**Deception and Subjectivity - Part 2**

**Bernoulli and Multinomial Naïve Bayes in Sci-kit Learn**

**Maya Mileva**

**HW6 IST 736**

A picture containing tiled, white

Description automatically generated

**Introduction**

Lying is the part of life that we cannot evade. Whether it is intended or a white lie, we come across lying every day. Evolution did not provide us with some kind of protection from being deceived. Though people can randomly detect that their interlocutor might conceal something, the scientists work on technologies that will make it easier. The accuracy of a polygraph test that has been around for years has for long been debated, so there must be some alternative. It is interesting how technologies can uncover if somebody is trying to deceive you. People seems to value other people`s opinion, but as evolved and complicated as they are as human being, lying is still part of their lives.

During the last few years, reviews have become crucial to the success of a restaurant, as every restaurant owner is aware of the fact that good reviews can boost popularity and profitability, whereas terrible reviews even have the potential of closing businesses down. That is why it is crucial for restauranteurs to understand the impact of review websites such as Yelp, Toptable or TripAdvisor and the role they play the success or downfall of a business.

People write about their positive and negative experiences, give recommendations and advices, which is good and helpful. Then there are the questions: Are those reviews honest? Are they from real people? With the prosperity of businesses included, marketing teams are getting creative. The more positive reviews the company have, the higher in the search engine it appears. There are people getting paid just to write fake reviews. How did the humans do at spotting the fakes? In a word: terrible. New practices use machine learning to train a computer program to judge the trustworthiness of reviews.

**Analysis and Models**

**About the data**

Two data sets will be used in this project. The first one contains two folders: Dog and Travel. Each one has 5 text files: about dog and travel. Those small data sets are added just for evaluation methods.

|  |  |
| --- | --- |
| **Fig. 1 Wordclould on DOG A close up of text on a black background  Description automatically generated** | **Fig. 1 Wordclould on TRAVEL A picture containing text  Description automatically generated** |

The second dataset consists of restaurant reviews. Each row contained a review and a label if the review was a lie (t/f) and the  sentiment of the review (p/n). Some rows contained three columns, while others contained five or more. The original structure intended to consist of three columns - lie, sentiment, and review. Data was imported and converted to a pandas  data frame with tab delimitation.

A white sign with black text

Description automatically generated

The two columns of labels were separated, and the reviews were cleaned.

A close up of a keyboard

Description automatically generated

A csv was exported.  Similarly, four separate corpuses were exported -- two for lie, two for sentiment. Alternative way was just to subset the clean data frame and export two csv files: one for lie and reviews and one for sentiment and reviews. The final  corpuses were exported and then re-imported. Count Vectorizer was used to create labeled data for modeling.

**Authenticity + Reviews**

A screen shot of a computer

Description automatically generated

**Sentiment + Reviews**

**A picture containing black, showing, remote, photo

Description automatically generated**

Data was examined before modeling.

**Fig 2**A screenshot of a cell phone

Description automatically generated

A word cloud was generated on the lie column for both truthful and untruthful reviews. Word similarity between the two groups indicate that probably some additional stopwords can be appended to the default nltk English stopwords. Namely, the words “restaurant”, “food”, and “place” could potentially be added to a custom stopword list.

|  |  |
| --- | --- |
| **A close up of a logo  Description automatically generated**  **Fig. 3 Wordclould on untruthful reviews** | **A picture containing food  Description automatically generated**  **Fig. 4 Wordclould on truthful reviews** |

Similar to earlier wordclouds on truthfulness (i.e. lie), the sentiment wordclouds indicate a familiar pattern. Specifically, the same custom stopword list could be created and utilized.

|  |  |
| --- | --- |
| **A picture containing food  Description automatically generated**  **Fig. 5 Wordclould on negative reviews** | **A close up of a logo  Description automatically generated**  **Fig. 6 Wordclould on positive reviews** |

**Models**

Multinomial Naïve Bayes algorithm was used to build models to classify the small corpuses and the customer review by sentiment(positive or negative) and by authenticity (true or fake, lie detection). Naive Bayes methods are a set of supervised learning algorithms based on applying Bayes’ theorem with the “naive” assumption of conditional independence between every pair of features given the value of the class variable.

**MultinomialNB** implements the naive Bayes algorithm for multinomially distributed data and is one of the two classic naive Bayes variants used in text classification (where the data are typically represented as word vector counts. tf-idf vectors are also known to work well in practice).

**BernoulliNB** implements the naive Bayes training and classification algorithms for data that is distributed according to multivariate Bernoulli distributions, i.e., there may be multiple features but each one is assumed to be a binary-valued (Bernoulli, boolean) variable. Therefore, this class requires samples to be represented as binary-valued feature vectors; if handed any other kind of data, a BernoulliNB instance may binarize its input (depending on the binarize parameter).

Three data frames were built (one for Bernoulli, one for normal frequency count and one for TfidfVectorizer (creates normalized dataframe) for Small corpuses (DOG and TRAVEL), Authenticity + Reviews and Sentiment + Reviews. MNB is used on the frequency data frames and on the tfidf. Bernoulli on the boolean.

Models for lie detection and sentiment classification were built. Support-vector machine model was added too for evaluation.

**Part 1: Small Corpuses**

The small corpus data has two folders: DOG and TRAVEL, each containing five text files. The text files have been combined and 3 new data frames were created using CountVectorizer and TfidfVectorizer.

**Normal DF Freq**

**A picture containing remote, black, photo, computer

Description automatically generated**

**BINARY DF**

A picture containing black, remote, photo, computer

Description automatically generated

**TFIDF DF**

A black sign with white text

Description automatically generated

The results from the models were very interesting.

**MNB BNB**

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | A screenshot of a cell phone  Description automatically generated |
| A black sign with white text  Description automatically generated | A screenshot of a cell phone  Description automatically generated |

The size of the data set here is very important. Due to the small amount Multinomial NB got 100% accuracy. Bernoulli`s model accuracy was only 67%. The results from TFDF were similar, depending on the train and test set split.

A screenshot of a cell phone

Description automatically generated A screenshot of a cell phone

Description automatically generated

The accuracy from this model is really subjective. The only purpose of them was to show the algorithms for building models using different vectorizers.

**Part 2: Authenticity + Reviews and Sentiment + Reviews**

After using the vectorizer for the big data frame, the end results are 6 small data frames. Two for Authenticity + Reviews and Sentiment + Reviews with Count Vectorizer, two for Authenticity + Reviews and Sentiment + Reviews with Count Vectorizer = Binary and two for TFIDF. This provide an opportunity to compare count, binary and tfidf normalized.

**MNB**

(**Authenticity + Reviews)**

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

(**Sentiment + Reviews)**

A screenshot of a cell phone

Description automatically generated

A picture containing outdoor, clock, meter, city

Description automatically generated

From the confusion matrix is obvious that sentiment was classified better than authenticity. That might be due to the huge subjectivity when the second category is classified. Generally, sentiment is easier to be determined and recognized by computers.

**BNB**

(**Authenticity + Reviews)** (**Sentiment + Reviews)**

|  |  |
| --- | --- |
| A screenshot of a computer  Description automatically generated | A screenshot of a cell phone  Description automatically generated |
| A screenshot of a cell phone  Description automatically generated | A screenshot of a cell phone  Description automatically generated |

The Bernoulli Naïve Bayes (BNB) was implemented. For the Sentiment, better result and accuracy were observed. It seems that Authenicity was classified better this time. The algoritim classified correcly more fasle reviews. Both ‘f’ and ‘t’ has higher precision. Multinomial NB cares about counts for multiple features that do occur, whereas Bernoulli NB cares about counts for a single feature that do occur and counts for the same feature that do *not* occur.

**TFIDF**

**(Authenticity + Reviews)** (**Sentiment + Reviews)**

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | A screenshot of a cell phone  Description automatically generated |
| A screenshot of a cell phone  Description automatically generated | A screenshot of a cell phone  Description automatically generated |

The results here are similar to the one with CV. Sentiment accuracy is higher. The precision of negative reviews is one, meaning that the model did good job predicting them. That can be seen in the ranking.

**FOR POS:**

**[(-3.8444729651374425, 'going'), (-4.011698321002088, 'salad'), (-4.112546588786621, 'amazing'), (-4.040658266817514, 'overall'), (-3.5927376400949385, 'favorite'), (-3.556209915732766, 'bad'), (-4.042430164239313, 'meal'), (-3.6611651489099795, 'prices'), (-3.781677678559012, 'glass'), (-4.0395795443013975, 'flavor')]**

**FOR NEG:**

**[(-5.205037269977528, 'table'), (-5.205037269977528, 'right'), (-5.205037269977528, 'staff'), (-5.205037269977528, 'asked'), (-5.205037269977528, 'delicious'), (-5.205037269977528, 'great'), (-5.205037269977528, 'recommend'), (-5.205037269977528, 'experience'), (-5.205037269977528, 'bring'), (-5.205037269977528, 'chinese')]**

**SVM**

(**Authenticity + Reviews)** (**Sentiment + Reviews)**

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated | A screenshot of a cell phone  Description automatically generated |
| A screenshot of a cell phone  Description automatically generated | A screenshot of a cell phone  Description automatically generated |

Lie detection accuracy in this model was high compared to the other two. SVM did better job with sentiment classification here.

**Results**

The Bernoulli Naïve Bayes (BNB) and SVM have the best accuracy for lie detection, followed by the Multinomial Naïve Bayes (MNB), then the TFIDF. In general the latter two models hovered at roughly 50% accuracy, indicating an inability to distinguish lies from reviews.

A Bernoulli Naïve Bayes, the SVM and Multinomial Naïve Bayes for sentiment analysis produces almost the same results. Unlike the lie detection., all models performed relatively well. Bernoulli had the highest accuracy.

**MNB**

|  |  |  |  |
| --- | --- | --- | --- |
| 75% acc | Sentiment + Reviews | 43% acc | Authenticity + Reviews |
| CORECT NEGATIVE | 10/28 | **CORECT FALSE** | 2/28 |
| CORRECT POSITIVE | 11/28 | **CORRECT TRUE** | 10/28 |

**BNB**

|  |  |  |  |
| --- | --- | --- | --- |
| 86% acc | Sentiment + Reviews | 54% acc | Authenticity + Reviews |
| CORECT NEGATIVE | 9/28 | **CORECT FALSE** | 6/28 |
| CORRECT POSITIVE | 15/28 | **CORRECT TRUE** | 8/28 |

**TFIDF**

|  |  |  |  |
| --- | --- | --- | --- |
| 71% acc | Sentiment + Reviews | 43% acc | Authenticity + Reviews |
| CORECT NEGATIVE | 9/28 | **CORECT FALSE** | 4/28 |
| CORRECT POSITIVE | 14/28 | **CORRECT TRUE** | 8/28 |

**SVM**

|  |  |  |  |
| --- | --- | --- | --- |
| 82% acc | Sentiment + Reviews | 54% acc | Authenticity + Reviews |
| CORECT NEGATIVE | 8/28 | **CORECT FALSE** | 7/28 |
| CORRECT POSITIVE | 12/28 | **CORRECT TRUE** | 8/28 |

As exploring MNB as main model for this paper, 20 most indicative words for each MNB model (lie detection and sentiment classification) were extracted.

**Lie detection**

|  |  |
| --- | --- |
| f 23.0 price  f 18.0 experience  f 12.0 right  f 12.0 order  f 11.0 asked  f 10.0 dishes  f 8.0 tofu  f 8.0 menu  f 8.0 little  f 8.0 friends  f 8.0 bread  f 7.0 try  f 7.0 southern  f 7.0 ordered  f 6.0 recommend  f 6.0 plate  f 6.0 meal  f 6.0 eve  f 6.0 great  f 6.0 food | t 33.0 price  t 28.0 experience  t 18.0 order  t 16.0 friends  t 12.0 try  t 12.0 glass  t 11.0 menu  t 11.0 asked  t 9.0 time  t 8.0 staff  t 8.0 right  t 8.0 bring  t 7.0 eve  t 7.0 fresh  t 7.0 food  t 7.0 dishes  t 6.0 wait  t 6.0 steak  t 6.0 pizza  t 6.0 need |

It is really hard for humans to distinguish the difference between lie and truth. None of the models performed good classifying authenticity. Based on those indicative words is hard to determine if the model have learned the concept, because human being can`t judge if single word is right or wrong without context. For better results, additional cleaning and preprocessing can be performed. The classifier can do better job is classify Parts of speech for example.

**Sentiment detection**

|  |  |
| --- | --- |
| n 31.0 experience  n 27.0 price  n 23.0 order  n 18.0 try  n 13.0 menu  n 13.0 little  n 9.0 meal  n 9.0 eve  n 9.0 dishes  n 9.0 bread  n 8.0 tea  n 8.0 table  n 8.0 right  n 8.0 friends  n 8.0 food  n 8.0 chinese  n 7.0 tofu  n 7.0 taste  n 7.0 staff  n 7.0 southern | p 32.0 price  p 29.0 experience  p 20.0 asked  p 17.0 glass  p 15.0 friends  p 10.0 right  p 10.0 order  p 10.0 eve  p 9.0 plate  p 8.0 try  p 8.0 dishes  p 8.0 15  p 7.0 menu  p 7.0 food  p 6.0 love  p 6.0 like  p 6.0 flavor  p 6.0 favorite  p 5.0 staff  p 5.0 restaurant |

Sentiment has a predictive value. Emotion classification gives a finer grained analysis of opinion, and more insight and explanation than traditional sentiment analysis. The MNB model learned the concept better in this case. Positive word like ‘love’, ‘like’, ‘favorite’ are detected successfully. On the negative side are ‘price’, ‘staff’, ‘little’ which are huge part of negative reviews.

If you give five people a printout of tweets and asked them to label them as  positive or negative, this would likely be an easier task than identifying a false review. That probably means that the “packets of meaning” that can convey sentiment (words,  sometimes word order) are smaller and more easily distinguishable in analysis. It is hard to train a computer something that you don`t even know or understand and find the right labeled data. Fake reviews can be created, but without knowing the motivation behind them is pointless.

Classifying reviews based on sentiment alone proved to be a fairly easy exercise. There is so much labeled data to learn from, letters, words and sentences can be classified and looked from many different angles (look at the “valence” of a word (using an  external dictionary, something like Vader ), look at the words that follow  negation words and look at all the words spread out together in a sparse matrix  and let the computer find patterns for itself).

**Conclusion**

Deception is a complex skill that takes part in human social interactions and can be achieved by different means. Machines have detected 5 key cues to spotting liars. But is that good enough? Sure, they don't have the intuition that humans do, but they are also not at risk of the kinds of mistakes that are specific to humans, such as emotional investment in wanting to think that a certain person is lying, or that another sort of person would never lie to them.

The promise of finding accurate computer-based lie-detection has gripped researchers, and dozens have gone on a quest to see if it works. There are computer programs written to find and count relevant linguistic cues in transcripts, and social scientists have used them to see if computers can find any reliable differences in the transcripts of communications known to be lies compared to the transcripts known to be truths. A perfect cue to deception would be one that occurs every time a person is lying, and never occurs when someone is telling the truth. The classic example is Pinocchio's nose. But the truth is, there is no Pinocchio's nose. It does not matter whether humans are looking for the cues or computers are: They just are not there.