SENTIMENT ANALYIS & DEEP LEARNING ON AMAZON PRODUCT REVIEWS

Kiran Talreja
Bashaer Alkhathlan
Chandni Sharma
Abhijit Shinde

Table of Contents

Contents

Table of Figures

Problem

- Applying sentiment analysis to scraped reviews from the web using R
- Label positive/negative words from each review of scraped corpus
- Vectorize each review in corpus using Doc2Vec and apply both DBN/RBM and SVM to compare performances

Analysis

- Amazon Reviews **dataset** consists of the columns:
- Text,
- Score,
- Emotion,
- and Polarity (Positive, Negative, and Neutral) for a 1000+ reviews.
- **Text column:** contains all scraped reviews of an Amazon product
- **Score column:** contains the scores of the positive/negative words in each review. Scores are calculated by using the function score.sentiment that takes sentences from product reviews and compare it to positive/negative wordlist dictionaries
- **Emotion column:** classify_emotion function has been used from the Sentiment package to classify the emotion (e.g. anger, disgust, fear, joy, sadness, surprise, or unknown) of a set of reviews using a naive Bayes classifier

- **Polarity column:** classify_polarity function has been used from the Sentiment package to classify the polarity (e.g. positive, neutral, and negative) of a set of reviews using the algorithm "bayes"

Scraping Reviews Process

- We have created a scraper method in which using SelectorGadget tool in the browser, we figured out what selectors were used for which part of the web page. And as we are using reviews of the products so the comments of the users and their ratings were to be considered. For that, we took the url of that page and the product code. Then, considering the pages we want to read from, used the 'read_html()' function to read all the comments of the users.
- For that, using html_nodes() method we selected what css selector had what type of data like the author, title of the comment, etc and Converting that into html_text.

Methods:

Sentimental Analysis

- Score.sentiment function has been created for this part, it basically takes each sentence, positive words, and negative words from each product review and then process them to calculate and return Scores data frame. This function first cleans sentences using gsub() function and then applies tolower() function to force sentences to appear in a lower case format. Then, the function splits sentences into words using the function str_split() and then it unlists words to be compared and matched with dictionaries of positive/negative words. Then finally, and to calculate the scores of all reviews, the technique: score = sum(pos.matches) - sum(neg.matches) is used. True-false will be treated as 1/0 by the function sum() in this case.

- To plot the scores of reviews, we took all the product reviews and prepared them for plotting by removing any word punctuation and empty spaces between words. A function to handle any errors was also applied, it can handle errors such as dealing with missing values.
- To calculate emotion within reviews, classify_emotion function was used from the Sentiment package to classify the emotion (e.g. anger, disgust, fear, joy, sadness, surprise, or unknown) of a set of reviews using a naive Bayes classifier with prior=1.0
- To calculate polarity within reviews, classify_polarity function has been used from the Sentiment package to classify the polarity (e.g. positive, neutral, and negative) of a set of reviews using the algorithm "bayes"
- The results of both classifications were placed inside a data frame to be further plotted. For plotting, ggplot() function was used to give a clear representation of the sentiment analysis process that was applied on 1000+ Amazon product reviews

Plots and Tables

	score	text
1	23	I'll preface this by saying that I own an iPad, and was n...
2	45	If you read my Fire TV review you know that I am toug...
3	23	For the low price, this tablet really does more than ex...
4	1	My 95 father announced, "I want a Kindle for Christmas...
5	4	I'm starting off my saying I'm not just anti-kindle.I've ow...
6	26	I was very surprised to find that I absolutely love this li...
7	4	First, let me say that I love Amazon and feel it truly pro...
8	1	It's fine, just like my last one that was stolen. I need to...
9	0	Ordered this for our 8 year old to replace her first gen...
10	11	Incredible little machine. Ads on the main pages, but th...
11	8	I have had the tablet for a week. I am a member of the ...
12	-1	I bought this for my son 10 year old for Christmas. He l...
13	10	i love this tablet. i am a tablet geek -- don't know how ...
14	0	was everything and more then what I expected
15	4	For the price, you cannot go wrong with this tablet! It ...
16	0	Warning if you have a limited data plan. This device sho...
17	12	If you love Kindles you will love the 6. Three of us in m...
18	12	Let me stress that for the feature I enjoy most you MU...
19	2	I bought them for my boys. It's a perfect size for them....
20	14	I have a Kindle Fire 8.9" 1st generation and I love it. I di...
21	1	I would recommend this for anyone
22	3	I bought this tablet for my 8 year old son. This past ye...
23	5	I love this tablet. So far it does everything I need it to ...
24	11	Love love love. Coming from an Apple fan, owner of t...

Figure 1 Few Reviews.scores for 1000+ Reviews

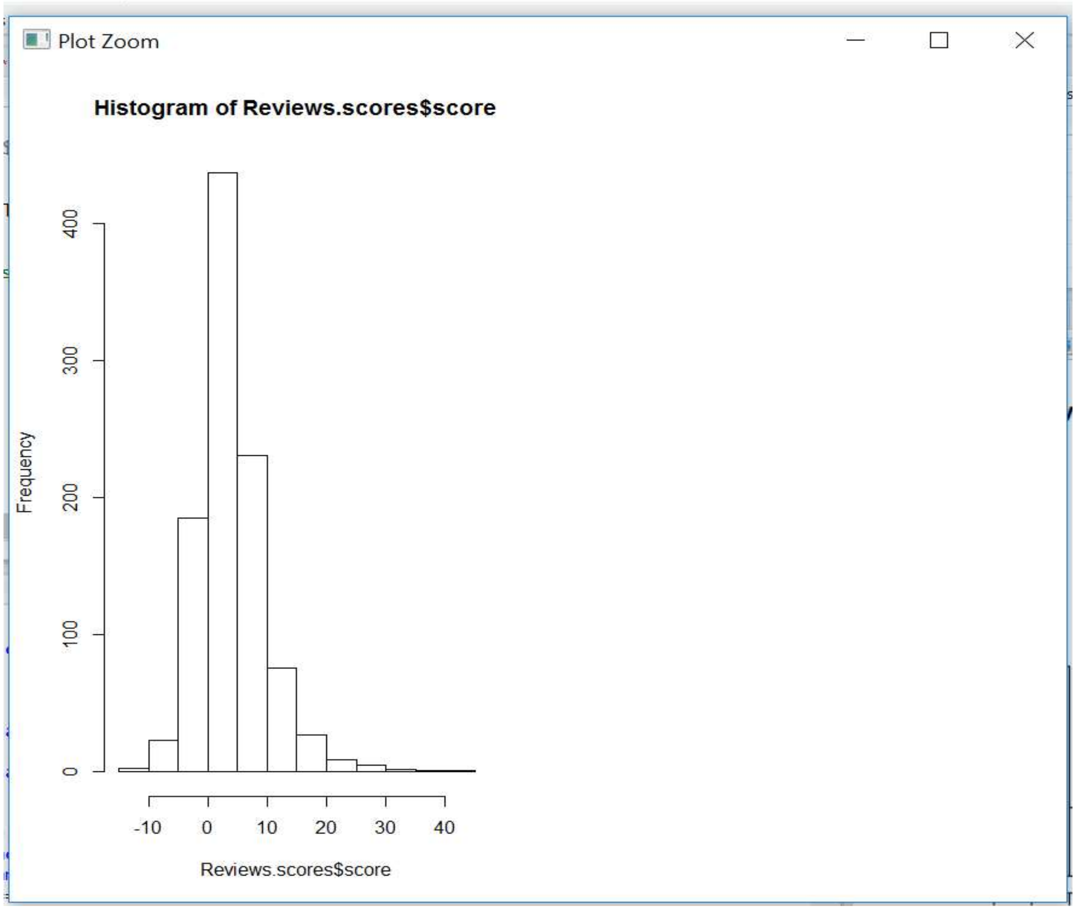


Figure 2 Histogram for Reviews and Their Scores (Reviews.scores) for 1000+ Review

	text	emotion	polarity
1	ill preface this by saying that i own an ipad and was no...	joy	positive
2	if you read my fire tv review you know that i am tough ...	joy	positive
3	for the low price this tablet really does more than exp...	joy	positive
4	myfather announced i want a kindle for christmasno o...	joy	neutral
5	im starting off my saying im not just antikindleive own...	surprise	negative
6	i was very surprised to find that i absolutely love this li...	joy	neutral
7	first let me say that i love amazon and feel it truly provi...	joy	positive
8	its fine just like my last one that was stolen i need to f...	joy	negative
9	ordered this for ouryear old to replace her first genera...	sadness	negative
10	incredible little machineads on the main pages but the...	anger	neutral
11	i have had the tablet for a week i am a member of the ...	joy	neutral
12	i bought this for my sonyear old for christmas he love...	joy	neutral
13	i love this tableti am a tablet geekdont know how i go...	joy	positive
14	was everything and more then what i expected	NA	positive
15	for the price you cannot go wrong with this tablet it a...	joy	neutral
16	warning if you have a limited data plan this device sho...	sadness	neutral
17	if you love kindles you will love thethree of us in my h...	joy	positive
18	let me stress that for the feature i enjoy most you mu...	surprise	positive
19	i bought them for my boys its a perfect size for them t...	anger	positive
20	i have a kindle firest generation and i love it i didnt nee...	joy	positive
21	i would recommend this for anyone	NA	positive
22	i bought this tablet for myyear old son this past year i ...	joy	positive

Showing 1 to 23 of 1,000 entries

Figure 3 Calculating Emotion and Polarity for Each Review

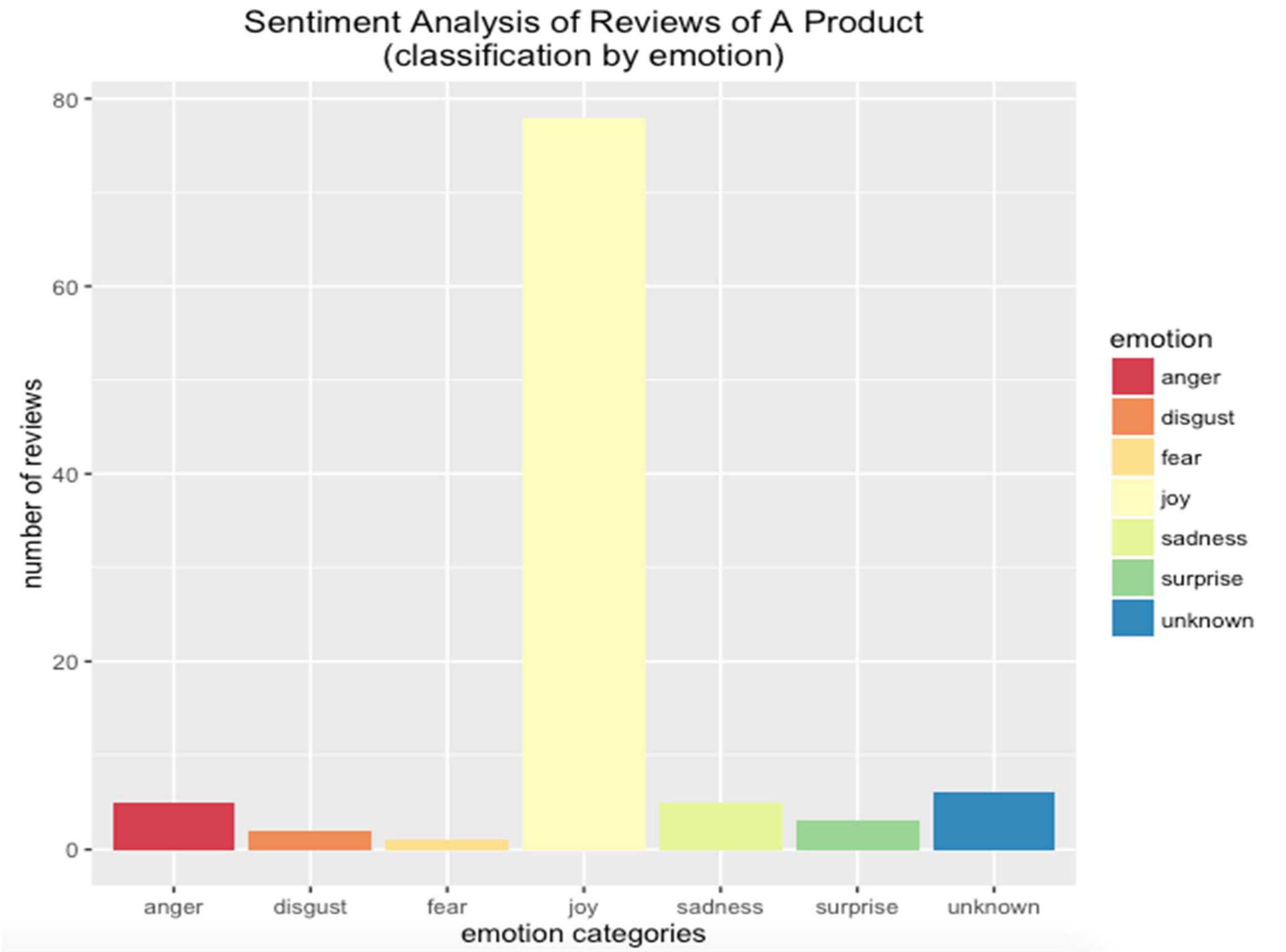


Figure 4 Ggplot of Reviews Classified by Emotion

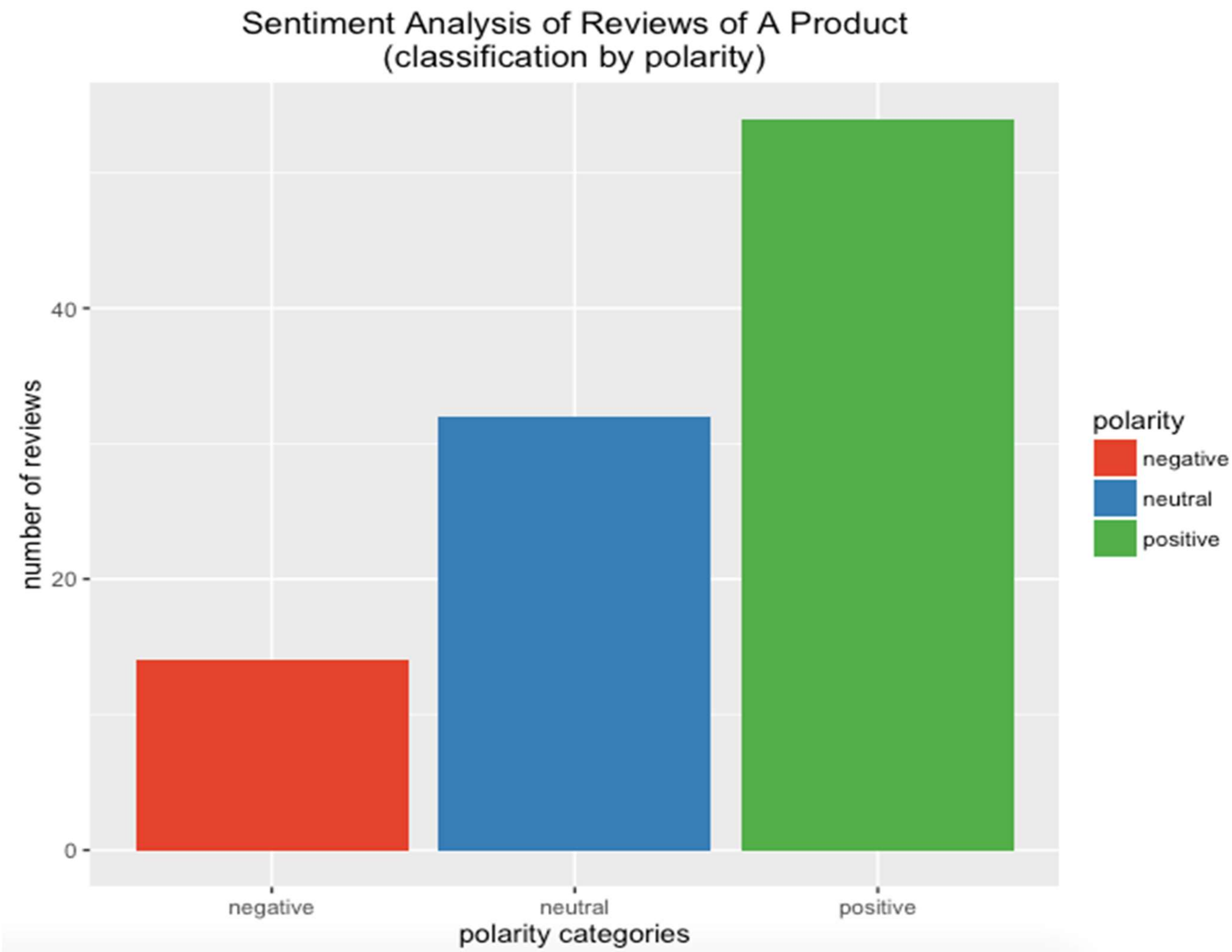
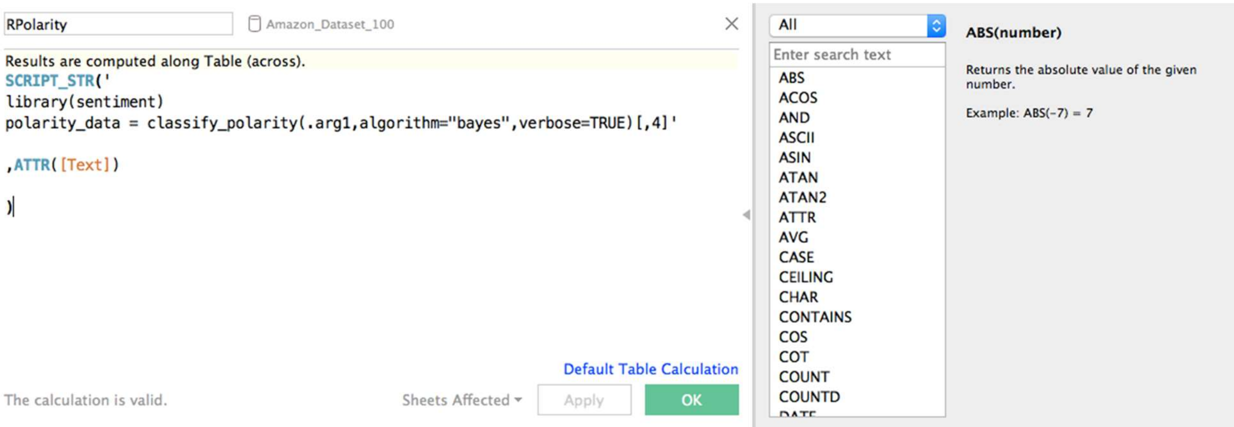
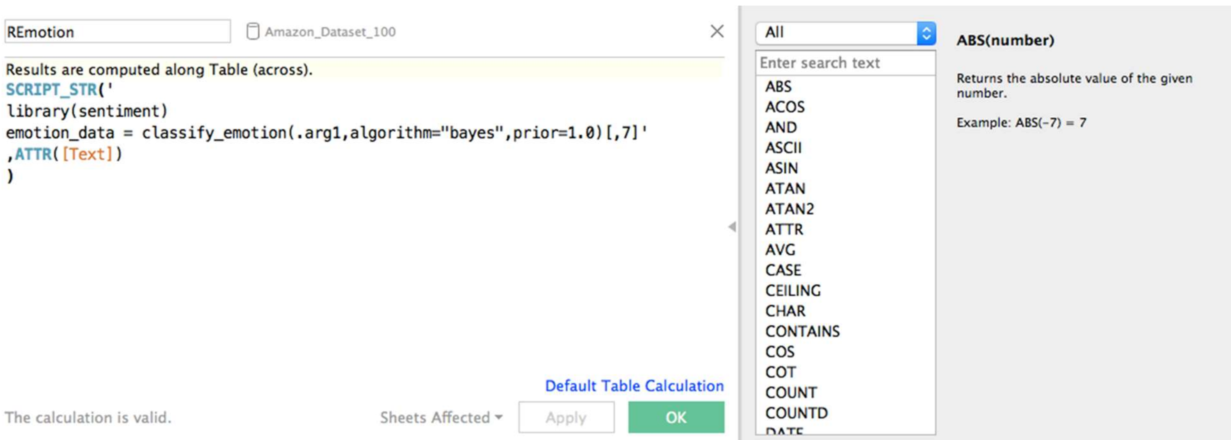


Figure 5 Ggplot of Reviews Classified by Polarity

Sentimental Analysis in Tableau

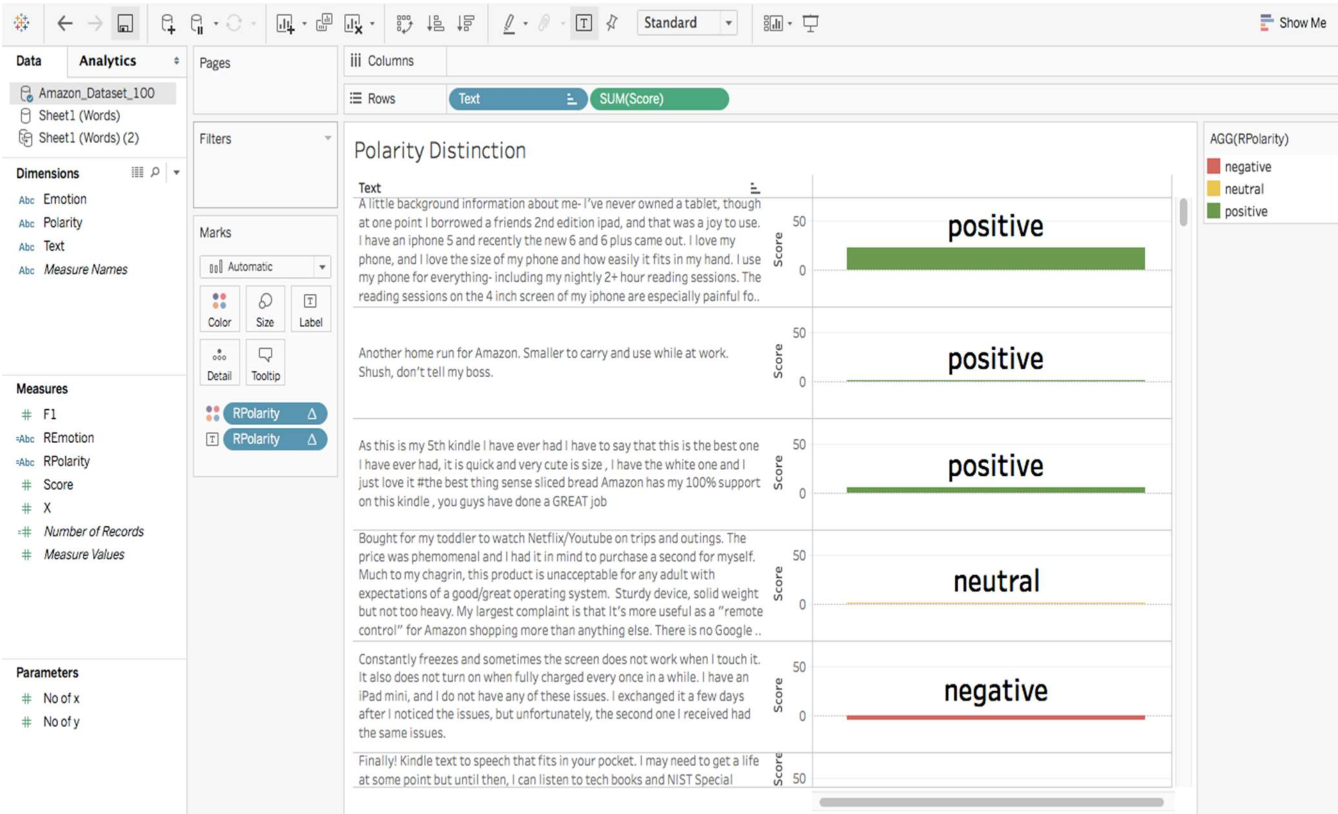
In Tableau, we uploaded the csv file containing all the reviews of the customers. Also, we did **Integration of R with Tableau** by using scripts to do sentimental analysis as follows:





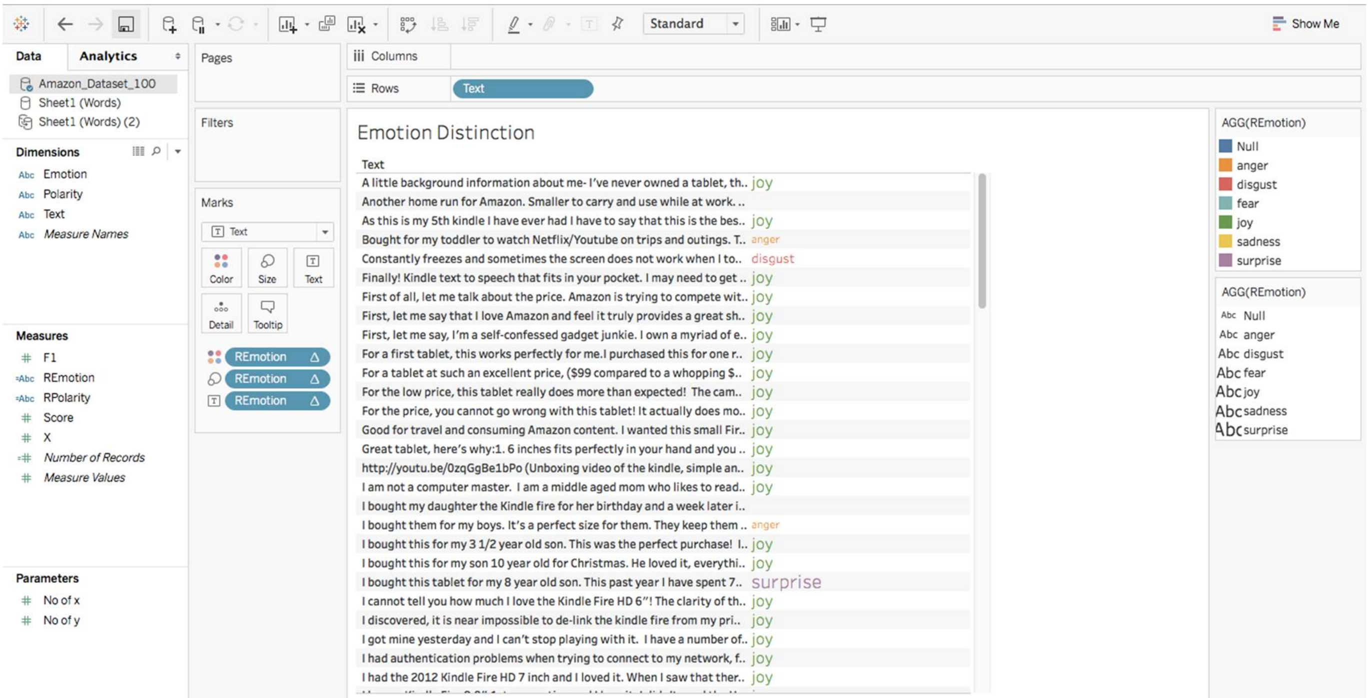
First we created the connection between R and Tableau by using the library '**RServe**' in R and then connecting tableau to that server. In REmotion and RPolarity calculated fields, I used the SCRIPT_STR function to classify emotion or polarity of the 'sentiment' library along with using the attribute from the csv file, 'Text'. This produced the same results as in R scripting.

Polarity Distinction:



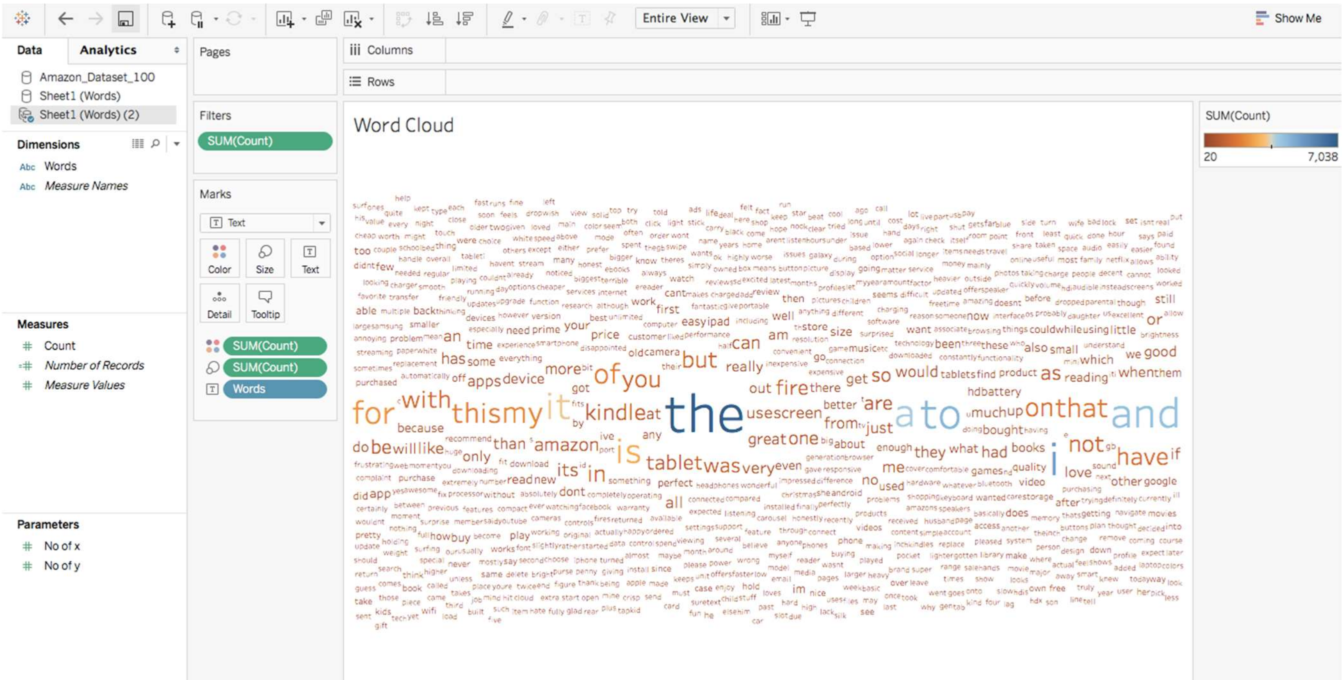
in this, we have used **RPolarity** calculated field to show the label of Polarity along with what color it can be associated with. We have also used score to show the relative scoring of those reviews.

Emotions Distinction:



In this, we have used **REmotion** calculated field to show the label of Emotion along with color. Also, the size is relative, meaning if that emotion is more, then that emotion word is shown bigger in size.

Word Cloud:



This is a Word Cloud with the count of all the words used in all the reviews. We have set the filter to show only those words which has more than 20 count. For this, we used another excel file containing only words and their counts. This file was generated using a **split_text_tool**.

Doc2Vec

We have built Text2Vec for our **Documents to Vector conversion**. The main package used is **text2vec** in R and especially provides tools for Text Analysis & NLP.

Vocabulary based Vectorization:

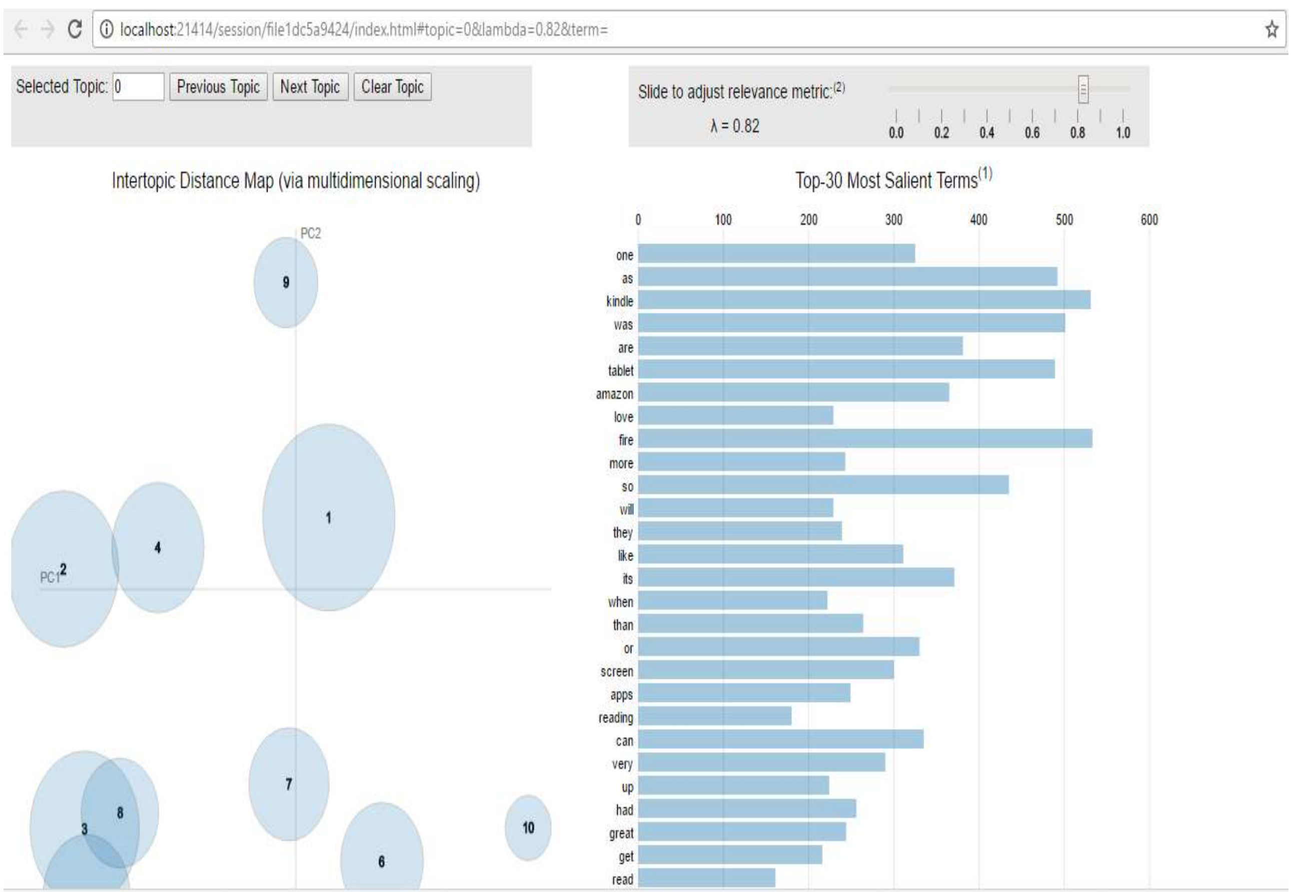
- Fast text vectorization using Vocabulary.
- We insert documents into Corpus which are nothing but C++ classes.
- Then we created DTM (Document Term Matrix) from the input documents.
- In DTM, we vectorize the text by creating mappings.
- Copy-on-modify semantics involves reading the documents in RAM but text2vec is far better.

Tuning Text2Vec

- We further Pruned the Vocabulary by removing the STOP WORDS (useless words) for better performance.
- We then apply **TF-IDF (Term Frequency – Inverse Document Frequency) transformation**, which increased the weight for document-specific terms and decrease weight for widely used terms.
- We observed that the columns reduced after performing Pruning since we removed many of the unnecessary words and it was helpful for our further modelling.

LDA (Latent Dirichlet Allocation)

- LDA is a statistical model which helps in telling the observations of the words like Salient terms (most noticeable or important words)
- The plot runs on the Web browser locally connected from the Rserver and shows the Top 30 Most Salient Terms in the dataset with an adjustable relevance metric (Lambda)



DBN/DNN

We used the Darch package of R to perform Deep Belief Network (DBN/DNN).

Following are the steps:

1. Divided Input and output variables
2. Using **darch** function, we created the DBN/DNN model
3. The parameters for above function has layers with 1026 inputs and 1 outputs and 4 layers
4. Following are the steps which are performed in Darch
   - Create and configure a Darch instance.
   - Pre-train the network.
   - Fine-tune the network.
   - Back-propagate further data through the network to create predictions and Retuning.

DBN/DNN Performance Results: (Confusion Matrix)

```
> caret::confusionMatrix(data=predictions,
+                 testingoutput1$polarity,
+                 positive='yes')
Confusion Matrix and Statistics

          Reference
Prediction negative neutral positive
  negative        0       0        0
  neutral         0       0        0
  positive       37     108      155

Overall Statistics

               Accuracy : 0.5167
                 95% CI : (0.4585, 0.5745)
    No Information Rate : 0.5167
    P-Value [Acc > NIR] : 0.5233

                  Kappa : 0
 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: negative Class: neutral Class: positive
Sensitivity                   0.0000           0.00          1.0000
Specificity                   1.0000           1.00          0.0000
Pos Pred Value                   NaN            NaN          0.5167
Neg Pred Value                0.8767           0.64             NaN
Prevalence                    0.1233           0.36          0.5167
Detection Rate                0.0000           0.00          0.5167
Detection Prevalence          0.0000           0.00          1.0000
Balanced Accuracy             0.5000           0.50          0.5000
>
```

SVM

- SVM is a classification technique that seeks to find a hyperplane that partitions the data by their class labels
- It avoids over-fitting the data by maximizing the margin of the separating hyperplane.
- SVMs are useful in Text & Hypertext Classification.

- SVM is known to significantly reduce the need for labels in the training instances.

Further, we calculated the Best Parameters for tuning the SVM (below results):

```
> svm_tune <- tune(svm, polarity ~ .,data = training, scale = F, kernel="radial",
+                  ranges = list(gamma = seq(0,0.2,0.01), cost=10^(-1:1)))
> print(svm_tune)

Parameter tuning of 'svm':

- sampling method: 10-fold cross validation

- best parameters:
 gamma cost
  0.01   10

- best performance: 0.3914286

>
```

And then we tuned the SVM to get better accuracy.

SVM Performance Results: (Confusion Matrix)

```
> caret::confusionMatrix(data=predictions,
+ testing$polarity,
+ positive='yes')
Confusion Matrix and Statistics

          Reference
Prediction negative neutral positive
  negative       10       4        1
  neutral        16      48       36
  positive       11      56      118

Overall Statistics

               Accuracy : 0.5867
                 95% CI : (0.5286, 0.643)
    No Information Rate : 0.5167
    P-Value [Acc > NIR] : 0.0087862

                  Kappa : 0.2556
 Mcnemar's Test P-Value : 0.0001796

Statistics by Class:

                     Class: negative Class: neutral Class: positive
Sensitivity                  0.27027         0.4444          0.7613
Specificity                  0.98099         0.7292          0.5379
Pos Pred Value               0.66667         0.4800          0.6378
Neg Pred Value               0.90526         0.7000          0.6783
Prevalence                   0.12333         0.3600          0.5167
Detection Rate               0.03333         0.1600          0.3933
Detection Prevalence         0.05000         0.3333          0.6167
Balanced Accuracy            0.62563         0.5868          0.6496
> 
```

Performance Comparison between Models (Evaluations)

We have evaluated the accuracy of the models based on the Confusion Matrix.

Interestingly, the tuned SVM model performed better with the accuracy of 59% as compared to the Deep Belief Network's accuracy of 52 %.

Model	Accuracy
DNN	52%
SVM	50%
Tuned SVM	59%

References

https://cran.r-project.org

https://en.wikipedia.org/wiki

https://iralab.unt.edu/sentiment-analysis

https://boraberan.wordpress.com/2013/12/24/sentiment-analysis-in-tableau-with-r/

http://www.theinformationlab.co.uk/2016/03/02/text-analysis-101-sentiment-analysis-in-tableau-r/

http://alexloth.com/2016/01/31/how-to-implement-sentiment-analysis-in-tableau-using-r/

http://www.clearlyandsimply.com/clearly_and_simply/2015/03/word-clouds-with-tableau.html