Project Report [Sentiment Analysis using R]

The document captures the steps for performing sentiment analysis on Social media using text mining and machine learning using R and excel.

2016

Madhulika Joshi

9/21/2016

Contents

[Introduction to Objective 3](#_Toc464142455)

[Data used for analysis 4](#_Toc464142456)

[Data Formatting 4](#_Toc464142457)

[Using opinion lexicon to estimate sentiment 5](#_Toc464142458)

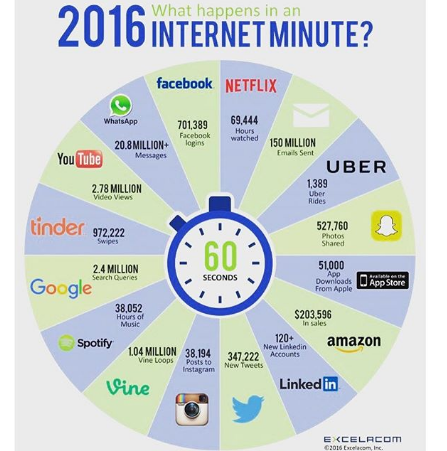
[Using sentiment package to perform sentiment analysis 8](#_Toc464142459)

[Supervised learning methods/ML to predict sentiments 15](#_Toc464142460)

[Conclusion 16](#_Toc464142461)

# Introduction to Objective

In today’s internet savvy world, Social media has reached a new level of power and usage. Statistics says that each internet minute sees a lot of action happening worldwide (pic attached). These numbers are continuously on rise.



In recent times, the popularity of social media has grown exponentially and is increasingly being used as a channel for mass communication, such that the brands consider it as a medium of promotion and people largely use it for content sharing. With the increase in the number of users online, the data generated has increased many folds, bringing in the huge scope for gaining insights into the untapped gold mine, the social media data.

Social media spans lots of Internet-based platforms that facilitate human emotions such as:

Networking, for example, Facebook, LinkedIn, and so on

Micro blogging, for example, Twitter, Tumblr, and so on

Photo sharing, for example, Instagram, Flickr, and so on

Video sharing, for example, YouTube, Vimeo, and so on

Stack exchanging, for example, Stack Overflow, Github, and so on

Instant messaging, for example, Whatsapp, Hike, and so on

It is important to understand all the aspects of social media today because real people/customers are using it. The real sentiment of the customers, users reflect via the various posts, reviews, tweets and that is the objective of the project. It is based around performing sentiment analysis on the reviews posted by real customers on 3 famous sites- Amazon, Imdb and Yelp. The purpose is to identify sentiments through various machine learning algorithms and text mining. The results can be used to :-

1. Understand actual performance of a product.
2. Focus on marketing strengths through positive sentiments.
3. Identify Negative sentiments and actions can be taken to improve required areas.
4. Identifying preferred trends and expectations and can work like real time CRM.

# Data used for analysis

The data being used is an already labelled data set available at UCI Machine Learning Repository. Link below-

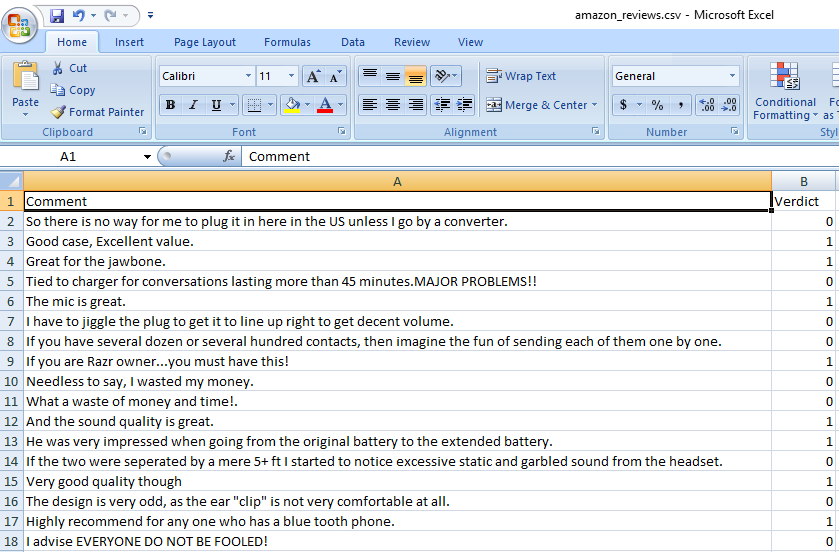
[https://archive.ics.uci.edu/ml/datasets/Sentiment+Labelled+Sentences#](https://archive.ics.uci.edu/ml/datasets/Sentiment+Labelled+Sentences)

This data set contains statements from three different websites ([imdb.com](http://imdb.com/), [amazon.com](http://amazon.com/), and [yelp.com](http://yelp.com/)) dealing with reviews of movies, products, and restaurants, respectively. There are several thousand reviews at each site with about 500 of them at random at each site marked positive indicating the underlying sentence sentiment to be positive. Similarly there are about 500 sentences selected at random from each site marked negative. According to the contributors of the data set, the positive or negative labels were assigned manually to sentences with clear non-neutral sentiments. So, this data can be very well used to run supervised learning methods also.

# Data Formatting

After downloading and unzipping the data, I read it into an Excel file. Data corresponding to each website was read into a separate sheet. Next, I filtered the data to keep only labelled sentences and created amazon\_reviews.csv, imdb\_reviews.csv and yelp\_reviews.csv. A header row has been added to all the files.

A screenshot of an Excel sheet after filtering is shown below. A zero/one in column two implies negative/positive sentiment for the sentence in column one.



These csv files were read into different data frames to kick start the analysis.

# Using opinion lexicon to estimate sentiment

This method uses a simple algorithm which gives a score based on the number of times a positive or a negative word occurred in the given sentence. A predefined set of positive and negative words is used in this method.

The downloaded file contains positive and negative opinion/sentiment (nearly 68, 000) words from English language. This opinion lexicon will be used as a first example in our sentiment analysis experiment. Here are a few examples of existing positive and negative sentiment words:

1. **Positive**: Love, best, cool, great, good, and amazing
2. **Negative**: Hate, worst, sucks, awful, and nightmare

**Ref- negative\_words.txt, posisitve\_words.txt**

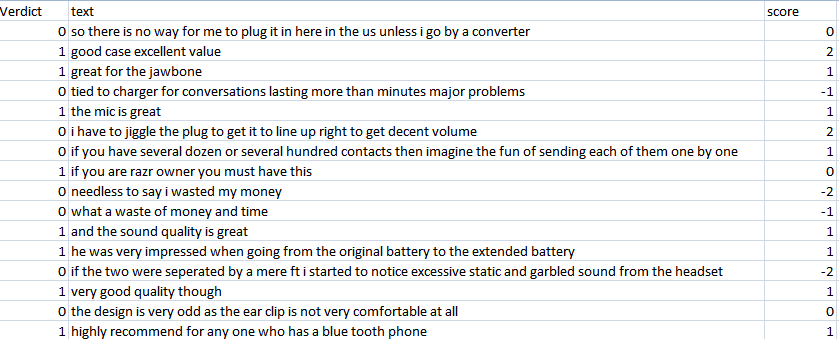
**The sentiment score of a statement = Number of positive words – Number of negative words**

**Function created – getSentimetScore (sentiment\_analysis.R)**

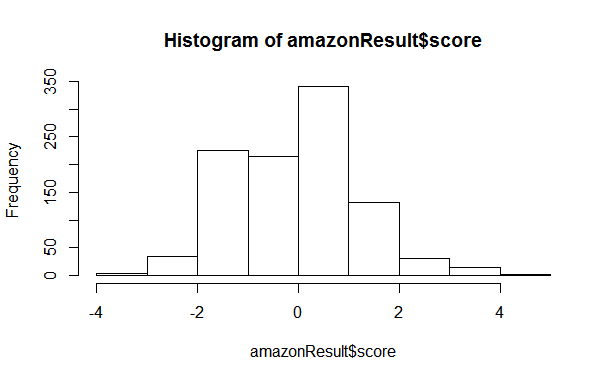
A positive score means the statement is positive, a negative score means it is negative and zero means the statement is neutral. The input reviews have been selected to either communicate a positive or negative feeling. This method successfully identified strong positive/negative words but not all were correct.

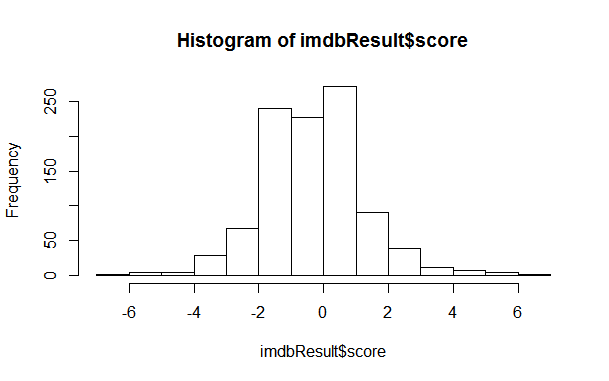
**Results captured – amazon\_sentiment\_score.csv, imdb\_sentiment\_score.csv, yelp\_sentiment\_score.csv**

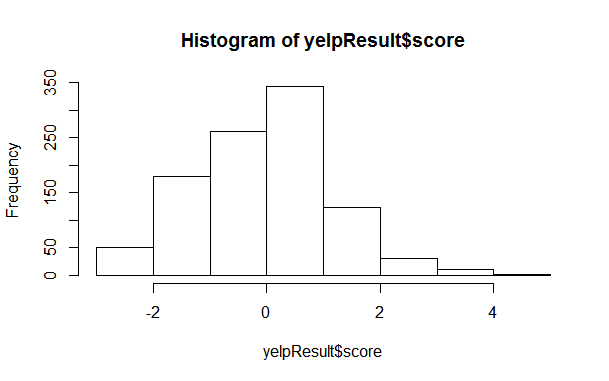
**Few of the reviews from amazon and their scores are given below.**

****

**Histograms for all the scores show a uniform distribution for positive and negative scores which was expected. For all the zero scores, the statement was assumed neutral by the algorithm.**





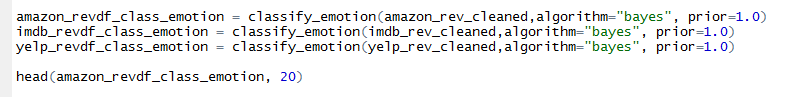
****

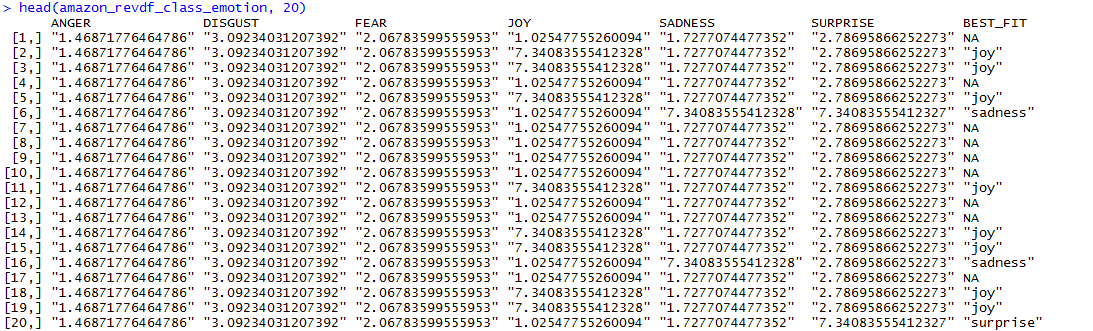
**RESULT - Accuracy achieved using this method is close to 60% and as it is entirely using words to make a decision it may be helpful to read overall sentiment from a set of sentences. This can be used in twitter analysis to gather general sentiments about any topic, trend, product, person or anything in question.**

# Using sentiment package to perform sentiment analysis

As a first step, we need to have the sentiment and Rstem packages installed in our R workspace, after that we can build the bayes classifier for sentiment analysis of entire dataset. Two functions which are used for analysis-

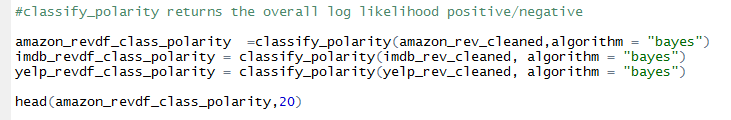
1. Classify\_emotion (sentiment\_analysis.R)- The sentiment package was built to use a trained dataset of emotion words (nearly 1,500 words). The function classify\_emotion() generates results belonging to one of the following six emotions: **anger, disgust, fear, joy, sadness, and surprise**. When the system is not able to classify the overall emotion as any of the six, NA is returned as BEST\_FIT:

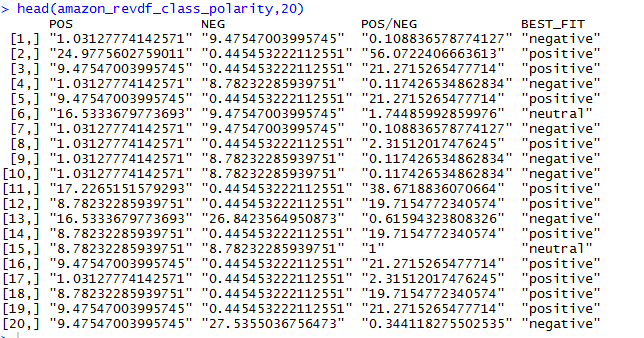
****

****

The above snapshot shows how for each statement emotions are classified using bayes algorithm. Next step was to take out BEST\_FIT as final emotion and substitute ‘NA’ as “unknown”.

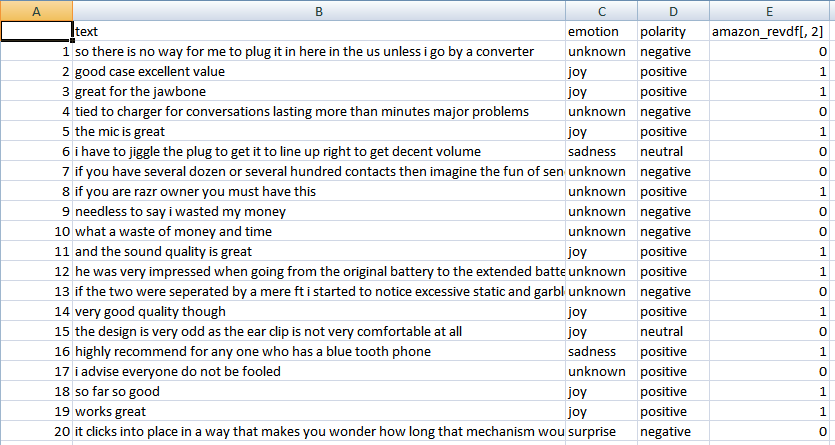
1. Classify\_polarity(sentiment\_analysis.R) – Similar to the above function, this predicts the polarity of the statement whether positive or negative.



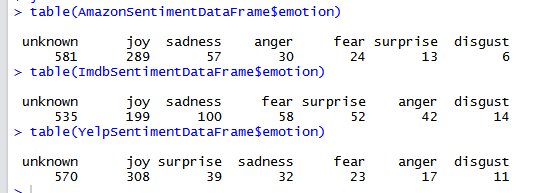


The final outcome has been written to csv files-

Amazon\_sentiment\_final.csv, Imdb\_sentiment\_final.csv, Yelp\_sentiment\_final.csv

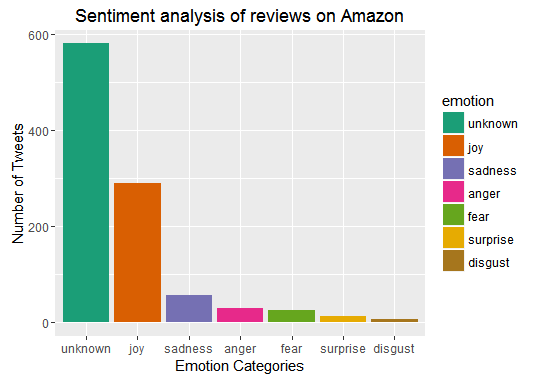


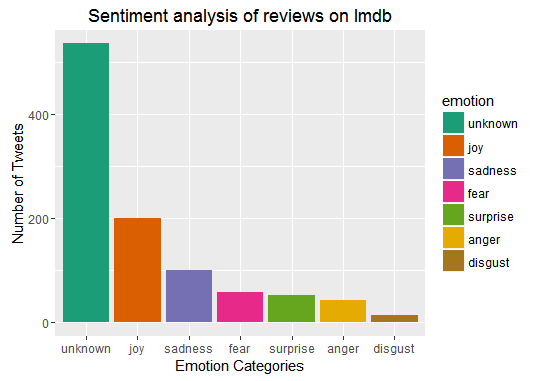
In the above screenshot, the first five rows correctly identify the sentiment polarity.

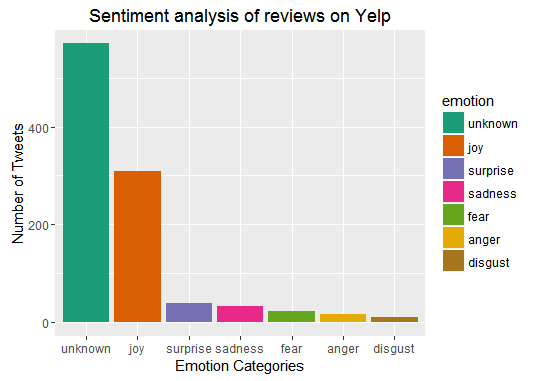


**The emotions in a sorted manner can be identified in the above screen shot. More than 50% statements were not identified with any emotion. However, the function identified close to 50% statements into emotions.**

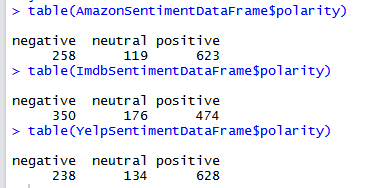
**The sentiments can be plotted into a graph using function-**

****

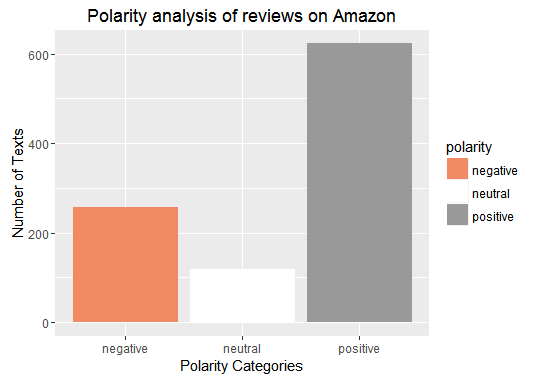
****

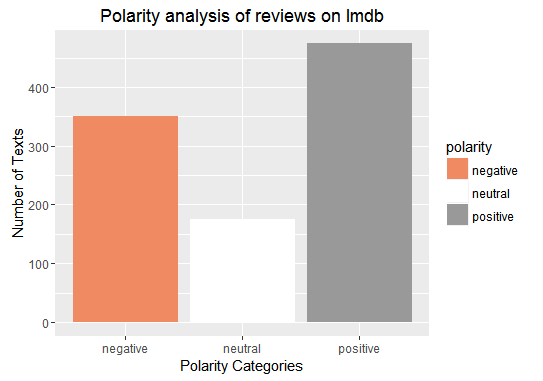
****

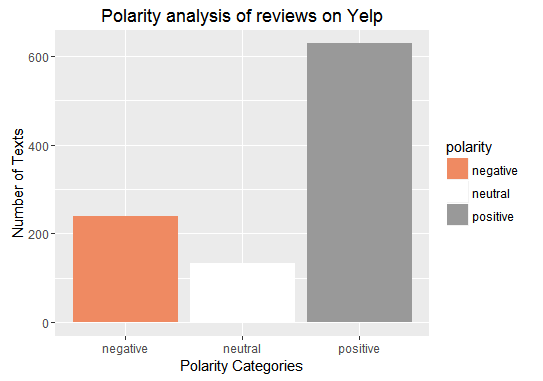
**Similarly for the polarity, we can draw visuals to identify the statistics-**

****

**Function to draw the graph-**

****

****

****

|  |  |  |  |
| --- | --- | --- | --- |
| **File** | **True positives** | **True negatives** | **%accuracy** |
| Amazon\_sentiment\_final | 416 | 222 | 0.638 |
| Imdb\_sentiment\_final | 339 | 277 | 0.616 |
| Yelp\_sentiment\_final | 407 | 203 | 0.61 |

**RESULT- Greater than 60% accuracy in identifying positive/negative polarity in all datasets.**

**Word clouds are wonderful in identifying sentiments at a higher level. Below are the word clouds for the given datasets. GetWordCloud function has been created in sentiment\_analysis.R for this.**

1. **For Amazon reviews-**

****

1. **For Imdb reviews-**

****

1. **For Yelp reviews-**

****

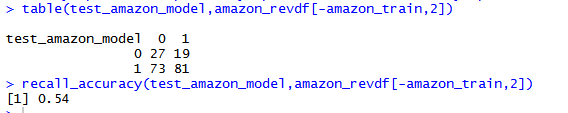
# Supervised learning methods/ML to predict sentiments

**For these methods, we are dividing the datasets into 80:20 for training and test data. To distribute the data randomly, we are using createDataPartition function from package Caret ( Sentiment\_analysis2.R)**

**Steps-**

1. **Create Document term matrix for the distributed data**
2. **Create model using naiveBayes function in package e1071 using training 800 records.**
3. **Test the model using training model on 200 test records.**
4. **Find confusion matrix and accuracy**
5. **Create a csv file for the final output (amazon\_naive.csv).**

**RESULT- For amazon data, we got following results- 54% accuracy on 200 statements.**

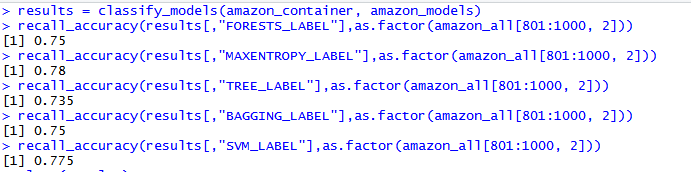
****

**Similar to this, we can use other Machine learning algorithms ( used only amazon data)**

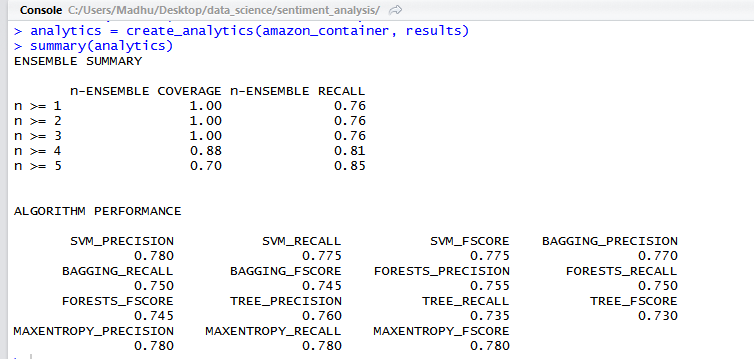
**Steps-**

1. **Create Document term matrix.**
2. **Create container using RTextTools package.**
3. **Train models using various algorithms- "MAXENT" , "SVM", "RF", "BAGGING", "TREE".**
4. **Classify models into results using classify\_models() function.**
5. **Store results into csv file (amazon\_ml\_results.csv)**
6. **Check accuracy for various algorithms.**

The following screen shot captures the performance of the algos-



Clearly, these algos are performing better than naïve bayes in predicting the sentiment output.



# Conclusion

Sentiment analysis can be done in many ways depending on the requirements. Sentiment package offers a lot of utilities which classify emotions and polarity clearly in most of the cases. Like many other supervised learning methods, it has its limitations. Using RTextTools package, other machine learning algorithms can be used to perform classification/prediction of sentiments and that offers more than 75% accuracy in our case.