SENTIMENT ANALYSIS ON TWITTER DATA

L545
Spring 2017

Saheli Saha, Nirad Ranjan Parhi

ABSTRACT

Sentiment Analysis is a technique employed in the field of Computer Science and linguistics to automatically detect sentiment contained in a piece of text. Using this technique, we can get a general sense of what is being said about a particular topic or subject. The sentiment contained in text is generally categorized as positive, negative and neutral. If we dive deeper into analyzing the text, we can find even subtler sentiments like sarcasm, mood, intensity. If we want to get some insights into which words are occurring together most frequently, a technique called Word Co-occurrence Matrix is employed. Along with sentiments extracted from text analysis, word co-occurrence gives us deeper insights into what users are talking about on Twitter, related to a particular topic of interest.

Index Terms - Sentiment Analysis, Word Co-occurrence Matrix

1. INTRODUCTION

Sentiment Analysis of text is a vast field which has been frequently explored. Our aim in this project is to perform Sentiment Analysis using our own code implementations. We plan to implement most of the required functionalities in Python from scratch, so that we get a better understanding of all the steps involved in performing sentiment analysis on text.

2. DESCRIPTION OF DATA

Our dataset is taken from a Twitter dump related to the Financial Services company Fintech. Fintech or Financial Technology is a conglomerate of companies consisting of both startups and established financial and technology companies trying to replace or enhance the usage of financial services of incumbent companies. We have taken the Tweeter dump of tweets of people talking about the said company, scraped the dataset and stored it in a CSV file.
Our data file is scrapped from Twitter. We converted it into a csv file. Computer programs and other technologies used to support or enable banking and financial services.

This dataset has seven features: Topic, Tweet, User, Date, Description, Location, stakeholders.

Topic: - While scrapping the data we only considered data related to Fintech so for this column the only value is Fintech. So, we have discarded this column while cleaning the data.
Tweet: - This is the column where all tweets are contained. We have analyzed this column.
User: - Corresponding user for a given tweet.
Date: -Date when the tweet is posted.
Description: - Description of the user.
Location: - Location of the user.
Stakeholders: - People from different groups. Like – Academia, Business.

Currently we have analyzed the tweet column. As a future scope we can analyze the other variables like – description, stakeholder, location to understand people from which location and which group of people are talking about Fintech.

Following is a small snapshot of the dataset that we are going to use in our project:

#bitcoin #fintech Fedcoin: The U.S. Will Issue E-Currency That You Will Use https://t.co/iblPXUPPgfŠ—_ https://t.co/lcuq3qBbGe
RT @guzmand: No one innovation is a silver bullet for #banking: 5 ways they're adapting & innovating #fintech #mobile #paymentsŠ—_
How Smart Contracts Are Changing Financial Services Financial Services Technology #fintech https://t.co/ECwRAp2KiF https://t.co/Ow68aKNGd6
Satoshium Project Announced & White Paper Released https://t.co/WBbt8aPANZ White Paper https://t.co/UwpYjIxtq9 #bitcoin #fintech #goldcoins
RT @sdubois: NYSE to allow all US listed securities to trade on its floor #fintech @NYSE https://t.co/iq3Hosir2G
How collaboration in the #fintech industry can unlock #digital #growth @wef https://t.co/M92ZxVCoBt https://t.co/feOe3ck5ZE
Banking Technology: Belgium gets sweet on London #fintech... thanks to InnFin and bhive_eu: Š—_ https://t.co/pgoOLvIVWM

RT @neiljpearce: The latest The CIO's Daily! https://t.co/3AlXAji15t Thanks to @CarlosGarriga @PrestonW @jeremychobbs #fintech #iot
Company announcement: Token partners with VirtusaPolaris https://t.co/B64e7Tm7ys #fintech
#fintech Australian Fintech, Forecast for 2020 - Research and Markets - Yahoo Finance https://t.co/hFBAOF8hke
Australian #fintech, Forecast for 2020 - Research and Markets https://t.co/KEfOnVpMUu
A look at Happy Miles by Idea Bank https://t.co/xUKzvJ2JA0 #fintech #banking #loans
RT @StarlingBank: Found yourself wondering what #fintech really is? You're not alone. Max puts it in plain terms here ?Š—_

We have worked on around 30 MB dump of this twitter dataset.Our objective is to analyze the text and find the general sentiment trending on twitter about the company.

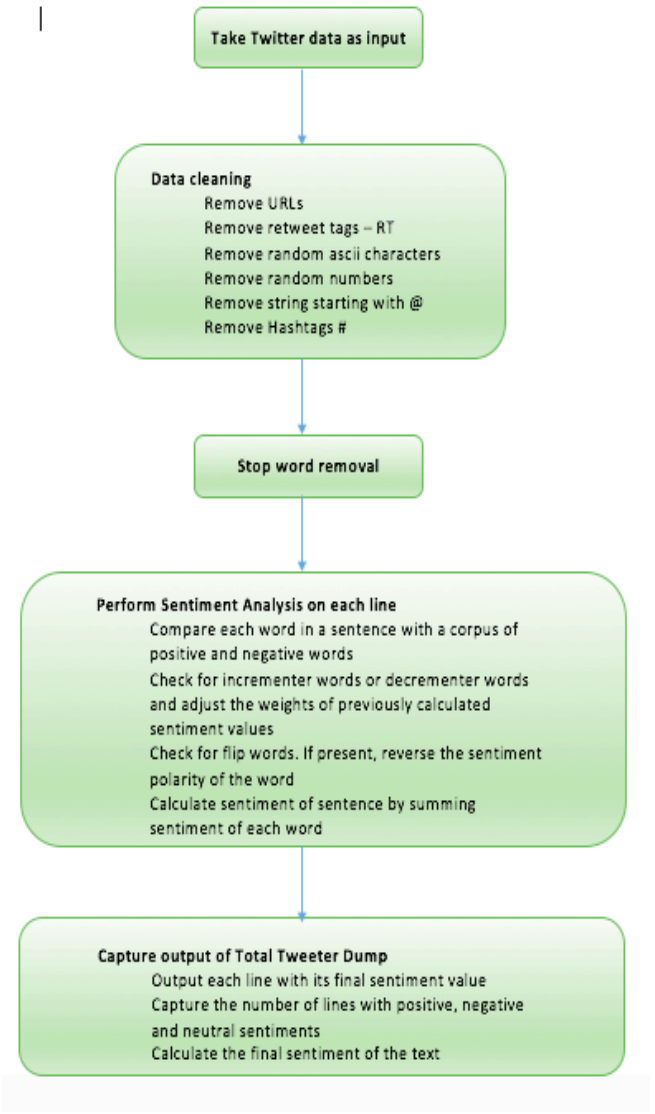
The dataset normally consists of sentences which are short and do not follow proper grammatical syntax of the English language. The majority of the data dump consists of single sentences. There are a few cases where there is paragraph like structures. As is the case with twitter data, "#"and "@" are frequent in our dataset. There are 150000 lines of tweets in total in our dataset.

3. TECHNOLOGIES USED

We have used Python-2.7 for implementing our code to perform Sentiment Analysis and finding the Word Co-occurrence Matrix. We have used Tableau to generate the visualizations used to depict the top occurring word-pairs in the Word Co-occurrence Matrix. NLTK package has been used to extract sentiment from the twitter dump.

4. THE SENTIMENT ANALYZER

Following is a diagrammatic representation of the sentiment analyzer:



We take a sample line from the input Tweeter dump (modified to suit the current explanation) and try to show how it passes through each step of the algorithm and how it changes its form:

Sample input tweet:
RT @neiljpearce: The latest The CIO's Daily! https://t.co/3AlXAji15t Thanks to @CarlosGarriga @PrestonW @jeremychobbs #fintech #iot Š—_

4.1. Data Cleaning

4.1.1. Remove URL

RT @neiljpearce: The latest The CIO's pretty bad Daily! Not Thanks to @CarlosGarriga @PrestonW @jeremychobbs #fintech #iot Š—_

4.1.2. Remove Retweet Tags

@neiljpearce: The latest The CIO's pretty bad Daily! Not Thanks to @CarlosGarriga @PrestonW @jeremychobbs #fintech #iot Š—_

4.1.3. Remove Random ASCII characters

@neiljpearce: The latest The CIO's pretty bad Daily! Not Thanks to @CarlosGarriga @PrestonW @jeremychobbs #fintech #iot

4.1.4. Remove @ References

The latest The CIO's pretty bad Daily! Not Thanks to #fintech #iot

4.1.5. Remove

The latest The CIO's pretty bad Daily! Not Thanks to fintech iot

4.1.6. Tokenization

'The', 'latest', 'The', 'CIO's', 'pretty', 'bad', 'Daily!' ,'Not','Thanks', 'to', 'fintech', 'iot'

4.2. Stop Word Removal

'latest', 'CIO's', 'pretty', 'bad', 'Daily!', 'Not', 'Thanks', 'fintech', 'iot'

4.3. Sentiment Analysis

4.3.1 Sentiment evaluation

Here, the code checks each token against a list of words in the Positive file and negative file. If the word is present in the positive list it is assigned a value of '+1'. If the word is present in the negative list, it is assigned a value of '-1'.

So,

Bad – (-1)
Thanks – (+1)

4.3.2 Intensity evaluation

Now, we check if the word preceding the current word is present in the Increment or Decrement file.
Here we see that the word 'pretty' is present in the Increment file. Hence, we multiply the value of bad by 1.5.

Bad – (-1) * 1.5 = -1.5

The value of thanks remains the same.

Thanks – (+1)

4.3.3 Presence of flip words

Again, we check, if the preceding word of the current word is in the Flip list.
Here, 'not' is present in the Flip list. So we simply flip the sentiment value by multiplying -1 with it

Thanks – (+1) * -1 = -1

4.3.4 Summation of all values

In the final step, we sum all the calculated values.

Bad – (-1.5)
Thanks – (-1)

Total = (-1.5) + (-1) = -2.5

Hence, this sentence is classifies as having negative sentiment having a value of -2.5.

Similarly, we sum the sentiment values of all the tweets in the dump to find the overall sentiment of the Tweets.

4.4. Final Statistics Obtained

After we run the code on the dataset, we obtain the following statistics related to the sentiment of the tweets:

Total Tweets	150000
Tweets with Positive Sentiment	43711
Tweets with Negative Sentiment	13324
Tweets with neutral Sentiment	92965
Sum of sentiment scores of total dataset	38181.5

As per our analysis of the dataset, around 29% of the tweets have positive sentiment, 9% have negative sentiments and 61% have neutral sentiments.

5. SENTIMENT ANALYSIS USING NLTK PACKAGE

Sentiment Analysis using Naive Bayes Classifier:

We implemented sentiment analysis without using any packages. To verify the accuracy of our code, we applied Naive Bayes Classifier algorithm present in NLTK package.

Naive Bayes Classifier: -

Naive Bayes Classification Algorithm will classify each sentence of corpus as positive or negative based on the probability calculation from Bayes Theorem.

Bayes Theorem: - $P(A|B) = \frac{P(B|A)*P(A)}{P(B)}$

Bayes theorem is supervised Machine Learning Algorithm as we are classifying something from previously known classes.
We calculate three things- Posterior Probability, probability of individual class and probability of one observation to belong to each of the two classes in a specified region.
We are running this classification to see each sentence of the corpus are positive or negative.

At the end, it will compare probability of a sentence (x),
If P(Positive|x) > P(Negative|x), Then the sentence (x) will get assigned to positive.

To train the model we used **Opinion Lexicon by Bing Lu**. Using that our model got 86% accuracy. After that we have defined a class named 'SentimentClassifier'. In this class, we have defined two methods- one will classify the data and then we are calculating the confidence by dividing the derived sentiment value with the length of each sentence.
Output after applying Naive Bayes: -

('Original Naive Bayes Algorithm accuracy percent:', 86.345763)

The following table explains how well our model has been trained using positive and negative lexicons.

Most Informative Features	Positive: Negative
envious = True	2.4: 1.0
enviousness = True	2.4: 1.0
enviously = True	1.0: 1.0
keenness = True	9.8:1.0
posh = False	1.0:1.0
loving = True	8.4:1.0
tenaciously = False	3.4:1.0
feature-rich = False	2.6:1.0
cheerful = True	3.2:1.0
adaptive = False	1.0:1.0
mesmerize = True	4.3:1.0
stimulating = True	13.6:1.0
liberation = True	11.9:1.0
perfect = True	9.6:1.0
convincing = False	4.6:1.0

From the above table, let us pick any row and try to understand what it means:

stimulating = True	**13.6:1.0**

This record indicates that the word 'stimulating' has been convincingly classified as having positive sentiment.

After training the model with a sample dataset, the model know which words are tagged as negative and which as tagged as having positive sentiment. Now, we apply this trained model on our twitter dataset and find out the relevant statistics. The following snapshot is taken from the output after applying NLTK package:

Index	Type	Size	Value
73	tuple	2	('pos', 1)
74	tuple	2	('neg', 1)
75	tuple	2	('neg', 1)
76	tuple	2	('neg', 1)
77	tuple	2	('neg', 1)
78	tuple	2	('neg', 1)
79	tuple	2	('neg', 1)
80	tuple	2	('pos', 1)

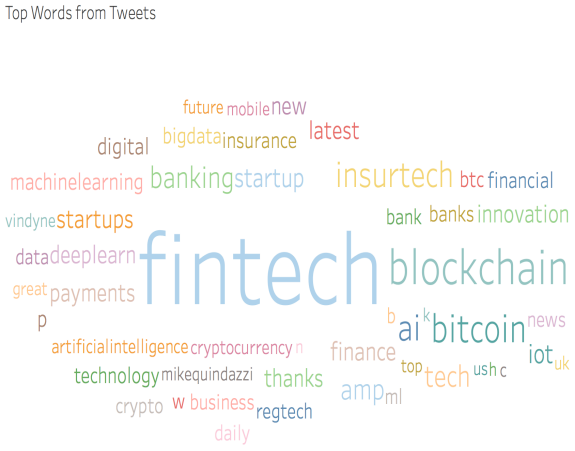
Details of sentiment analysis that we get after applying NLTK package on our data set:

Total Tweets	150000
Tweets with Positive Sentiment	65214
Tweets with Negative Sentiment	28964
Tweets with Neutral Sentiment	55822
Sum of sentiment scores of total dataset	36250

As we can see from the above table, around 43% of the tweets have positive sentiment, 19% have negative sentiments and 37% have neutral sentiments.

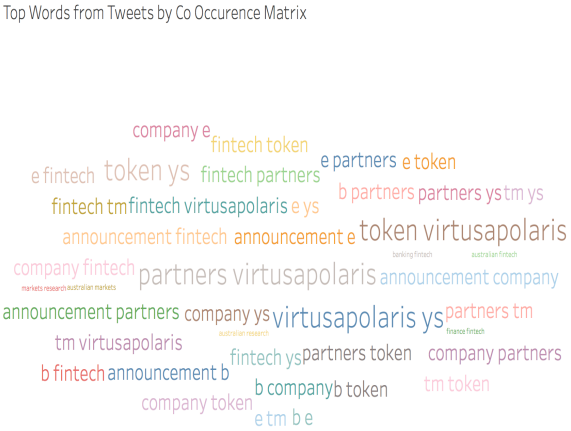
To get more insight about the Fintech dataset, we experimented with the frequency distribution.

Frequency Distribution: - To get the most frequently used single word we have used FreqDist() method from NLTK python package and then most_common() method to get the top frequent words with the count.



After visualizing the data in word cloud we can see the word Fintech has most frequency. We are not getting the points that people are talking about so we implemented Co-Occurrence Matrix.

Co Occurrence Matrix: - Initially, we have used Bigrams but we have not got satisfactory result so we extended our research and designed co-occurrence matrix. We designed a matrix like com[x][y] which contains the number of times the term x has been seen in the same tweet as the term y. We have discarded the count of com[y][x] In this method, as the count will be redundant then.



After visualizing it in word cloud we can see many word pairs with similar importance are coming together.

Word	Frequency
fintech	363342
blockchain	79039
bitcoin	79039
ai	41880
insurtech	38166
banking	28705
tech	26716
amp	24928
iot	21337
startup	20004

Word1	Word2	Frequency
b	fintech	30297549
e	fintech	30292982
company	fintech	30290931
Company	fintech	30287452
announcement	fintech	30287229
fintech	token	30287083
b	e	30285863
finech	tm	30285546
b	Company	30285516
fintech	ys	30285500

We did some research on words we are getting from both frequency distribution and co-occurrence, hence we can conclude: -

- People discussed about Virtusa Polaris, Fintech Lab. As we have already seen most people are positive about Fintech, hence we can conclude that people are discussing about the development of Fintech.
- People discussed about Australian Fintech world and its clash, collaboration with Banks.
- We can see the word announcement has a high frequency, hence tweet users are mostly discussed about new partners, company updates and latest news of Fintech.
- Technologies related to Fintech. - Bitcoin which use the Block chain method or crypto currency is an application of Artificial Intelligence, Machine Learning, Big Data etc.
- People expressed positive opinion for latest technologies related to Fintech - Insurtech, wealthtechbook, Iot, Regtech.

6. EXECUTION DETAILS

6.1. Details of custom code

The following files have been provided in a separate folder which will help in running the Sentiment Analyzer:

1. Code: SentimentAnalyzer.py
2. Input: Fintech_Twitter_Dump.txt
3.Other files: inc.txt, dec.txt, flip.txt, positive.txt, negative.txt
4. Output file: SentimentAnalysisOutput.txt

6.2. Details of code using NLTK package

1. Code: Sentiment_Analysis_Naive_Bayes.py
2.Input:https://drive.google.com/file/d/0B5ruspDK7KIccE5jVE1HdHBxSWM/view?ts=59091663

7. EVALUATION

We have already provided the output of sentiment analysis both from our custom code and using NLTK package. The following table provides a comprehensive comparison between both:

	Custom Code	**NLTK**	**% difference**
Positive Sentiment	43711	65214	32.9%
Negative Sentiment	13324	28964	53.9%
Neutral Sentiment	92965	55822	66.8%
Sum of sentiment scores	38181.5	36250	5.3%

As the above table suggests, the output of our custom code is quite different from the output of Naïve Bayes's algorithm of the NLTK package.

8. FUTURE WORK

During our implementation of the python code, one thing that became quite clear to us was the speed of execution of our code. The predefined NLTK packages perform quite faster as compared to our implementation. This presents us with a scope of future work, where we can try to optimize our code and make it run faster.
During our proposal of project outline, we had mentioned that we plan to include subtler sentiments like sarcasm, mood etc. in our analysis. But due to shortage of time, we are not able to implement the same. In our future project, we plan to include these sentiments as part of our analysis.
The Increment, Decrement and Flip files present in the current implementation are not extensive. We will try to add more relevant words to these lists in the future.
Fine tune the Positive and Negative word files.
As the deviation from NLTK package is quite high, a lot of fine tuning is also required in the code.

9. CONCLUSION

As we have presented in our analysis, the sentiment of the Tweeter users about the company Fintech is overall positive. Around 9% of the tweets convey a negative sentiment, 61% convey neutral sentiments and remaining 29% are talking positively about Fintech. When we look at the output of the analysis we get after applying NLTK packages on the Tweeter data set, we observe that around 19% of the tweets convey a negative sentiment, 37% convey neutral sentiments and remaining 43% are talking positively about Fintech.
Looking at the output of the Word Co-occurrence Matrix, we see that the wordpair 'b,fintech' occur very frequently followed by 'e,fintech''

We can safely conclude that there is a positive sentiment about the Fintech on Tweeter.

10. REFERENCES

1.http://fjavieralba.com/basic-sentiment-analysis-with-python.html

2.Apoorv Agarwal, Fadi Biadsy, and Kathleen Mckeown. 2009. Contextual phrase-level polarity analysis using lexical affect scoring and syntactic n-grams. *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 24–32, March.

3.Luciano Barbosa and Junlan Feng. 2010. Robust sentiment detection on twitter from biased and noisy data. *Proceedings of the 23rd International Conference on Computational Linguistics*: *Posters*, pages 36–44.

4.Adam Bermingham and Alan Smeaton. 2010. Classifying sentiment in microblogs: is brevity an advantage is brevity an advantage? ACM, pages 1833–1836.

C. Fellbaum. 1998. *Wordnet, an electronic lexical database*. MIT Press.

5.Michael Gamon. 2004. Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. *Proceedings of the 20th international conference on Computational Linguistics*.

6.Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. Technical report, Stanford.

7.David Haussler. 1999. Convolution kernels on discrete structures. Technical report, University of California at Santa Cruz.

8.M Hu and B Liu. 2004. Mining and summarizing customer reviews. KDD. S M Kim and E Hovy. 2004. Determining the sentiment of opinions. Coling.