iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** IV **Month of publication:** April 2022

DOI: https://doi.org/10.22214/ijraset.2022.41785

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue IV Apr 2022- Available at www.ijraset.com

Comparison of Sentiment Analysis Algorithms Using Twitter and Review Dataset

Bhanu V Gupta[1], Varun Singh[2], C. Lakshmi[3]
[1, 2, 3]*Department of Computational Intelligence,* SRM Institute of Science and Technology (SRMIST), Chennai, India

Abstract: Sentiment Analysis or opinion mining is the computational examination of sentiments, perspectives and emotions conveyed in written language. In recent years, sentiment analysis has become an active study in fields of natural language processing and text mining. The internet is full of textual data in the form of papers, articles and blogs. Sentiment analysis is the best way to extract key information from this data. In this paper we compare classification of tweets using rule based methods, Machine Learning method and Deep Learning method.
Keywords: Sentiment Analysis, Opinion Mining, Twitter Dataset, Classification, Natural Language Processing.

I. INTRODUCTION

People may communicate their opinions, ideas, sentiments, and judgments on a variety of issues ranging from education to entertainment, thanks to the introduction of social media platforms such as Facebook, Twitter, LinkedIn, and Instagram. These platforms house a massive quantity of data. 90% of this data is in textual or media form. Sentiment Analysis is a technique for deducing the polarity of emotions such as joy, sadness, grief, hatred, rage, and affection, as well as opinions, from text, reviews, and postings accessible on these platforms.

Opinion mining for sentiment analysis identifies a text's sentiment in relation to a certain source of information. Due to various slang phrases, wrong spellings, short forms, varied characters, regional language, repeated characters, and incoming emojis, sentiment analysis is very complex and ever growing research field. Social media is one of the places where we can see effective use of sentiment analysis.

Efficacy is what makes sentiment analysis very popular. There are billions of text documents in the world and sentiment analysis can be carried out of all of them. It is an accurate process with high success results. A few applications where sentiment analysis proves to be useful are:•

Purchasing Merchandise or Service: When buying a certain commodity and/or utility we want to get the perfect option available for us. We can go through the reviews one by one and find out for ourselves if the product or service is good enough or we can let a sentiment analysis algorithm take over this task for us. This way now we can evaluate reviews or other text and opinions of any service and/or product.

1) Recommendation System: The algorithm that can forecast what all items should be suggested according to user preferences by assessing and classifying people's opinions based on their preferences and interests.
2) Quality Reviews and Improvements: Opinion mining allows manufacturers to gather consumer feedback on their product or service, whether positive or negative, and then improve the features and quality as per user needs..
3) Making a Decision: People's sentiments, thoughts, and feelings are all crucial factors to consider while making a decision. When purchasing any item, whether it is a book, clothing, or electronic devices, the user first reads the thoughts and reviews of that product, and these significantly influence customer's thinking.
4) Marketing research: Using sentiment analysis in for market research purpose may be used to assess customer attitudes toward a product or service, as well as any new government policy.

The stages of sentiment analysis are as follows:

a) *Pre-Processing:* The raw data is initially cleaned up.
b) *Feature Extraction:* The keywords are assigned a token, which is then subjected to analysis.
c) *Classification Phase:* These keywords are assigned to one of several categories based on various methods.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue IV Apr 2022- Available at www.ijraset.com

II. LITERATURE REVIEW

Saad and Yang [1] in year 2019 gave way for new developments in sentiment analysis using machine learning algorithms. They both put forward a model which encapsulated preprocessing and feature extraction of tweets. Support Vector Regression and Multinomial Supply Regression were used for classification of sentiments. The results show that the proposed model outperformed others and earned the simplest accuracy over other methods. Fang [2] in year 2018 urged a multi-strategy model based on victimization of the linguistics blurriness for partitioning the issues. This method increased the model efficiency multiple folds.

Afzaal [3] in year 2019 proposed a new aspect based approach to sentiment analysis. This method was very precise and the most effective way to classify the corpus with maximum accuracy.This model was made for mobile applications as a result it was also tested on other systems and proved to be effective in all types of conditions.

Ray and Chakrabarti [4] in year 2019, devised a deep learning method to extract features from textual Daya and perform sentiment analysis. A 7 layers Convolution Neural Network is implemented for feature extraction and labelling the corpus so as to boost the performance. In this way simplest accuracy and efficiency was achieved.

Joscha[5] compared varied ways and techniques of algorithms to boost the performance of sentiment analysis. A few of these techniques involved algorithms like Bag of words models and n-grams for mistreatment linguistics data. The 1st approaches did not consider the relation between words and took them into account individually without making a correlation with other words. A. Hogenboom [6] planned a Rhetoric Structure Theory (RST) tree structure to do the task of sentiment analysis. They put the binary data in the random forest. The machine learning implemented raised the accuracy and efficiency of the model with F1 score of 71.9%.

Xu [7] in year 2020, introduced NB methodology for multi-domain review classification of sentiment platform for mainly E-commerce. Similarly this method was also introduced to learning fashion trends. Upon further fine tuning and review of algorithm in various real-case aspects it was found that the algorithm works extremely well with the Amazon product reviews and picture show review and give a high accuracy score for same.

Smadi [8] improved the existing model for specifying defects based on feature extraction. Firstly, he conducted this research on Arabic hotel reviews. Aimed to get the exact sentiments he was able to build a decent model with provided with high accuracy using the correct features from the corpus. Secondly, SVM and Deep RNN were developed and trained with lexical, word, morphological, semantic, and grammar language features. The dataset of the Arabic hotel' review dataset was used for evaluating the given proposed model. The outcomes have shown that SVM was playing well in comparison to the RNN model. In 2020, Masood explored the impact of assorted patterns happening within the year of 2012 to 2016 on various stock markets. In this case, dataset from Twitter was utilized for computing the sentiment analysis of every one of those events. The Twitter dataset enclosed many tweets from all backgrounds that were made for outlining the event sentiment.

III. ALGORITHMS FOR SENTIMENT ANALYSIS

A. VADER

VADER (Valence Aware dictionary for Sentiment Reasoning) is a NLP algorithm model used for textual sentiment analysis. The model divides the text data into each polarity (positive/negative) and the strength of each feeling or sentiment. This method is applied on unlabelled textual data and is out there within the NLTK package. VADER method of sentiment analysis depends on a dictionary that maps each word options to sentiment referred to as sentiment scores. The sentiment score of a text is obtained by summarizing the intensity of every word in the text.

Example - Words like “happy’’, ‘enjoy’, ‘love’ all shows a positive sentiment, conjointly the model is smart. The model grasps the underlying meaning of words and sentences, words like “did not approve’’ is perceived as a negative statement. VADER is able to understand the stress of punctuations and capitalisations, such as “FUN ".

B. SENTIWORDNET

SENTIWORDNET algorithm is an upgarade over the VADER model and is made from the automatic annotation of the sets of WORDNET in line with the polarity of “positivity”, “negativity”, and “neutrality”. Every textual data(corpus) is associated with 3 numerical scores

Pos(s), Neg(s), and Obj(s) that indicate how positive, negative, and “objective” (i.e., neutral) are the terms contained within corpus. Completely varied senses of a similar term might have varied sentiment related properties. Words like not do show a negative impact on the whole sentence.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue IV Apr 2022- Available at www.ijraset.com

C. NAIVE BAYES

Thomas Bayes' Theorem forms the basis of Naive Bayes classifier. Naive Bayes classifier works on the assumption that one feature is completely unrelated to all other features and moreover they don't affect each other in any way. Naive Bayes works on the concept of basic probability and the model is straightforward to create. It is useful where very large dataset comes into play. Besides simple implementations, Naive Bayes exceeds even extremely subtle classifications.

Naive Bayes is an easy to implement machine learning model which runs on fundamental concepts of probability. Equations of Naive Bayes are:

1) P C|X = P(X|C) ∗ P(C)/P(X)
2) P(C|X) = C class posterior probability (3) P(C) = C class probability
3) P(X|C) = probability of predictor given the class. (5) P(X) = predictor probability

IV. EXPERIMENTAL SETUP

We begun by analysing a few research papers and open-source softwares for sentiment analyzing. We made a summary and draft pro and cons for each algorithm and proposed methodology. We studied about classification algorithms and how they can be used to best fit our use case in terms to classifying textual data. We studied about frequently used algorithms for classification.

We used Cornell university movie reviews dataset also known as Yelp movie review dataset. The dataset contains 10,000 movie reviews in the form of textual data. We have attached the dataset in the rt-polaritydata directory folder inside this repository. This subset is of 5000 positive reviews and 5000 negative movie reviews. We have built a rule based binary classifier (using corpus VADER and SentiwordNet) and machine learning based classifier(Naive Bayes classifier) to do the comparison between the two.

Next step was to get the dataset into usable format and ready for our use using the right modifications. This was done in following steps.

1) Data Processing: Process the raw data, convert it to a pandas data frame, and manipulate the data according to your need and save it to another csv file.
2) Data Vectorisation: The process of converting the textual dataset to a vector of integers using one hot encoding(single layer). Deep learning models do not accept any text based input rather you need to feed the inputs as integers or floats types.
3) Data Vocabulary: we need to create a vocabulary for an NLP sentiment analysis task, because our algorithm can learn only from the words and corpus it has seen before. So for this reason we need to know the frequency of words in a text corpus, along with where it appears in the text. We can easily store this in a python dictionary.
4) Data processing in PyTorch: We process and compute the data in PyTorch in using torch data loaders. These automatically converts the dataset in batches for us. We need not to split the dataset. It also handles the auto-grad for us.
5) Batching of the data
6) Shuffling of the data
7) Loading of the data in parallel: using multiprocessing workers.

With the preprocessing of our data done we next moved to create our own deep learning model to classify the tweets. For this classification we choose Convolution Neural Network (CNN) algorithm because of its adaptability to change and the latency to be modified easily. Changes in dimensionality and channels which the CNN algorithm was the reason to choose this over any other classification algorithms.

The model here has a goal to find out how many layers for convolution neural network will be suitable considering both accuracy and computational power required to train the model. It is always that the linear (or fc) layers perform better when it comes to classification of data.

We begin creating the model by constructing a data tensor and running it on our algorithm. We used a 3 - dimensional data. If you use a one-hot vector for each character in a sequence of characters, a sequence of one hot vectors is a matrix, and a mini batch of one-hot matrices is a three-dimensional tensor. By utilizing the notion of convolutions, the dimension of every one of one hot vector (the corpus size in general) is same as the number of "input channels" and the length of the character sequence is known as the "width."

Table 1: Statistics of the Dataset used	
Features	Total Numbers
Total sentences	65,845
Total words	1,4,54,723
Average words/sentence	22.09

Our focus was to present a comparison on normal algorithms, machine learning algorithms and our deep learning model. We used the CNN type of deep learning algorithm due to the numerous variation possibilities which it offers to modify and fine-tune our model in any way we want. Theoretically the novice deep learning should have outperformed the generic algorithm as well as ML model.

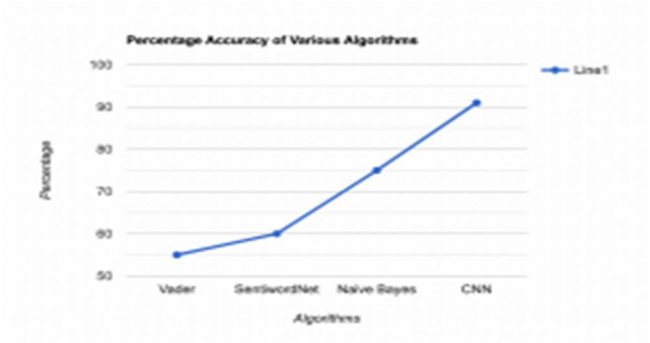
The model was trained using a CUDA. Whole of corpus was divided into train and test dataset and then the computations were executed.

V. RESULTS

Vader, SentiwordNet and Naive Bayes Algorithm, were performed on the Data set and these algorithm's performance was noted against our deep learning model.

Table 2: Percentage Accuracy			
Vader	SentiwordNet	Naive Bayes	CNN
55%	60.82%	75%	91.3%

Fig 1: Percentage Accuracy of Various Algorithms



The Fig 1 and Table 2 indicate that our CNN algorithm outperforms both machine learning and normal algorithm based techniques for sentiment analysis in terms of model accuracy.

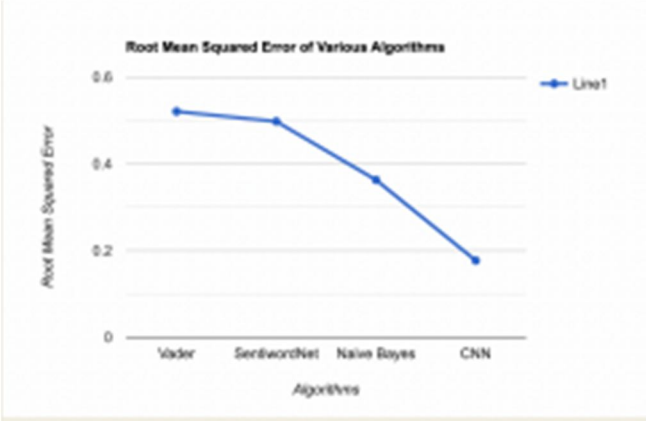
Table 3: Mean Absolute Error			
Vader	Sentiword Net	Naive Bayes	CNN
0.448	0.419	0.271	0.084

Fig 2: Mean Absolute Error of Various Algorithms

The Fig 2 and Table 3 indicate that our CNN algorithm outperforms both machine learning and normal algorithm based techniques for sentiment analysis.

Table 4: Root Mean Squared Error			
Vader	SentiwordNet	Naive Bayes	CNN
0.521	0.498	0.363	0.177

Fig 3: Root Mean Squared Error of Various Algorithms



The Fig 3 and Table 4 indicate that our CNN algorithm outperforms both machine learning and normal algorithm based techniques for sentiment analysis.

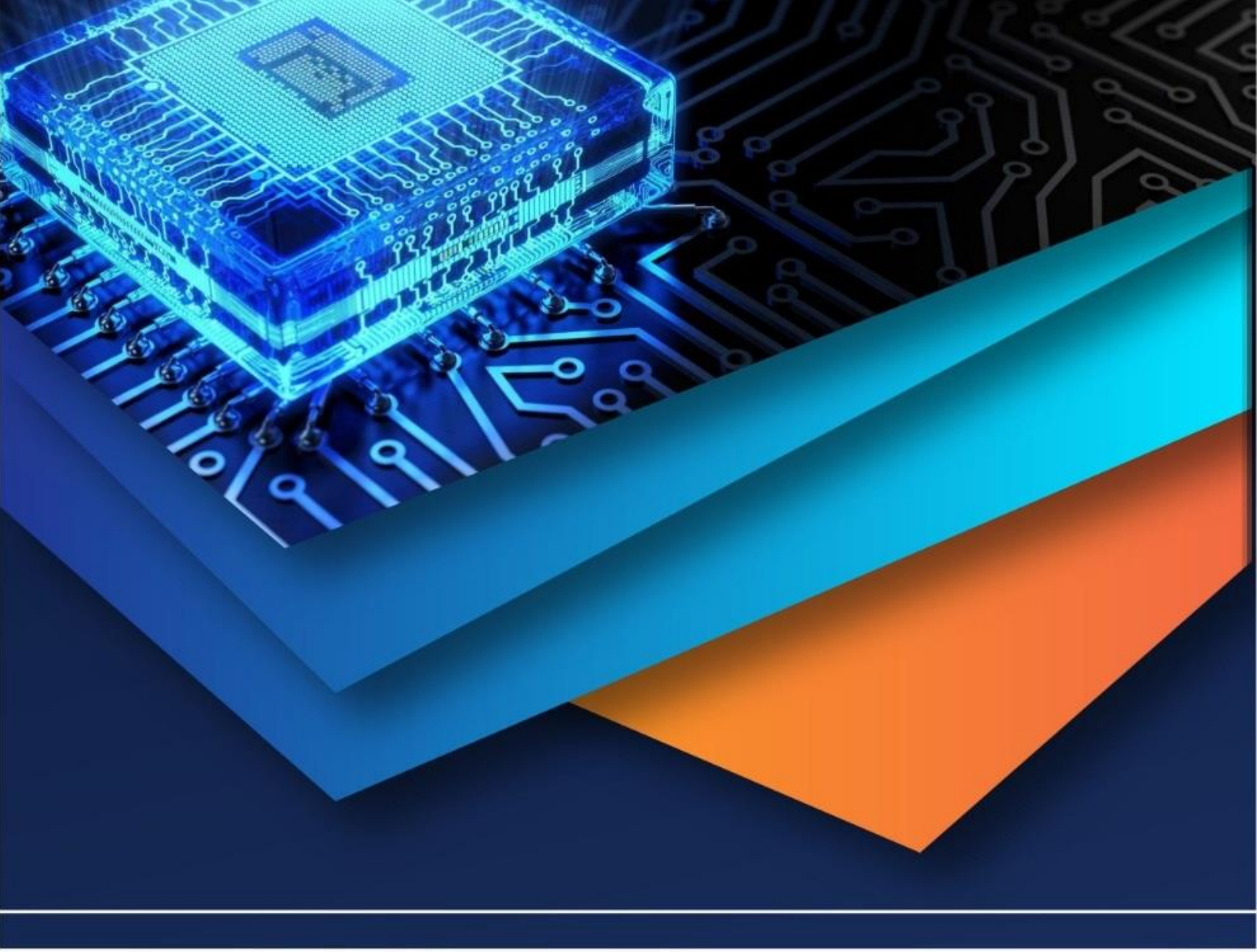
VI. CONCLUSIONS

This research was conducted with various algorithms and various techniques were used to determine the emotion and sentiments by deducing the polarity of the corpus (textual dataset). We took use of algorithms like Vader, SentiwordNet, Naive Bayes and CNN. Out of all these algorithms the one which came on top giving the best results was the CNN classifier with the accuracy of 91.30%. As only less algorithms and methodologies were tested against our dataset, it is required to test other ways or create hybrid methods from different algorithms so that accuracy and efficiency of the our model can be increased.

Finding the sentiment of the text data can help in various domains. Smart technologies can be developed which can give the users comprehensive analytics about reviews of books, services items, and more without creating a need for them to go and study each reviews one at a time, they can directly take decisions based on the data and analytics given by smart algorithms.

REFERENCES

[1] S. E. Saad and J. Yang, "Twitter Sentiment Analysis Based on Ordinal Regression," IEEE Access, vol. 7, pp. 163677-163685, (2019)
[2] Y. Fang, H. Tan and J. Zhang, "Multi-Strategy Sentiment Analysis of Consumer Reviews Based on Semantic Fuzziness," IEEE Access, vol. 6, pp. 20625-20631, (2018)
[3] M. Afzaal, M. Usman and A. Fong, "Tourism Mobile App With Aspect-Based Sentiment Classification Framework for Tourist Reviews," IEEE Transactions on Consumer Electronics, vol. 65, no. 2, pp. 233-242, (2019)
[4] ParamitaRay, and AmlanChakrabarti, "A Mixed approach of Deep Learning method and Rule-Based method to improve Aspect Level Sentiment Analysis", Applied Computing and Informatics, (2019)
[5] Joscha Markle-Huß, Stefan Feuerriegel, Helmut Prendinger, "Improving Sentiment Analysis with Document Level Semantic Relationships from Rhetoric Discourse Structures", Proceedings of the 50th Hawaii International Conference on System Sciences, (2017)
[6] A. Hogenboom, F. Frasincar, F. de Jong, and U. Kaymak, "Using Rhetorical Structure in Sentiment Analysis", Communications of the ACM, vol. 58, no. 7, pp. 69–77, (2015)
[7] FengXu, ZhenchunPan, and RuiXia, "E-commerce product review sentiment classification based on a naïve Bayes continuous learning framework", Information Processing & Management, (2020)
[8] MohammadAl-Smadi, OmarQawasmeh, MahmoudAl Ayyoub, YaserJararweh, and BrijGupta, "Deep Recurrent neural network vs. support vector machine for aspect based sentiment analysis of Arabic hotels' reviews", Journal of Computational Science, vol. 27, pp. 386-393, (2018)



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY