DETECTION OF SEMANTIC OBSESSIVE TEXT IN MULTIMEDIA USING MACHINE AND DEEP LEARNING TECHNIQUES AND ALGORITHMS

[1]AADIL GANI GANIE, [2]DR SAMAD DADVANDIPOUR, [3]MOHD AAQIB LONE

[1]PhD, University of Miskolc, Institute of Information Sciences, Hungary

[2]Associate Professor, University of Miskolc, Institute of Information Sciences, Hungary

[3]PhD, Pondicherry University, Computer Science Engineering, Kashmir, India

E-mail: [1]aadilganianie.com, [2]dr.samad@uni-miskolc.hu, [3]aqiblone768@gmail.com

ABSTRACT

Word boycott has been seen frequently trending in India on various social media platforms. We studied the obsession of Indians with the word boycott; to show the protest or dissent against any government policy, Netflix series, political or religious commentary, and on various other matters, people in India prefer to trend word "Boycott" on multiple mediums. We studied how ingrained the word "Boycott" is in Indians in our research and how it affects daily life, unemployment, and the economy. The data was collected using Youtube API with the next page token to get all the search results. We preprocessed the raw data using different preprocessing methods, which are discussed in the paper. To check our data's consistency, we fed the data into various machine learning algorithms and calculated multiple parameters like accuracy, recall, f1-score. Random forest showed the best accuracy of 90 percent, followed by SVM and Knn algorithms with 88 percent each. We used word cloud to get the most dominant used words, Textblob, for sentiment analysis, which showed the mean Polarity of 0.07777707038498406 and mean subjectivity 0.2588880457405638. We calculated perplexity and coherence score using the LDA model with results -12.569424703238145 and 0.43619951201483725, respectively. This research has observed that the word boycott is a favorite to the Indians who are often using it to show opposition or support related day-to-day matters.

Keywords: *Sentiment analysis, NLP, Machine learning, Deep learning, Topic modeling*

1. INTRODUCTION

We all know about boycotting occasions or places in today's world; anybody in some way, shape, or type boycotts anything. Social media assists in furthering a specific cause trending option. It is whether in political or apolitical environments. People show their agitation by using hashtags; there are boycotts, TV show boycotts, movies, and political rallies boycotts at sporting events. Indians seem to be more obsessive about the boycott; almost every day, the word boycott trends toward India's Twitter. In times of the Irish "Land War" of the late 1800s, a British Captain named Charles Boycott was an absentee landlord agent named Lord Erne in County Mayo, Ireland. By 1881, the word "boycott" was used to describe products[1] figuratively. Some are linking this obsession with unemployment as well.

The Business Standard daily paper detailed that the government's official study showed that India's unemployment rate had taken off to a 45 year high amid 2017-2018, bringing Prime Minister of India affliction months before what was anticipated to be a closely fought general election. The evaluation conducted by the National Sample Survey between July 2017 and June 2018 detailed that the unemployment percentage expanded to 6.1 percent, the most extreme since 1972-73, the daily paper detailed. The study showed that unemployment in urban zones stood at 7.8 percent compared to 5.3 percent in provincial zones [2]. The last study detailed by the Statistics Ministry found that in 2015/16, the unemployment percentage expanded to 5.0 percent, from 4.9 percent within the past year and 4.0 percent in 2012/13. Government measurements showed that male unemployment remained at 4.3% and 8.7% among females in 2015/16. An independent driving think-tank, the Centre for Checking the Indian economy, said that at the starting of January 2019, the nation lost almost 11 million employments last year. The Amul milk

Table : 1

Study	Method	Measure	Score Achieved
Fuller et al. [8]	ML(NN)	Adhoc	73
Almela et al. [9]	ML(SVM)	Adhoc	73
Fornaciari [10]	ML(SVM)	Adhoc	69
Banerjee et al. [11]	ML(SR)	Otto et al.	71
Chen [12]	ML(SVM)	Adhoc	61
Hernandez et al. [13]	ML(PRU)	Otto et al.	78
Kim et al. [14]	ML(SVM)	Extend. Otto et al.	92
Mihalcea et al. [15]	ML(NB)	Adhoc	70
Rubin et al. [16]	ML(LR)	Adhoc	56

brand is not only well known for its dairy items. The cartoon has a colossal following on social media, especially on Twitter. It caused a stun among its supporters when its Twitter account was stopped briefly after animation in China. In this way, on Twitter itself, hashtags #BoycottTwitter[3] were trending. In India, consumer boycotts go back to the movement for independence. Between 1905 and 1912, the Swadeshi movement was raised in India. Between the colonial-era boycotts and those who now seek "Swadeshi's" goods, a lot has changed — from a need for socio-economic change to a sense of recompense. "Pakistan and China are widely perceived as nation states hostile to India by consumers, and recent attempts at boycott are being made against or belonging to these countries.

Indians are boycotting the Chiese products and trending the same on Twitter. However, the question remains whether it is pragmatic to boycott Chinese goods. Exchange figures illustrate India is the colossal merchant of Chinese consumer items. India imports from China about seven times more than it exports to it. India incorporates a significant exchange shortfall with China, its biggest with any locale. In 2018-19, exports from India to China were insignificant, close to $16.7 billion, whereas the number of imports was nearly $70.3 billion, which leaves a trade deficit of $53.6 billion. Trending issues on Twitter are sometimes perceived as a vague signal of the significance of a given topic, according to Charles Warzel [4]. Despite being a highly arbitrary and mostly "worthless metric," The Wire[5] claimed that many of the accounts posting these tweets are followed by ministers in the Bharatiya Janata Party government. It also pointed out that according to Indian law, the tweets and the trending hashtag may be seen as illegal. To study the obsession with the word boycott in Indians, we collected the data from Youtube comments and annotated it manually. Some experts in the psychological field did the annotation, and some experts from the international relation faculty. Many deep learning methods were applied to data to get more information from it.

2. LITERATURE REVIEW

The critical functionality of sentiment analysis is to get meaning out of the text. Several researchers have focused on developing the text sentiment analysis, and extensive work has been completed. We did the Youtube comments' sentiment analysis, which is more like tweet sentiment analysis. Generally, there are two ways of doing sentiment analysis on data 1. Conventional Machine learning approach, and 2. Deep learning approach.

2.1 Conventional Machine learning approaches

Traditional Machine learning approaches mainly involve developing a classifier or using existing classifiers like Support Vector Machine, Linear regression, RandomForest, Naïve Bayes. In [6], they used SVM and Naïve Bayes for online movie reviews; researchers concluded that the Naïve Bayes approach outperforms SVM on the product review dataset. Researchers in [7] collected tweets from March 25 to March 28, 2020 using hashtags #IndiaLockdown and #IndiafightsCorona. For the study, a total of 24,000 tweets was considered. The study was finished using the R product, and a word cloud was developed that represents the tweets' opinions. The optimistic sentiment stood out; researchers concluded. Some researchers worked on choosing the feature vector for context building and, eventually, sentiment analysis. Below are some of the studies with a feature vector, approach used, lexicon, and dataset.

Table 2

Work	Time Period	Feature Set	Lexicon	Classifier	Dataset
Read et al. [17]	2005	N-gram	Emoticons	Naïve Bayes and SVM	Read [17]
Go et al. [18]	2009	N-gram and POS	_	Naïve Bayes, SVM and Maximum entropy	Go et al. [18]
Davido et al. [19]	2010	Punctuation, N-gram, Pattern, Tweet based feature	_	Ken	O'Connor et al.[20]
Zhang et al. [21]	2011	N-gram, emoticons	Ding et al. [31]	SVM	Zhang et al. [21]
Agarwal et al. [22]	2011	Lexicon, POS, percentage of capitalized data	Emoticon listed from Wikipedia. Acronym dictionary	SVM	Agarwal et al. [22]
Speriosu et al. [23]	2011	N-gram, emoticon, hashtag, lexicon,	Wilson et al. [32]	Maximum entropy	Go et al. [18]and Speriosu et al. [23]
Saif et al. [24]	2012	N-gram, POS, semantic feature,	_	Naïve Bayes	Go et al. [18]
Hu et al. [25]	2013	N-gram, POS	_	_	Go et al [18]and Speriosu et al [23] and Shamma et al [26]
Saif et al. [27]	2013	N-gram, POS, lexicons, capitalized text	Mohammed et al.	SVM	Go et al. [18]and Speriosu et al. [21] Nakov et al. [28]

2.2 Deep learning methods

Many researchers have a preference to apply deep learning methods for text classification because of advanced data visualization, better accuracy, the inclusion of LSTM's and BiLSTM's. One such deep learning method used by [29] estimates sentiment polarity and emotions from extracted tweets of COVID-related datasets. LSTM was used to calculate the accuracy of the model using 140 datasets. According to Bishwo [30], most of the Nepalese took a positive approach; however, there was some disgust, fear, and sadness. Many famous deep learning approaches have been tabularized below to cut a long story short and better comprehension and visualization.

3. DATASET

The novelty comes from the novel data. In this research paper, we choose a unique way of data collection. The data was collected from Youtube through Youtube API. We extracted many columns like video title, channel Id, published date, channel name. However, we choose three columns for our research: video title, description, and channel name. Data was collected using the search query method by default; Youtube returns only the first 50 results; we use the next page token of Youtube API to extract all the search results.

Process of collecting data into CSV file:

- Open google spreadsheet and rename it;
- Type the search query in the excel cell;
- Go to the tools and launch script editor
- Go to resources and select advanced google services;
- Enable Youtube data API;
- Write the code for scrapping the data in the script editor;
- Enable the next page token to get all the search results.

3.1 Process of cleaning the data

We got the raw data from the data collection process, and it needs to be preprocessed. We use Python Jupyter notebook for data preprocessing. Our data contains a lot of rudimentary data that needs to be cleaned. Preprocessing was done in the following phases using the regex library of Python.

- Remove HTML data;
- Remove numbers, emails, special symbols;
- Tokenization and removal of punctuation;
- Removal of stopwords;
- Stemming and lemmatization.

For any NLP task preprocessing is a must to avoid inconsistent results from building vocabulary to feature engineering.

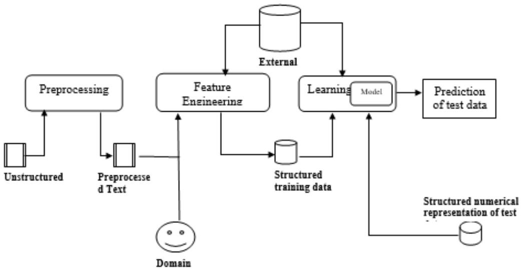


Figure: 1 Natural language process model architecture

We used Onehot encoding for categorical values and TF-IDF for standard text for word representation. The one-hot encoded presentation is a simplified way of representing words. This means that if we have a V-size vocabulary, we describe the word $w_i$with a V-long vector [0, 0, 0,..., 0, 1, 0,..., 0, 0, 0] for each i^{th} word w_i, where the i^{th} element is one, and other elements are zero. Take this sentence as an example.

India China tensions Calls for boycott of Chinese products

The one-hot encoded representation for each word might look like this:

India: [1,0,0,0,0,0,0,0,0]
China: [0,1,0,0,0,0,0,0,0]
tensions: [0,0,1,0,0,0,0,0,0]
calls: [0,0,0,1,0,0,0,0,0]
for: [0,0,0,0,1,0,0,0,0]
boycott: [0,0,0,0,0,1,0,0,0]
of: [0,0,0,0,0,0,1,0,0]
Chinese: [0,0,0,0,0,0,0,1,0]
products: [0,0,0,0,0,0,0,0,1]

TF-IDF is a framework based on the frequency that includes the frequency at which a term in a corpus appears. This is a word representation in the sense that it reflects the value of a particular word in a given text. Intuitively, the higher the word frequency, the more significant in the document the word is. In a text on the boycott, for instance, the term boycott would appear more. However, it would not quantify the frequency since terms like *this* are prevalent but do not hold that much detail. For such standard terms, TF-IDF takes this into consideration and gives a value of zero. Again, TF represents term frequency, and IDF stands for inverse document frequency:

$$TF(w_p) = \frac{\text{number of times } w_i \text{ appear}}{\text{total number of words}}$$

$$\text{IDF } (w_i) = \text{log(total number of documents / number of documents with } w_i \text{ init)}$$

$$\text{TF} - IDF(w) = TF(w)x\text{IDF}(w)$$

Take an example to demonstrate the functioning of TF-IDF:

- Sentence 1: *India China tensions Calls boycott of China products*
- Sentence 2: *How Boycotting Chinese Products Now Will Ruin India*

$$\text{TF-IDF (China, sentence1)} = (2/8) * \log(2/1) = 0.075$$

$$\text{TF-IDF (Now, sentence2)} = (1/8) * \log(2/2) = 0.0$$

Therefore, the word *China* is informative while *Now* is not. In terms of evaluating the value of words, this is the ideal behavior we need.

Stemming is a process used by extracting affixes from words to remove the base structure of the terms. It is almost like chopping a tree's branches down to its stems. For instance, the root of the phrase eating, eats, eaten is eat. There are various algorithms for stemming. We used snowball stemmer in our dataset; the reason for choosing this algorithm is its flexibility in stemming a word. The lemmatization process is similar to stemming. The performance we get after lemmatization is known as 'lemma.' Result of lemmatization is a word that carries some meaning, as depicted below in the figure.

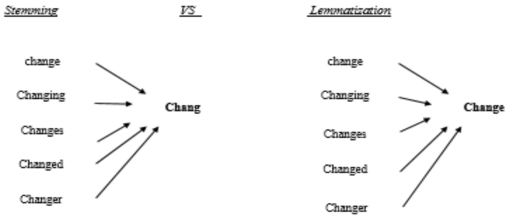
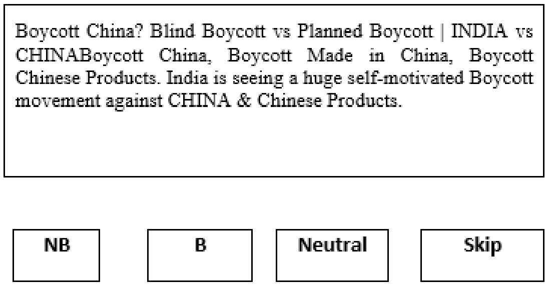


Figure: 2

3.2 Data annotation

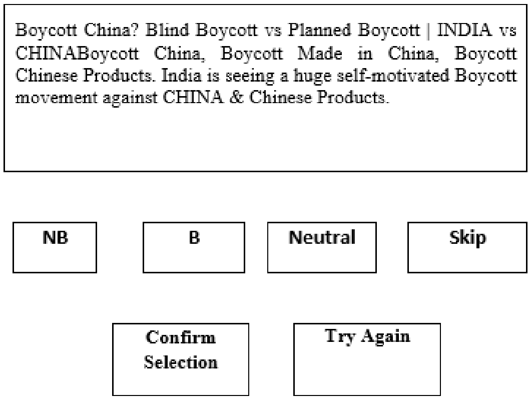
Any model's accuracy depends on annotation precision; annotation can be done manually and through automated means. Manual annotation requires more time and is error-prone, while a manual annotation is done by machine learning algorithms and requires less time. We used the tortus library of Python for automatic data annotation. We choose three labels for classifying a sentence into boycott labels (B), not-boycott (NB), and neutral (Neutral). The Boycott label specifies those sentences that support boycott for anything, non-boycott specifies which do not support the boycott, and neutral label specifies those sentences that are neutral.



Boycott China? Blind Boycott vs Planned Boycott | INDIA vs CHINABoycott China, Boycott Made in China, Boycott Chinese Products. India is seeing a huge self-motivated Boycott movement against CHINA & Chinese Products.

Figure: 3

This sentence falls under boycott (B). We have to click on the specific tab and then confirm its selection



Boycott China? Blind Boycott vs Planned Boycott | INDIA vs CHINABoycott China, Boycott Made in China, Boycott Chinese Products. India is seeing a huge self-motivated Boycott movement against CHINA & Chinese Products.

Figure: 4

After annotation, the dataset looks like the following:

Table 2

	Channel name	Text	Label
0	India Plus	India #Firstreaction on Boycott France campaign	B
1	Al Jazeera English	India-China tensions, calls for boycott of china	B
2	Dhruv Rathee	BoycottChina: The harsh truth….	NB
3	Akash Banerjee	#BoycottChina: Can India's wallet beat China	Neutral

Dataset before annotation

Table 3

	Channel name	Text
0	India Plus	India #Firstreaction on Boycott France campaign
1	Al Jazeera English	India-China tensions, calls for boycott of china
2	Dhruv Rathee	BoycottChina: The harsh truth….
3	Akash Banerjee	#BoycottChina: Can India's wallet beat China

4. Results and Discussion

Recent trends on various social media platforms like Twitter, Facebook, and Youtube show that word boycott is frequently used in India. Be it a skirmish with China or boycotting a particular person for his/her acts, word boycott seems to be in more limelight in India. Users also trend "boycott twitter" on Twitter itself; this shows the obsession of Indians with the word boycott. We tried to study this obsession by collecting Youtube data and applying specific NLP techniques, which are discussed below. To check the parameters like accuracy, precision, recall, f1-score, we applied specific machine learning algorithms like SVM, random forest, and K nearest neighbor. For evaluation, we plot the confusion matrix of every algorithm.

- Support vector machine
  We got 88 percent accuracy on our data.

Table 6: Results of SVM

	Precision	Recall	F1-score
Accuracy			0.88
Macro avg	0.29	0.33	0.31
Weighted avg	0.77	0.88	0.82

- K nearest neighbor
  The same accuracy was recorded for this model, as well as 88 percent.

Table 7: Results of Knn

	Precision	Recall	F1-score
Accuracy			0.88
Macro avg	0.29	0.33	0.31
Weighted avg	0.77	0.88	0.82

- Random forest
  This algorithm showed more accuracy than SVM and Knn, and it achieved 90 percent accuracy on the same data

Table 8: Results of Random forest

	Precision	Recall	F1-score
Accuracy			0.90
Macro avg	0.63	0.40	0.43
Weighted avg	0.89	0.90	0.87

It is evident from the above observations and experiments that there is no inconsistency or anomaly in the data. Wordcloud helps us to identify the most frequently used words in our data. The most used keywords stand better in the word cloud; it is easier to comprehend the word cloud and is visually engaging than table data. We plotted the word clouds for our data after converting it into the document term matrix using TF-IDF.

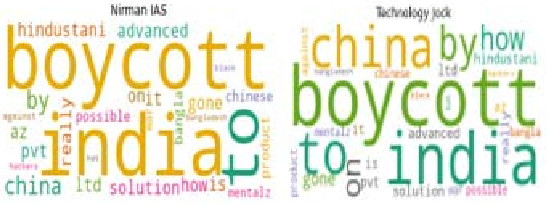


Figure: 5 Wordcloud

The frequency of the word "boycott" is evident from figure 7. This shows the obsession of Indians with the word "boycott." We calculated the Polarity of each YouTube channel using TextBlob. On the grounds of NLTK and Pattern, Textblob is created. A significant benefit of this is that it is quick to learn and provides many features such as sentiment analysis, pos-tagging, extraction of noun phrases. For performing NLP activities, it has now become a go-to library. TextBlob calculates the Polarity and subjectivity of the enumerated text. Polarity's value is in float ranges [-1,1] set where 1 means a positive and -1 means a negative. In general, subjective phrases cites personal opinion, sentiment, or judgment, while objective phrases refer to factual facts. A float in the range of [0,1] is also subjective. The mean Polarity for our data is 0.07777707038498406, and the mean subjectivity comes out 0.2588880457405638. We use topic modeling with the LDA model to verify which topic people are mainly talking about. Topic modeling is an unsupervised machine learning technique capable of automatically searching a set of documents, defining words and phrase patterns within them, and clustering word groups and related phrases that better represent a set of documents. LDA implies each word in each text comes from a subject, and the dispensation of topics per document, the topic is chosen. We have two matrices, therefore:

- $\theta td = \mathrm{P}(\mathrm{t} \mid \mathrm{d})$, the probability distribution (topics in documents)
- $\Phi_{\mathrm{Wt}} = \mathrm{P}(\mathrm{w} \mid \mathrm{t})$, probability distribution (words in topics)

Therefore, the probability of a particular word in a given document, i.e., $P(w \mid d)$ is equal to:

$$\sum_{t\in T} p(w \mid t, d) p(t \mid d)$$

T refers to the total number of subjects, even with all the documents; let us say that there are W words in our vocabulary. If we presume conditional independence, then

$$\mathrm{P}(\mathrm{w} \mid \mathrm{t}, \mathrm{d}) = \mathrm{P}(\mathrm{w} \mid \mathrm{t})$$

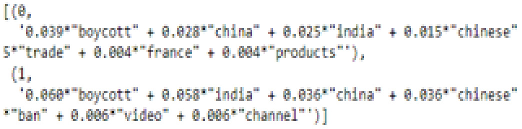
Moreover, hence $P(w \mid d)$ is given as:

$$\sum_{t=1}^{T} p(w \mid t) p(t \mid d)$$

This is the dot product of Θtd and Φwt for each topic t.

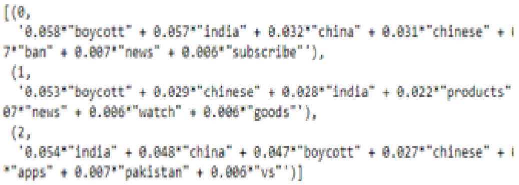
We analyzed with the different number of topics, i.e., 2,3 and 4, and with the number of passes 10 and 25 results are shown below:

Topics =2 and Passes=10



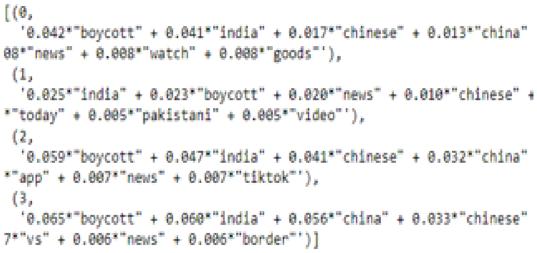
```
[(0,
  '0.039*"boycott" + 0.028*"china" + 0.025*"india" + 0.015*"chinese"
5*"trade" + 0.004*"france" + 0.004*"products"'),
 (1,
  '0.060*"boycott" + 0.058*"india" + 0.036*"china" + 0.036*"chinese"
*"ban" + 0.006*"video" + 0.006*"channel"')]
```

Topics = 3 and Passes =10



```
[(0,
  '0.058*"boycott" + 0.057*"india" + 0.032*"china" + 0.031*"chinese" +
7*"ban" + 0.007*"news" + 0.006*"subscribe"'),
 (1,
  '0.053*"boycott" + 0.029*"chinese" + 0.028*"india" + 0.022*"products"
07*"news" + 0.006*"watch" + 0.006*"goods"'),
 (2,
  '0.054*"india" + 0.048*"china" + 0.047*"boycott" + 0.027*"chinese" +
*"apps" + 0.007*"pakistan" + 0.006*"vs"')]
```

Topics =4 and Passes =15



```
[(0,
  '0.042*"boycott" + 0.041*"india" + 0.017*"chinese" + 0.013*"china"
08*"news" + 0.008*"watch" + 0.008*"goods"'),
 (1,
  '0.025*"india" + 0.023*"boycott" + 0.020*"news" + 0.010*"chinese" +
*"today" + 0.005*"pakistani" + 0.005*"video"'),
 (2,
  '0.059*"boycott" + 0.047*"india" + 0.041*"chinese" + 0.032*"china"
*"app" + 0.007*"news" + 0.007*"tiktok"'),
 (3,
  '0.065*"boycott" + 0.060*"india" + 0.056*"china" + 0.033*"chinese"
7*"vs" + 0.006*"news" + 0.006*"border"')]
```

We can interpret clearly from the above figures that the topic "boycott" is the dominant among all the topics, instead of choosing different values for "number of topics" and "number of passes" parameters. This clears the air for declaring the word "boycott" as the most obsessed word for Indians. At last, we calculated the perplexity and coherence score of our LDA model. Perplexity = -12.569424703238145 and Coherence score = 0.43619951201483725. Perplexity and coherence score are used to evaluate the model, lower the perplexity best the model is, higher the topic coherence, and the topic is more human interpretable.

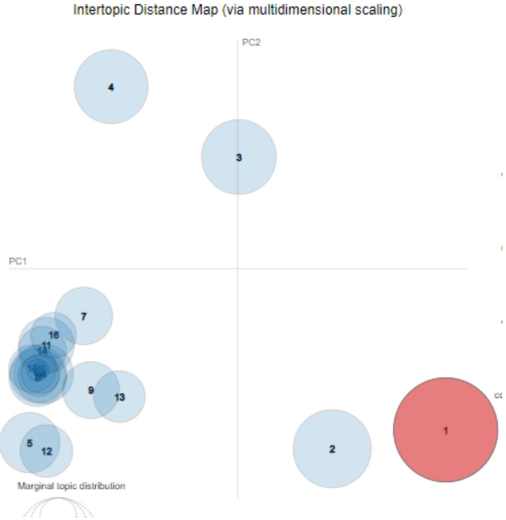


Figure: 6 Distance Map of top 30 topics

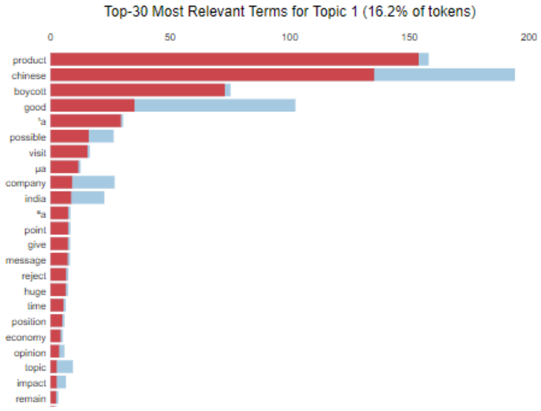


Figure: 7 Most discussed topics

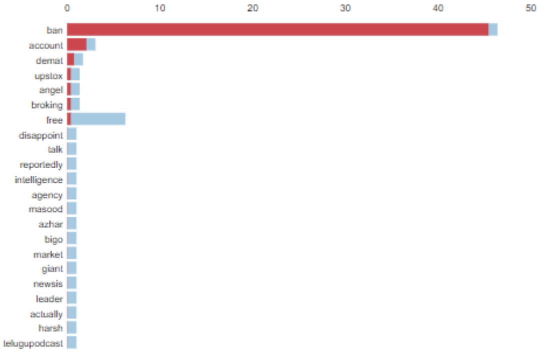


Figure: 8 Frequency of word "ban"

Topic modeling is a type of statistical data analysis to detect the abstract "topics" in a document collection. Latent Dirichlet Allocation (LDA) is a subject model example and is employed to categorize text to a specific subject in a document. It creates a topic per document model and words per theme model, modeled on distributions of Dirichlet. LDA is a technique of matrix factorization. Any corpus (paper collection) can be shown as a document-term matrix in the vector space. A corporation of N documents D1, D2, D3 are shown in the following matrix. Dn and M word size vocabulary W1,W2 ..Wn. The value i,j cells in Document Di gives the frequency count Wj.

Table 9

	W1	W2	W3	Wn
D1	0	2	1	3
D2	1	4	0	0
D3	0	2	3	1
Dn	1	1	3	0

This document-term matrix is transformed by LDA into two smaller dimensions – M1 and M2. M1 consists of a matrix of document-topics and M2 is the topic – the terms matrix of dimensions (N, K) and (K, M) where N is the number of documents, K the numbers of subjects and M the dimensions of the vocabulary.

Table 10

	K1	K2	K3	K
D1	0	2	1	3
D2	1	4	0	0
D3	0	2	3	1
Dn	1	1	3	0

Table 11

	W1	W2	W3	Wm
K1	0	2	1	3
K2	1	4	0	0
K3	0	2	3	1
K	1	1	3	0

LDA is used to change the current topic – word assignment with another assignment – by each word "w" for each document "d." The word "w" has a new theme "k" and P is the product of 2 chances: p1 and p2. Two p1 and p2 probabilities are calculated for each subject. P1 – t-t-document (d) = the percentage of words currently assigned to T-t-Theme in document d. P2 – p(word w / theme t) = the proportion of topic t assignments over all material from this word w.

5. Comparison with previous models

To illustrate the effectiveness of our research, we compared our work with the existing models.

Table 12: Results of [36]

Algorithm	Accuracy	Feature extraction	Dataset	Number of sentiments
SVM	45.71	Document level	IMDB	2
Naïve Bayes	65.75	Document level	IMDB	2

Table: 13 results of [37]

Algorithm	Accuracy	Feature extraction	Dataset	Number of sentiments
SVM	85.4	Uni-gram	Tweeter	2
Naïve Bayes	88.2	Uni-gram	Tweeter	2
Maximum Entropy	83.9	Uni-gram	Tweeter	2
Sematic Analysis	89.9	Uni-gram	Tweeter	2

Table: 14 results of [38]

Algorithm	Accuracy	Feature extraction	Dataset	Number of sentiments
SVM	79.54	Uni-gram, bi-gram, object-oriented	Tweet	2
Naïve Bayes	79.58	Document-level	Tweet	2

From the above observations, we concluded that our model performed competitively with the existing models. With model tuning and more data, we can achieve better results. Since our model considers both regular and sarcastic text, the accuracy it achieved is competitive. SVM performed well than the above methods with 88 percent accuracy, followed by K nearest neighbor with the same accuracy of 88 percent; however, random forest showed the best results with 90 per cent accuracy. The dataset we used is small, which will not show good results in deep learning methods like LSTM, BiLSTM, CNN, etc. It is advised to use traditional machine learning approaches like SVM, Naïve Bayes for shallow networks. Deep learning methods start overfitting when the number of data points is reasonably small. Some state-of-the-art transformer networks like BERT, GPT-2 are computationally very costly and is generally used for machine translation and next sentence prediction using masking language model; however, our task in this paper is entirely different to get the correlation between the word boycott with Indians for which the traditional machine approaches suffice.

6. Conclusion

This research was conducted to study the obsession of Indians with the word boycott as the word has been seen trending almost every day in India on various social media platforms. The data has been collected using Youtube API, and the same has been preprocessed using data cleaning methods. The information has been fed into various machine learning algorithms for anomaly and inconsistency detection. The random forest provided the best accuracy of 90 percent. The correlation between word obsession and Indians has been studied using the word cloud, topic modeling, sentiment analysis with TextBlob. The perplexity and coherence score for LDA comes out -12.569424703238145 and 0.43619951201483725, respectively, while as mean Polarity and subjectivity of TextBlob comes out to be 0.0777707038498406 and 0.2588880457405638, respectively. We concluded that the word boycott is ingrained in Indians, and they are using it for both support and oppose any activity, whether it is political or apolitical. Due to the nature of the dataset and its size, we didn't consider deep learning or transformer network approaches, which will edge the findings. An extensive dataset is needed to generalize the premise of India's obsession with the boycott and make a conclusion based on state-of-the-art approaches like BERT, GPT-2, etc.

REFRENCES:

[1] "The Fascinating Origins of the Word' Boycott' | The Fact Site." https://www.thefactsite.com/boycott-word-origins/ (accessed January 12, 2021).

[2] "unemployment in India: India's unemployment rate hit a 45-year high in 2017-18: report, Auto News, ET Auto." https://auto.economictimes.indiatimes.com/news/industry/indias-unemployment-rate-hit-45-year-high-in-2017-18-report/67772184 (accessed January 13, 2021).

[3] "Twitterati had a field day as #BoycottTwitter trended on the platform | Trending News. The Indian Express." https://indianexpress.com/article/trending/trending-in-india/amul-china-controversy-boycott-twitter-memes-6446964/ (accessed January 13, 2021).

[4] "What do we do about bigoted anti-Muslim hashtags trending on Twitter?" https://scroll.in/article/941244/what-do-we-do-about-bigoted-anti-muslim-hashtags-trending-on-twitter (accessed January 13, 2021).

[5] "Ministers Follow Hate Accounts That Made Call to Boycott Muslims a Top Twitter Trend." https://thewire.in/communalism/ministers-hate-accounts-twitter-follow-boycott-muslims (accessed January 13, 2021).

[6] S. V Wawre and S. N. Deshmukh, "Sentiment classification using machine learning techniques," *Int. J. Sci. Res.*, vol. 5, no. 4, pp. 819–821, 2016.

[7] G. Barkur and G. B. K. Vibha, "Sentiment analysis of nationwide lockdown due to COVID 19 outbreak: Evidence from India," *Asian J. Psychiatr.*, 2020.

[8] C. M. Fuller, D. P. Biros, and R. L. Wilson, "Decision support for determining veracity via linguistic-based cues," *Decis. Support Syst.*, vol. 46, no. 3, pp. 695–703, 2009.

[9] Á. Almela, R. Valencia-García, and P. Cantos, "Seeing through deception: A computational approach to deceit detection in Spanish written communication," *Linguist. Evid. Secur. Law Intell.*, vol. 1, no. 1, pp. 3–12, 2013.

[10] T. Fornaciari and M. Poesio, "DeCour: a corpus of DEceptive statements in Italian COURts.," in *LREC*, 2012, pp. 1585–1590.

[11] S. Banerjee and A. Y. K. Chua, "Applauses in hotel reviews: Genuine or deceptive?," in *2014 Science and Information Conference*, 2014, pp. 938–942.

[12] R.-B. Tang *et al.*, "Serum uric acid and risk of left atrial thrombus in patients with nonvalvular atrial fibrillation," *Can. J. Cardiol.*, vol. 30, no. 11, pp. 1415–1421, 2014.

[13] D. H. Fusilier, M. Montes-y-Gómez, P. Rosso, and R. G. Cabrera, "Detecting positive and negative deceptive opinions using PU-learning," *Inf. Process. Manag.*, vol. 51, no. 4, pp. 433–443, 2015.

[14] S. Kim, H. Chang, S. Lee, M. Yu, and J. Kang, "Deep semantic frame-based deceptive opinion spam analysis," *Int. Conf. Inf. Knowl. Manag. Proc.*, vol. 19-23-Oct-2015, pp. 1131–

1140, 2015, DOI: 10.1145/2806416.2806551.

[15] R. Mihalcea and C. Strapparava, "The lie detector: Explorations in the automatic recognition of deceptive language," in *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, 2009, pp. 309–312.

[16] V. L. Rubin and T. Lukoianova, "Truth and deception at the rhetorical structure level," *J. Assoc. Inf. Sci. Technol.*, vol. 66, no. 5, pp. 905–917, 2015.

[17] J. Read, "Using emoticons to reduce dependency in machine learning techniques for sentiment classification," in *Proceedings of the ACL student research workshop*, 2005, pp. 43–48.

[18] A. Go, R. Bhayani, and L. Huang, "Twitter sentiment classification using distant supervision," *CS224N Proj. report, Stanford*, vol. 1, no. 12, p. 2009, 2009.

[19] D. Davidov, O. Tsur, and A. Rappoport, "Enhanced sentiment learning using Twitter hashtags and smileys," in *Coling 2010: Posters*, 2010, pp. 241–249.

[20] B. O'Connor, R. Balasubramanyan, B. Routledge, and N. Smith, "From tweets to polls: Linking text sentiment to public opinion time series," in *Proceedings of the International AAAI Conference on Web and Social Media*, 2010, vol. 4, no. 1.

[21] L. Zhang, R. Ghosh, M. Dekhil, M. Hsu, and B. Liu, "Combining lexicon-based and learning-based methods for Twitter sentiment analysis," *HP Lab. Tech. Rep. HPL-2011*, vol. 89, 2011.

[22] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. J. Passonneau, "Sentiment analysis of Twitter data," in *Proceedings of the workshop on language in social media (LSM 2011)*, 2011, pp. 30–38.

[23] M. Speriosu, N. Sudan, S. Upadhyay, and J. Baldridge, "Twitter polarity classification with label propagation over lexical links and the follower graph," in *Proceedings of the First Workshop on Unsupervised Learning in NLP*, 2011, pp. 53–63.

[24] H. Saif, Y. He, and H. Alani, "Semantic sentiment analysis of Twitter," in the *International semantic web conference*, 2012, pp. 508–524.

[25] X. Hu, L. Tang, J. Tang, and H. Liu, "Exploiting social relations for sentiment analysis in microblogging," in *Proceedings of the sixth ACM international conference on Web search and data mining*, 2013, pp. 537–546.

[26] D. A. Shamma, L. Kennedy, and E. F. Churchill, "Tweet the debates: understanding community annotation of uncollected sources," in *Proceedings of the first SIGMM workshop on Social media*, 2009, pp. 3–10.

[27] S. M. Mohammad, S. Kiritchenko, and X. Zhu, "NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets," *arXiv Prepr. arXiv1308.6242*, 2013.

[28] P. Nakov, A. Ritter, S. Rosenthal, F. Sebastiani, and V. Stoyanov, "SemEval-2016 task 4: Sentiment analysis in Twitter," *arXiv Prepr. arXiv1912.01973*, 2019.

[29] A. S. Imran, S. M. Daudpota, Z. Kastrati, and R. Batra, "Cross-cultural polarity and emotion detection using sentiment analysis and deep learning on COVID-19 related tweets," *IEEE Access*, vol. 8, pp. 181074–181090, 2020.

[30] B. P. Pokharel, "Twitter sentiment analysis during the covid-19 outbreak in Nepal," *Available SSRN 3624719*, 2020.

[31] J. Islam and Y. Zhang, "Visual sentiment analysis for social images using transfer learning approach," in *2016 IEEE International Conferences on Big Data and Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom)(BDCloud-SocialCom-SustainCom)*, 2016, pp. 124–130.

[32] A. Severyn and A. Moschitti, "Twitter sentiment analysis with deep convolutional neural networks," in *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2015, pp. 959–962.

[33] L. Yanmei and C. Yuda, "Research on Chinese micro-blog sentiment analysis based on deep learning," in *2015 8th International Symposium on Computational Intelligence and Design (ISCID)*, 2015, vol. 1, pp. 358–361.

[34] C. Li, B. Xu, G. Wu, S. He, G. Tian, and H. Hao, "Recursive deep learning for sentiment analysis over social data," in *2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, 2014, vol. 2, pp. 180–185.

[35] R. Socher *et al.*, "Recursive deep models for semantic compositionality over a sentiment treebank," in *Proceedings of the 2013 conference on empirical methods in natural language processing*, 2013, pp. 1631–1642.

[36] Gautam, G., & Yadav, D. (2014, August). Sentiment analysis of twitter data using

machine learning approaches and semantic analysis. In *2014 Seventh International Conference on Contemporary Computing (IC3)* (pp. 437-442). IEEE.

[37] Le, B., & Nguyen, H. (2015). Twitter sentiment analysis using machine learning techniques. In *Advanced Computational Methods for Knowledge Engineering* (pp. 279-289). Springer, Cham.

[38] Neethu, M. S., & Rajasree, R. (2013, July). Sentiment analysis in twitter using machine learning techniques. In *2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT)* (pp. 1-5). IEEE.