 %
Kappa statistic                          0.1482
Mean absolute error                      0.3975
Root mean squared error                  0.4474
Relative absolute error                 89.8719 %
Root relative squared error             95.1505 %
Total Number of Instances             8814

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.979    0.862    0.698      0.979    0.815      0.233    0.802     0.875     0
               0.138    0.021    0.766      0.138    0.234      0.233    0.802     0.656     1
Weighted Avg.  0.702    0.584    0.720      0.702    0.623      0.233    0.802     0.803

=== Confusion Matrix ===

    a    b   <-- classified as
 5782  123 |    a = 0
 2507  402 |    b = 1
```

Naïve Bayes Summary

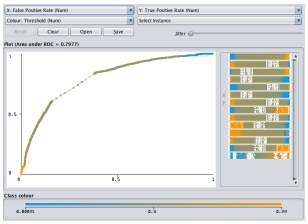
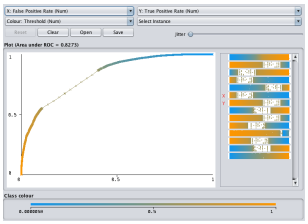
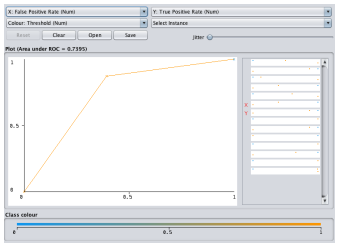
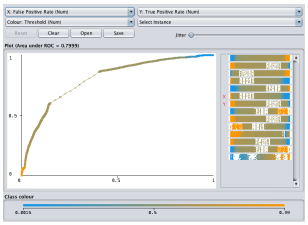
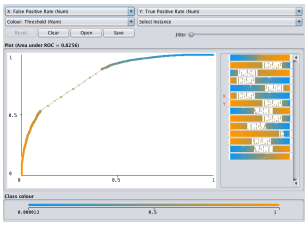
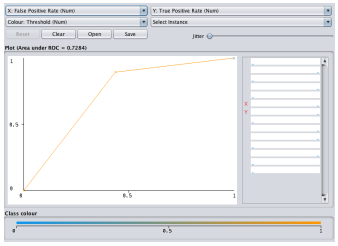
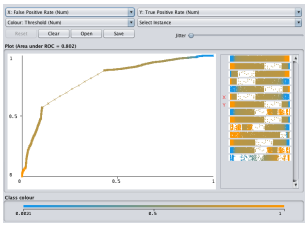
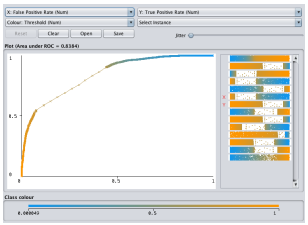
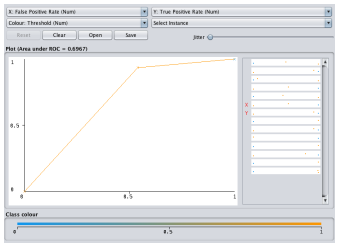
```
=== Summary ===

Correctly Classified Instances        7000               79.4191 %
Incorrectly Classified Instances      1814               20.5809 %
Kappa statistic                          0.4888
Mean absolute error                      0.2921
Root mean squared error                  0.38
Relative absolute error                 66.0475 %
Root relative squared error             80.81   %
Total Number of Instances             8814

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.933    0.488    0.795      0.933    0.859      0.511    0.838     0.908     0
               0.512    0.067    0.791      0.512    0.621      0.511    0.838     0.747     1
Weighted Avg.  0.794    0.349    0.794      0.794    0.780      0.511    0.838     0.855

=== Confusion Matrix ===

    a    b   <-- classified as
 5512  393 |    a = 0
 1421 1488 |    b = 1
```

Logistic Regression Summary

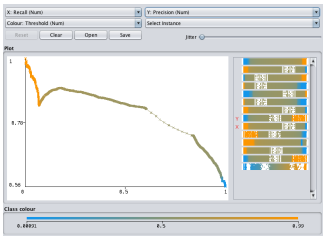
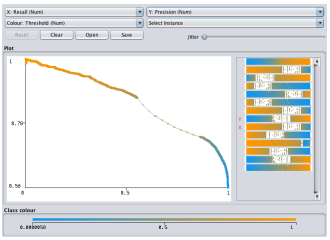
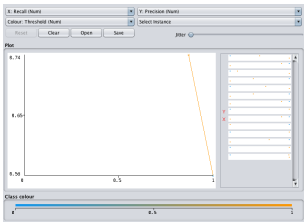
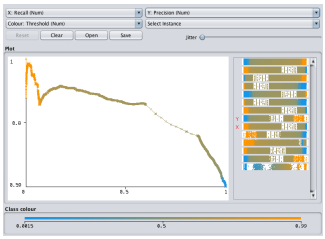
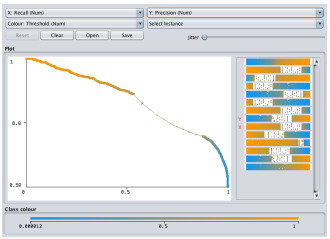
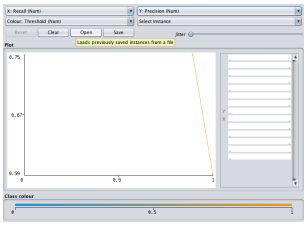
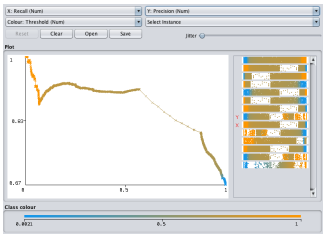
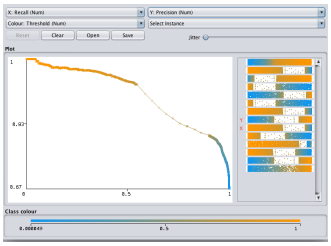
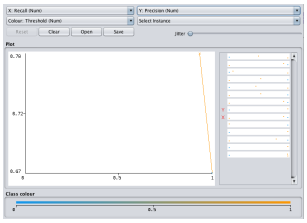
```
=== Summary ===

Correctly Classified Instances        6851               77.7286 %
Incorrectly Classified Instances      1963               22.2714 %
Kappa statistic                          0.4386
Mean absolute error                      0.2227
Root mean squared error                  0.4719
Relative absolute error                 50.36   %
Root relative squared error            100.361  %
Total Number of Instances             8814

=== Detailed Accuracy By Class ===

               TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
               0.934    0.540    0.778      0.934    0.849      0.466    0.697     0.771     0
               0.460    0.066    0.774      0.460    0.577      0.466    0.697     0.534     1
Weighted Avg.  0.777    0.384    0.777      0.777    0.759      0.466    0.697     0.693

=== Confusion Matrix ===

    a    b   <-- classified as
 5514  391 |    a = 0
 1572 1337 |    b = 1
```

SVM Summary

ROC Curves

Percentage of Data	Naïve Bayes	Logistic Regression	SVM
100			
66			
33			

Precision Recall Curves

Percentage of Data	Naïve Bayes	Logistic Regression	SVM
100			
66			
33			

EXPERIMENTAL RESULTS

After reviewing the spectrum of classification models and the amount of inputs provided, we observe noticeable differences in the accuracy, precision, and computation time amongst each set of conditions - where accuracy ranged between 60% and 80%. In each instance where the amount of training data varied, we observe the the Naive Bayes model underperformed performed the other two models in accuracy and in precision/recall. The differences between logistic regression and SVM is less pronounced; both maintain similar accuracies while logistic regression consistently maintains a higher area under ROC.

Perhaps the most distinguished observation that is consistent across the majority of models and data is the effect of amount of training data on accuracy and recall/precision. In the case of each statistical model, a smaller amount of training data amounted to heightened accuracy - with accuracy increasing nearly 5% in the case of logistic regression and almost 7% in the case of Naïve Bayes. In the cases of the logistic regression and Naïve Bayes, we see that ROC area is increased as data was reduced. Only in the case of SVM did ROC area actually decrease as the training data was lessened - though overall accuracy did increase. However, this accuracy can be attributed to the model classifying the vast majority of headlines as "not sarcastic" as "not sarcastic" headlines comprised the majority of analyzed headlines.

One of the factors that would have impacted all trials of the classifications would have been our specific preprocessing techniques and the short length of the data. Upon removing all of our stop words and eliminating "insignificant words" from the set of possible attributes after ranking the words, there were some headlines that possessed no attributes. In these instances, any sarcastic headlines that fell into this category would have assuredly been classified as "not sarcastic" because the model would have had no data to interpret and would likely classify the headline to whichever category was largest (i.e. the "not sarcastic" headlines). This phenomenon was found to have affected 3,258 "not sarcastic" headlines and 2,444 "sarcastic headlines for a total of 5,702 headlines of the initial 26,709.

Examples of headlines rendered featureless after preprocessing

Not Sarcastic	**Sarcastic**

1) "Actually, CNN's Jeffrey Lord has been indefensible for a while" 2) "Craig Hicks Indicted" 3) "Gillian Jacobs on what it's like to kiss Adam Brody" 4) "Amanda Slavin is not just a statistic" 5) "On Pilgrimage in India"	1) "Irish-Americans gear up for the 'reinforcin' o' the stereotypes'" 2) "Herbie goes bananas" 3) "Gondolier ordered to follow that gondola" 4) "Wal-Mart bans semi-nude pantyhose" 5) "Determined ant requires second flicking"

Summary

Percentage of Data	Naïve Bayes	Logistic Regression	SVM (nu-SVC, sigmoid kernel)
100	**Accuracy:** 63.312% **Error:** 36.688% **Area Under ROC:** 0.7977 **Time Req'd to Build Model:** 3.48 seconds	**Accuracy:** 74.9785% **Error:** 25.0215% **Area Under ROC:** 0.8273 **Time Req'd to Build Model:** 1106.64 seconds	**Accuracy:** 75.5775% **Error:** 24.4225% **Area Under ROC:** 0.7395 **Time Req'd to Build Model:** 32.98 seconds
66	**Accuracy:** 64.5564% **Error:** 35.4435% **Area Under ROC:** 0.7999 **Time Req'd to Build Model:** 1.91 seconds	**Accuracy:** 75.6637% **Error:** 24.3363% **Area Under ROC:** 0.8256 **Time Req'd to Build Model:** 1364 seconds	**Accuracy:** 75.8906% **Error:** 24.1094% **Area Under ROC:** 0.7284 **Time Req'd to Build Model:** 12.42 seconds
33	**Accuracy:** 70.1611% **Error:** 29.8389% **Area Under ROC:** 0.8020 **Time Req'd to Build Model:** 0.53 seconds	**Accuracy:** 79.4191% **Error:** 20.5809% **Area Under ROC:** 0.8384 **Time Req'd to Build Model:** 138.71 seconds	**Accuracy:** 77.7286% **Error:** 22.2714% **Area Under ROC:** 0.6967 **Time Req'd to Build Model:** 2.63 seconds

Percentage of Data	SVM* (C-SVC, linear kernel)
100	**Accuracy:** 60.5712% **Error:** 39.4287% **Area Under ROC:** 0.5532 **Time Req'd to Build Model:** 61.08 seconds
66	**Accuracy:** 62.0433% **Error:** 37.9567% **Area Under ROC:** 0.5374 **Time Req'd to Build Model:** 20.4 seconds

33	**Accuracy:** 67.2793% **Error:** 32.7207% **Area Under ROC:** 0.5043 **Time Req'd to Build Model:** 4.21 seconds

CONCLUSION

The performed analysis indicates that the model used to classify text data as “sarcastic” or “not sarcastic” does have a noticeable performance on the outcome though classifying a given model as superior is indeterminable. In each case, the logistic regression models and SVM models outperformed the Naïve Bayes model in the areas of accuracy and recall/precision. While on the smallest set of data, logistic regression demonstrated the greatest accuracy and recall/precision, SVM demonstrated superior accuracy and recall/precision when given more data.

Most curiously, we observed the effect of reduced training data on the accuracy and precision of the classifiers. In each case, the classifiers performed superiorly when data was reduced. This trend appears counterintuitive - that less training data would facilitate greater analysis of sarcastic words and greater accuracy and recall/precision. We do not yet understand exactly this correlation, but such a relationship may be identified through additional testing.

For additional analyses, it may be worthwhile to use separate training data and data for classification. It is possible that when using smaller data sets for training data and also using said data for classification, qualities associated with a given class appear stronger given smaller number of instances supplied. Accordingly, when these strongly associated attributes are identified within the classification trial, the models can better associate them with their proper class. We suggest that repeating these trials while separating training data and data for classification would isolate this potential issue and provide a more robust analysis of the effect of different quantities of training data supplied for text classification.