A Comparative Study of Twitter Sentiment Analysis Using Lexicon-based and Machine Learning Techniques on Environmental, Political and Economical Issues

Manideep Reddy KANCHERLA, Mamadou Yaya Cherif DIALLO, Nu Uyen Thi PHAN
Tutor: Themis PALPANAS
Université Paris Cité, Paris, France
April 2022

***Abstract* – Social networking services became an integral part of our society's existence. The amount of data that's being generated everyday contains every bit of our lives. In this context, sentiment analysis plays a key role in various different industries in order to grasp people's opinions and emotions on a person or an entity through text or through visual data. This report conducted a comparative study of sentiment analysis using lexicon-based and machine learning algorithms on twitter datasets of various current issues. We developed a detailed methodology in order to train models or predict sentiments. On one hand, we tested various algorithms for machine learning but decided on Naive Bayes based algorithms: ComplementNB, BernoulliNB and MultimonialNB along with logistic regression. On the other hand, we used VADER rule-based sentiment analysis tool for lexicon-based approach. This report discusses in detail all the above mentioned topics and provides a methodology with excellent accuracy for twitter text classification.**

***Keywords* – Machine Learning, Sentiment Analysis, Lexicon, Twitter, Logistic Regression, Naive Bayes, US airline, Indian Elections, Climate Change**

INTRODUCTION

Social networking services such as Twitter, Facebook and Whats-app have seen tremendous growth during the last decade and have become an integral part of today's social, political and economical landscape. Most importantly, Twitter has become a hub in today's world to discuss these issues. Therefore, understanding people's behaviour, emotions and opinions on these social networking services has become a key industry.

Sentiment analysis is the analysis of people's thoughts, feelings and opinions towards a certain individual, a group or an entity. It is a computational study which aims to determine whether these opinions are positive, neutral or negative. For example, Uniphore, one of the fastest growing automation companies on the planet, is developing an image-based system called Q for Sales which can deduce a potential customer's sentiment towards products mentioned by a salesperson during virtual meetings using advanced computer vision, speech recognition, natural-language processing and emotion AI to pick up on behavioral cues of the client [1]. While this raises a lot of potential privacy issues, this has the potential to create an industry with a market size of US$4.3 billion dollars in 2027 from US$1.6 billion dollars in 2020 [2]. While image-based and text-based sentimental analysis might need different methodologies, the essence remains the same in both the cases. In our project, we did a comparative study of twitter sentiment analysis using lexicon-based and machine learning techniques on multiple datasets covering the current environmental, political and economical issues. This helped us understand the sentiment towards today's problems on the most intense opinion sharing social networking service.

Twitter sentiment analysis became an even more important and interesting topic during and after COVID due to the surge in usage and political landscape. The world has never been this divided over various social, political and economic issues, which was further amplified by the the ascent of social networking services that led to people feeling comfortable expressing their opinions on a wide array of topics regardless of experience and expertise. In short, Twitter has become a place where these issues or discussions form microcosms and processing the data generated through these microcosms for sentiment analysis can help us gain vast amounts of knowledge.

Just like in real world, understanding and discerning human emotions through a computational study is a very complex task. The process of understanding human languages is complicated due to the cultural, social and individual diversity. This makes the process of training and detecting grammatical nuances even more difficult. For example, sarcasm is something where the sentiment can't be properly defined like love and hate [3]. The fundamental problem is that it is impossible to contextualize text due the immense amount of data that is scattered and inconsistent. Therefore, the core task of identifying the corresponding sentiment of any given text is very difficult. This project aims to establish a proper methodology to tackle this problem and compare the different techniques used to detect and classify sentiments.

In Section 1, we started with a comprehensive literature review to understand the subject better and try improving the existing techniques. Next, in Section 2, we defined in detail the problems that we aim to solve. In Section 3, we will describe the different methodologies. Lastly, in Section 4, we presented our experimental results along with a detailed summary of the comparative analysis.

I. Literature Review

During the course of our project we read various scientific articles to better understand three core ideas: Sentiment Analysis, Lexicon-based techniques and Machine Learning Techniques. We started with a wider view of the topic by reading about Sentiment Analysis in general and then we read articles that used Lexicon-based techniques and Machine Learning techniques for twitter sentiment analysis.

One of the first studies on feelings/opinions appeared 20 years back [4]. In their article, B.Pang et al. [4], concluded that sentiment analysis based on machine learning techniques was more accurate than human-generated baseline. However, their dataset was very small and the techniques used in their paper were limited to three standard machine learning algorithms. In 2019, Li et al. [5] developed an advanced system called TweetSenti to analyze the sentiments of entities in a sentence. They also developed a web application to allow the users to use TweetSenti in real time to analyze the various entities. This comparison shows the evolution of various approaches we can use today compared to 20 years ago for sentiment analysis. Therefore, in our project, we worked with various machine learning algorithms and a lexicon-based technique on datasets of every size for a comparative analysis.

A. Machine Learning Techniques

In 2021, Villavicencio et al. [6] worked on COVID-19 dataset acquired through twitter in Philippines to understand the Filipino people's sentiment towards their government's handling of COVID-19. They assigned a sentiment to the tweets using RapidMiner data science software. In order to do the classification, they used the standard Naive Bayes algorithm on dataset of 993 tweets post data preparation and preprocessing. They reported that the accuracy they managed to obtain is higher than other sentiment analysis articles for COVID-19 in the same period. Naive Bayes is the standard algorithm used for sentiment analysis using machine learning. However, the authors don't introduce a novel approach to sentiment analysis but rely on a standard Naive Bayes algorithm. Moreover, their dataset was relatively small in size. In 2017, Song et al. [7] developed a novel classification approach based on Naive Bayes algorithm that doesn't use the same number of attributes to estimate the weight of each class and excludes uncountable and meaningless attributes. They compared their novel approach to Multinomial Naive Bayes and Multivariate Naive Bayes algorithms, and demonstrated that their approach significantly increases the accuracy. Their novel approach puts emphasis on two methods to enhance the performance of Naive Bayes, feature selection and attribute weighting. They evaluated their classification model on a dataset of 1.6 million tweets (equal amounts of positive and negative tweets) and used further subsets of training data: 10000, 20000, 30000, 40000, 50000 and 70000. They were able to demonstrate their approach's significant advantage over Multinomial Naive Bayes and Multivariate Naive Bayes but also a bit of improvement compared to feature selection and attribute weighting. However, their comparison with results obtained in other scientific articles isn't completely valid because the size and the context of the training and test sets vary a lot. In our project, we used multiple machine learning algorithms and a rule-based lexicon on 4 different datasets of varying sizes to compare the different methods.

Villavicencio et al. [6] used a dataset with a mix of English and Filipino tweets to classify them into positive, negative and neutral tweets. In 2022, Al-Hashedi et al. [8] worked on people's sentiments towards COVID-19 with the dataset containing all the tweets in Arabic. They used machine learning model to analyze the tweets in Arabic. In this model, they used Word2Vec for word embedding, two pretrained continuous bag-of-words(CBOW) and Naive Bayes as a baseline classifier. Once the baseline was established they used other machine learning algorithms to find the optimal accuracy. Even though all the tweets in our datasets were in English, these articles were an impportant reference in terms of prerprocessing and machine learning techniques.

B. Lexicon-based techniques

In 2021, Alvinika et al. [9] worked on helping analyze vaccine sentiment of Indonesian population during COVID-19. To accomplish this, the authors used lexicon-based method to determine the polarity of the tweets but they didn't measure the accuracy score. In our project, we used the vader lexicon to determine the polarity of the tweets and then compare them to the target sentiments to find the accuracy score.

In 2022, Dashtipour et al. [10] described the improvement the authors made to the already existing sentiment lexicon to facilitate the sentiment analysis of texts in Persian since the majority of articles today are dominated by English. The authors recognised that the the first version of PerSent-based sentiment analysis approach fails to classify the real-world sentences with idiomatic expressions. Therefore the authors added 1000 idiomatic expressions to the dataset. After the classification, the results were promising. While we didn't get to work on other languages or make lexicons, it was interesting to read the innovative solution they came up with for sentiment analysis in Persian.

II. Problem Formulation

The core problem we addressed concerned studying 4 different datasets with a unique problematic each using 4 machine learning algorithms and 1 lexicon-based twitter sentiment analysis. Moreover, using the natural language processing libraries, we were able to conduct a thorough data preprocessing to obtain the best possible results. Our results were a comparative study of the different techniques used along a detailed discussion on the eventual conclusion.

The next problem was regarding the analysis for each dataset. Each of the dataset concerns an active political or social or economical issue along with a dataset to create a baseline. We made our own conclusions through the results obtained by each of these datasets.

In the grand scheme of things, we searched for the best possible methodology in order to get the most optimal accuracy through a comparative analysis in order to identify people's emotions in bubbles across all the social media

platforms.

III. Methodology

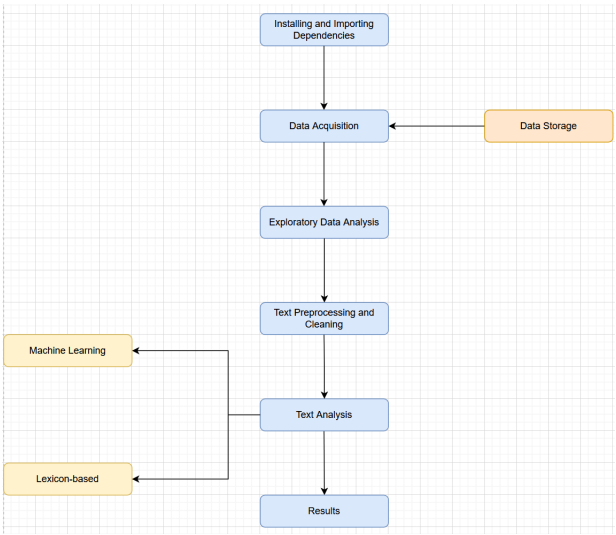

Fig. 1. Flowchart representing the methodology

Firstly, we started by installing and importing all the necessary dependencies. Secondly, we uploaded the data using an external database. Thirdly, we conducted an exploratory data analysis. Next, we proceeded to do the text preprocessing and cleaning. Subsequently, we did the text analysis using machine learning and lexicon-based approaches.

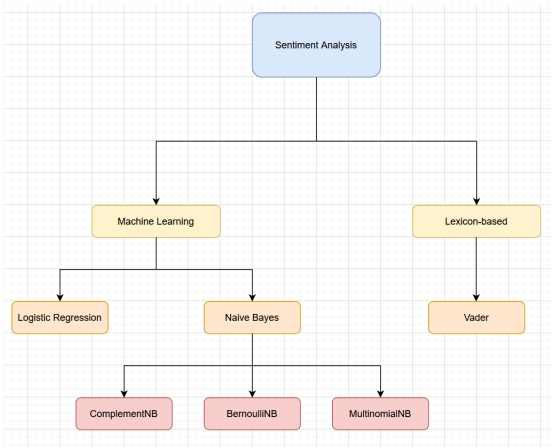

Fig. 2. Flowchart representing the methodology

Lastly, we presented the results obtained after training the models and predicting sentiments.

A. Installing and Importing Dependencies

Here are some of the important dependencies:

- **pandas**
- **Seaborn**
- **sqlite3**
- **nltk (Natural Language Toolkit**
- **sklearn**

B. Data Acquisition and Data Storage

All the 4 datasets were stored in a database using sqlite, which is a database engine. Once the database was uploaded on drive, we mounted the drive to google colab and imported the datasets directly from the sqlite database.

C. Exploratory Data Analysis

At this step, we used data visualisation to better understand the structure and content of the data before proceeding to preprocessing. This helped us better summarize the main characteristics of the dataset and make the next steps more comfortable.

D. Text Preprocessing and Cleaning

This step was one of the most important steps since it had a significant impact on the results. All the functions used for tweet text preprocessing belonged to nltk. Here's a brief introduction to some of them:

- **Stopwords**: These are words that can be safely ignored as they don't change or add meaning to a sentence. Ex: "in", "a", "the"
- **Cleaning Punctuations**: This function was used to clean all the punctuation marks. Ex: "!", "?"
- **Cleaning URLs**: With the help of this function, we were able to remove links starting with "http" or "www".
- **Stemming**: Stemming is used to reduce a word to it's root form and to remove the suffix form this word.
- **Lemmatizing**: It is similar to stemming but it adds context to the words. It links words with similar meaning to one word.

The next step was to split the data into X_train and X_test for machine learning. Then, we transformed the collection of raw tweet text into a vector on the basis of the frequency of each word that exists in the collection of tweet text. This is to help the machine learning algorithms understand the text as a feature.

E. Text Analysis

This was the most crucial part of the methodology where we used machine learning algorithms to classify the data and lexicons to predict the sentiments.

1) Machine Learning Approach

Machine Learning helps us predict the outcome without being programmed to do so by using historical data as input. We used a total of 12 different algorithms on the first dataset which established a baseline. Later, we decided on 4 algorithms which might not be all at the top of accuracy but they consume way less time and processing power. Here are the 4 (actually it's 2 algorithms since three of them are part of Naive Bayes):

- **Logistic Regression**: Logistic Regression is a supervised machine learning algorithm based on probability. It is a predictive analysis algorithm used for classification tasks. In

logistic regression, the dependant variable is a binary variable (0 or 1).

• **Naive Bayes**: This is a supervised learning algorithm based on the Bayes theorem and mainly used for classification tasks. It is a probabilistic classifier used mainly for text classification and high-dimensional datasets [11].

$$P(y \mid x_1, \ldots, x_n) = \frac{P(y) \prod_{i=1}^{n} P(x_i \mid y)}{P(x_1, \ldots, x_n)}$$

Fig. 3. Bayes theorem using the naive conditional Independence assumption with class variable y and dependent feature vector x1 through xn.

Here are the three types of Naive Bayes algorithms used:

• **Multinomial Naive Bayes**: This is a frequency based model used as one of the two best text classification algorithms. In the case of text classification, the frequencies of words in a documetn can be manipulated

$$c(d) = \arg\max_{c \in C} \left[\log P(c) + \sum_{i=1}^{m} f_i \log P(w_i|c) \right]$$

Fig. 4. Equation of text classification with Multinomial Naive Bayes with document d composed of a word vector <w1, w2, w3, ... wm>.

• **Bernoulli Naive Bayes**: This implements the naive Bayes classification algorithms for data that has multiple features but each one is assumed to be a binary-valued variable.

$$P(x_i \mid y) = P(i \mid y)x_i + (1 - P(i \mid y))(1 - x_i)$$

Fig. 5. Decision rule for Bernoulli Naive Bayes.

• **Complement Naive Bayes**: This implements the Complement Naive Bayes classification and Complement Naive Bayes is an adaptation of the standard multinomial naive Bayes algorithm mostly suited for imbalanced data.

2) Lexicon-based Approach

Lexicon-based approach involves using an already established lexicon with sentiment scores for words to score a document with text by aggregating the sentiments of all he words [12].

For lexicon-based approach, we used VADER(Valence Aware Dictionary and sEntimentReasoner) which is a lexicon and rule-based sentiment analysis tool. VADER is one of the most used tools for social media based sentiment analysis. Moreover, VADER supports emojis for sentiment analysis and it doesn't suffer from the speed-performance trade off. Vader's output is in the form of a compound score with a positive sentiment if the compound score >= 0.05, neutral sentiment if the compound score > -0.05 and < 0.05, and negative sentiment if the compound score <= -0.05.

3) Results

Once the solution was applied to any of the 4 datasets, we generated the ac curacies from the 4 machine learning algorithms and VADER rule-based lexicon. We used tools like confusion matrix, classification report and graphs to interpret the results and deduce the most pertinent conclusions. On top of it, we also analyzed the datasets individually to try to understand more about the context of the environmental or political or economical datasets.

At the end of the section, we summarized all the results in the form of a table to deduce the most optimal sentiment analysis tool.

IV. Experiments and Discussion

A. Hardware

For the entirety of the sentiment analysis, we used a laptop with AMD Ryzen 7 4800H processor with Radeon Graphics 2.90 GHz and a 16gb RAM.

B. Baseline Dataset - Sentiment140

Sentiment140 **?** is a dataset that started as a class project from Stanford University for sentiment analysis classification. The dataset has 1.6 million tweets in total with two important columns: target and text. The target column represents the polarity or the sentiment of the tweet with 0 representing a negative tweet and 4 representing a positive tweet. The text column contains the tweet of every user that we used preprocessed and cleaned to build a classification model.

We used Sentiment140 as a baseline to select the algorithms and parameters for sentiment analysis. We tested a total of 10 machine learning algorithms and 2 (TextBlob and VADER) lexicon-based tools. We selected 4 algorithms and 1 lexicon-based tool respectively after taking into account the time taken to execute large datasets, load on the performance and the accuracy.

1) Exploratory Data Analysis

At the very beginning, we changed the negative and positive values to -1 an 1 respectively. Both the categories occupied 50% of the total dataset each. Here's a graph constructed using seaborn:

2) Text Preprocessing and Cleaning

As described in the solutions section, the next step was to go through text preprocessing and data cleaning. Here's an example of data preprocessing of the first 5 lines of the text column:

As you can see, all of the symbols, words, numbers which didn't bring any context to the sentence and might have affected the model training were eliminated. This helped us improve the accuracy by at least double digits.

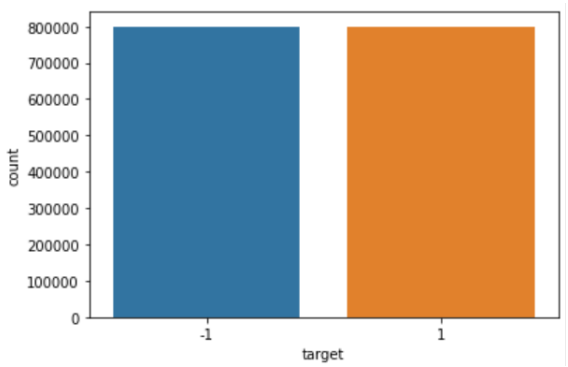

Fig. 6. Graphic representation of the distribution of negative and positive sentiments.

```
0    @switchfoot http://twitpic.com/2y1zl - Awww, t...
1    is upset that he can't update his Facebook by ...
2    @Kenichan I dived many times for the ball. Man...
3       my whole body feels itchy and like its on fire
4    @nationwideclass no, it's not behaving at all....
Name: text, dtype: object
```

(a)

(b)

```
0    switchfoot httptwitpiccomzl  A s bummer You sh...
1    upset cant update Facebook texting it might cr...
2    Kenichan I dived many times ball Managed save ...
3                    whole body feels itchy like fire
4    nationwideclass no behaving all im mad here I ...
Name: final_text, dtype: object
```

Fig. 7. (a) Tweet text before text preprocessing and data cleaning, (b) Tweet text after text preprocessing and data cleaning.

3) Text Analysis

The files were split into 80-20 for training and testing respectively.

• **Machine Learning Approach:** The accuracy results of the 4 machine learning algorithms can be found in Fig. 8.

```
Train Accuracy using BernoulliNB:  0.82439765625
Test Accuracy using BernoulliNB:  0.776165625
Train Accuracy using Logistic Regression:  0.853540625
Test Accuracy using Logistic Regression:  0.784453125
Train Accuracy using MultimonialNB:  0.83874609375
Test Accuracy using MultinomialNB:  0.773340625
Train Accuracy using ComplementNB:  0.838746875
Test Accuracy using ComplementNB:  0.77338125
```

Fig. 8. Results - Machine Learning.

The optimal algorithm for classification with the highest train and test accuracy score was Logistic Regression with a training score of 0.85 and testing score of 0.78. Logistic regression is used to solve binary classification tasks and is a very good entry point to text classification.

Next, we generated the classification report and the confusion matrix (Fig. 9 and 10):

As you can see in Fig. 9 and 10, the precision and recall values of negative and positive occurrences was almost the same. We can conclude that the model was consistent in terms of it's judgement on whether a text was positive or negative.

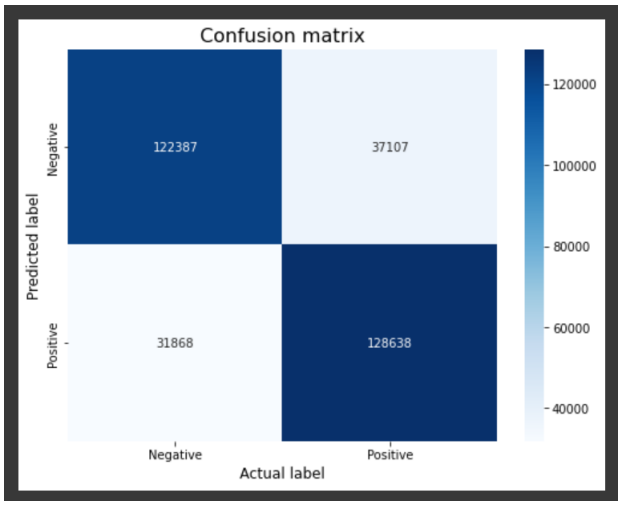

Fig. 9. Confusion Matrix for logistic regression using Sentiment140.

Classification Report:				
	precision	recall	f1-score	support
-1	0.79	0.77	0.78	159494
1	0.78	0.80	0.79	160506
accuracy			0.78	320000
macro avg	0.78	0.78	0.78	320000
weighted avg	0.78	0.78	0.78	320000

Fig. 10. Classification report for logistic regression using Sentiment140.

As for the scores itself, any score around 0.80 is a relatively good score in terms of text analysis. Moreover, the f1-score was around 0.79 which wasn't far from 1. Therefore, the excepted model performance should be acceptable. However, this also proved to us that we could have improved the preprocessing step a bit more.

• **Lexicon-based Approach:** After using VADER, we obtained an accuracy of about 0.52. However, there was a problem with the synchronisation of our target values and the sentiment values generated by VADER. Let's have a look at the classification report and the confusion matrix:

As you can see in Fig.11, the confusion matrix showed a relatively healty distribution of positive and negative values. However, the problem was apparent in the classification report. VADER generated three different sentiments, in which neutral was always 0 and it wasn't possible to judge if a neutral leans more towards positive or negative. Therefore, this resulted in significant error in the accuracy score.

At last, we concluded that the results of machine learning techniques (Logistic Regression) were better than lexicon-based analysis.

C. Environmental Dataset - Climate Change

This particular dataset was funded by a Canada Foundation for Innovation JELF Grant to Chris Bauch, University of Waterloo. It contained fours kinds of target labels: 1 for positive (belief in man-made climate change), 0 for neutrality,

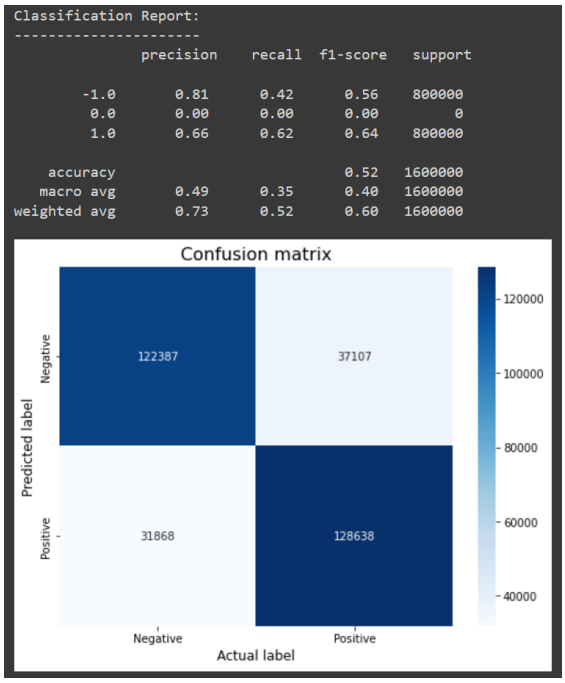

Fig. 11. Confusion Matrix and Classification report for Vader.

-1 for negative (not believing in man-made climate change) and 2 for news. The dataset contained around 44000 tweets.

1) Exploratory Data Analysis

By looking at the raw numbers, we deduced that a lot more people believed in man-made climate change followed by news then people who are neutral and people who don't believe in man-made climate change.

Here's a graph constructed using seaborn:

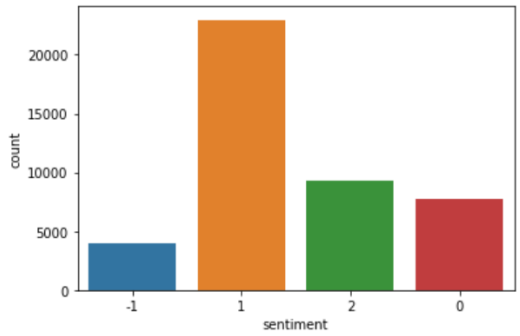

Fig. 12. Graphic representation of the distribution of sentiments.

2) Text Analysis

The files were split into 80-20 for training and testing respectively.

• **Machine Learning Approach:** Here were the accuracy results of the 4 machine learning algorithms used:

The optimal algorithm for classification with the highest train and test accuracy score was again logistic regression with a training accuracy of about 95% and testing accuracy of about 74%. While the training accuracy was outstanding, the gap

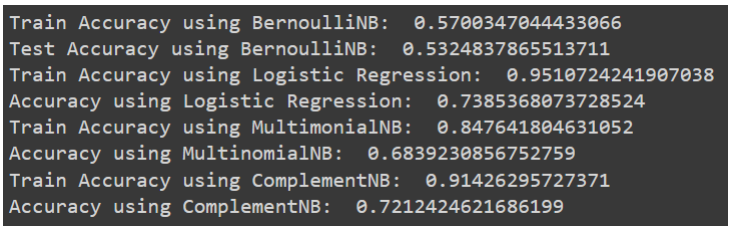

Fig. 13. Results - Machine Learning.

between both the accuracy was a cause for concern.

Next, we generated the classification report and the confusion matrix in Fig.14.

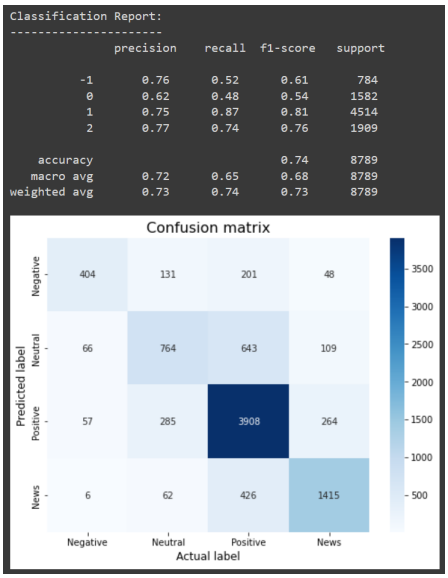

Fig. 14. Confusion Matrix and Classification Report.

As you can observe in Fig.14, the inconsistencies between the results obtained. The positive sentiment has the best f1 score followed by news, while the remaining two are pretty mediocre. Therefore, the model worked better with positive and factual tweets than the negative and neutral tweets. This might be due to the contexts of the negative and neutral tweets being harder to understand for the model, or there might have been some inconsistencies during text preprocessing and cleaning. However, the model worked pretty well in terms of identifying positive emotions.

• **Lexicon-based Approach:** After using VADER, we obtained an accuracy of about 0.28. However, there was still the same problem with the synchronisation of our target values and the sentiment values generated by VADER. Let's have a look at the classification report and the confusion matrix:

The results were pretty bad for every sentiment except the positive sentiment towards climate. However, this might be due to the relatively high number of positive sentiments. Our model was able to recognise with a higher probability the sentiment of man-made climate change.

At last, we concluded that the results of machine learning techniques (Logistic Regression) were way better than lexicon-based analysis.

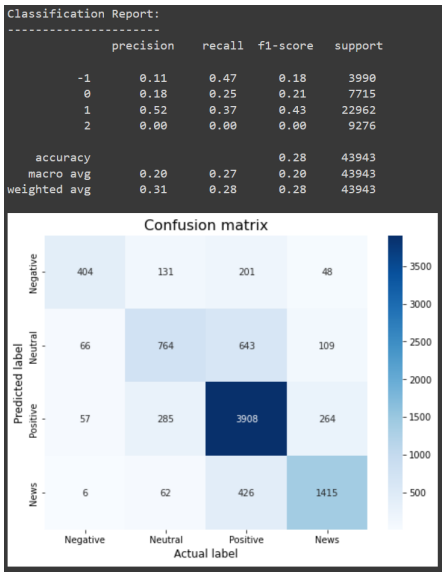

Fig. 15. Confusion Matrix and Classification Report.

D. Political Dataset - Indian Elections 2019

This dataset recorded the tweets discussing people's opinions on the prime minister and other leaders of India during the 2019 Lok Sabha elections. It contained three kinds of target labels: 1 for positive, 0 for neutrality, and -1 for negative. The dataset contained around 163000 tweets.

1) Exploratory Data Analysis

Here's a graph constructed using seaborn:

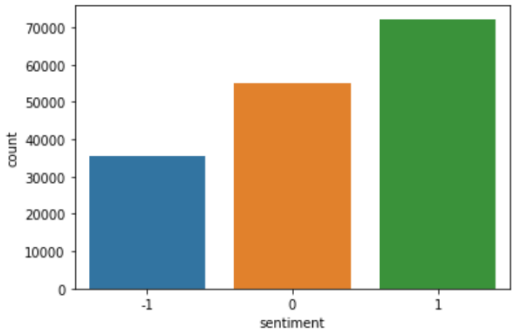

Fig. 16. Graphic representation of the distribution of sentiments.

The large majority of the people had either a neutral or a positive attitude towards political leaders.

2) Text Analysis

The files were split into 80-20 for training and testing respectively.

• **Machine Learning Approach:** Here were the accuracy results of the 4 machine learning algorithms used:

The optimal algorithm for classification with the highest train and test accuracy score was again logistic regression with a training accuracy of about 94% and testing accuracy of about 90%. This was the best result to date.

Next, we generated the classification report and the confusion matrix in Fig.14.

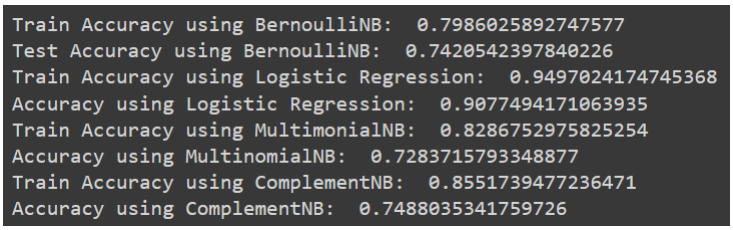

Fig. 17. Results - Machine Learning.

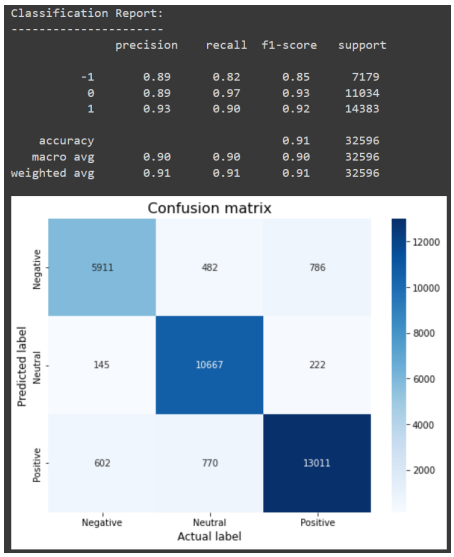

Fig. 18. Confusion Matrix and Classification Report.

As you can observe in Fig.14, the results were almost perfect with f1 score being close to 1 for almost the 3 cases.

• **Lexicon-based Approach:** After using VADER, we obtained an accuracy of about 57%. Even though, this was the best score until now, it was still not acceptable.

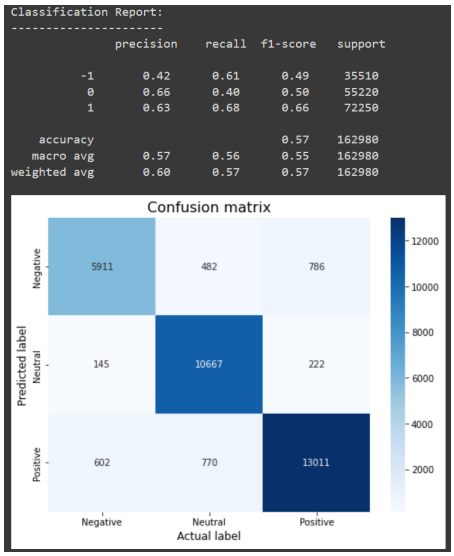

Fig. 19. Confusion Matrix and Classification Report.

The results were pretty bad for every sentiment except the positive sentiment towards leaders. However, this might be due to the relatively high number of negative sentiments Our model was able to recognise with a bit higher probability the

positive sentiment people have toward Indian political leaders.

At last, we concluded that the results of machine learning techniques (Logistic Regression) were way better than lexicon-based analysis.

E. Economical Dataset - US Airlines

This dataset recorded the tweets discussing people's opinions of US airline companies. It contained fours kinds of target labels: 1 for positive, 0 for neutrality, and -1 for negative. The dataset contained around 14700 tweets.

1) Exploratory Data Analysis

Here's a graph constructed using seaborn:

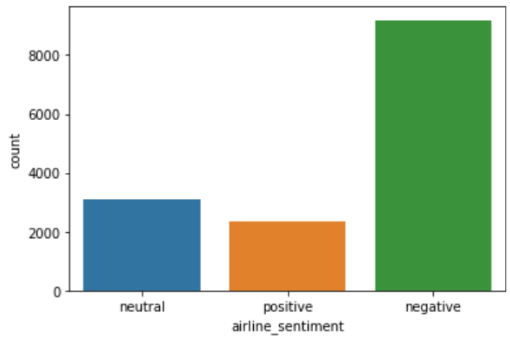

Fig. 20. Graphic representation of the distribution of sentiments.

The large majority of the people had a negative sentiment towards the US airlines.

2) Text Analysis

The files were split into 80-20 for training and testing respectively.

• **Machine Learning Approach:** Here were the accuracy results of the 4 machine learning algorithms used:

```
Train Accuracy using BernoulliNB:  0.7098702185792349
Test Accuracy using BernoulliNB:  0.682035519125683
Train Accuracy using Logistic Regression:  0.9236680327868853
Accuracy using Logistic Regression:  0.8155737704918032
Train Accuracy using MultimonialNB:  0.8505806010928961
Accuracy using MultinomialNB:  0.780396174863388
Train Accuracy using ComplementNB:  0.8769637978142076
Accuracy using ComplementNB:  0.7937158469945356
```

Fig. 21. Results - Machine Learning.

The optimal algorithm for classification with the highest train and test accuracy score was again logistic regression with a training accuracy of about 92% and testing accuracy of about 81%.

Next, we generated the classification report and the confusion matrix in Fig.14.

As you can observe in Fig.14, the results were mediocre and the negative sentiments had the highest f1 score.

• **Lexicon-based Approach:** After using VADER, we obtained an accuracy of about 55%. Another mediocre score generated by lexicon-based approach.

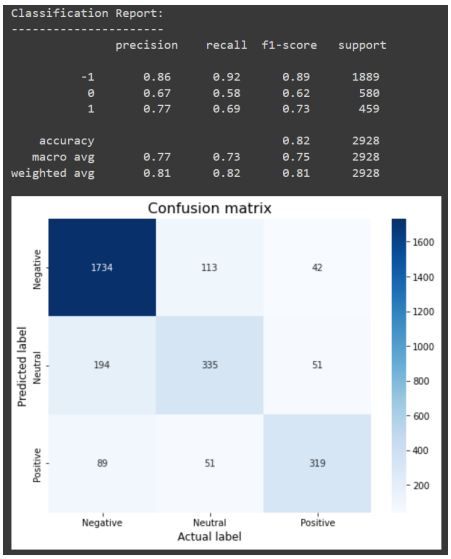

Fig. 22. Confusion Matrix and classification report.

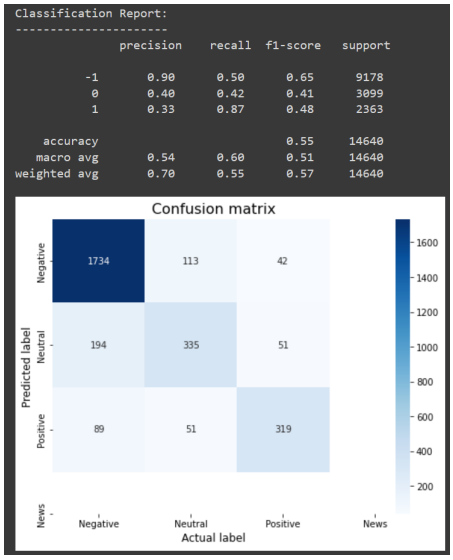

Fig. 23. Confusion Matrix and classification report.

The results were pretty bad for every sentiment except the negative towards the airlines. However, this might be due to the relatively high number of negative sentiments.

At last, we concluded that the results of machine learning techniques (Logistic Regression) were way better than lexicon-based analysis.

F. Results Summary

We concluded after analyzing the results of the 4 datasets that logistic regression was the best machine learning sentiment analysis algorithm used during this project. The results obtained by using VADER were pretty disappointing. We deduced that this might be due to problems with dimensionality and text preprocessing.

One more very important thing to remember was that the datasets were all of different sizes, the training and testing

data are also of different sizes. Therefore, we were only able to use these results as a reference.

Here's the final table with the accuracy of different datasets from the best machine learning algorithms (Logistic Regression) and VADER:

Accuracy of the Datasets in Accordance to the Sentiment Analysis Tool					
	Datasets	Sentiment140	Climate Change	Indian Elections 2019	US Airlines
Logistic Regression	Training Data	85%	95%	95%	92%
	Testing Data	78%	74%	91%	82%
VADER		52%	28%	57%	55%

Fig. 24. Results Summary.

‘

Conclusion

During this project we were able to do an extensive literature review to familiarize ourselves with the subject. Then, we proposed the problem we were planning on solving. Followed by, a detailed methodology on how to use two different sentiment analysis approaches to train a model or predict sentiments. In the end, we were able to obtain interesting results that helped us better understand the challenges and the progress still needed to be done in the field of sentiment analysis.

While we were able to propose an interesting methodology using logistic regression, there is still a lot that can be done . The next step in this project might be to enlarge the scope of study and explore other deep learning algorithms, expand towards sentiment analysis of visual data, or exploring other interesting lexicons like TextBlob and AFINN.

REFERENCES

[1] Kate Kaye, "Companies are using AI to monitor your mood during sales calls. Zoom might be next."

[2] "Global Sentiment Analysis Software Industry" "https://www.globenewswire.com/news-release/2021/01/05/2153121/0/en/Global-Sentiment-Analysis-Software-Industry.html",

[3] Ane Berasategi,"Sarcasm detection with NLP"

[4] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? Sentiment Classification using Machine Learning Techniques. In Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP 2002), pages 79–86. Association for Computational Linguistics.

[5] Quanzhi Li, Qiong Zhang, and Luo Si. 2019. TweetSenti: Target-dependent Tweet Sentiment Analysis. In The World Wide Web Conference (WWW '19). Association for Computing Machinery, New York, NY, USA, 3569–3573. DOI:https://doi.org/10.1145/3308558.3314141

[6] Villavicencio, C.; Macrohon, J.J.; Inbaraj, X.A.; Jeng, J.-H.; Hsieh, J.-G. Twitter Sentiment Analysis towards COVID-19 Vaccines in the Philippines Using Naïve Bayes. Information 2021, 12, 204. https://doi.org/10.3390/info12050204

[7] J. Song, K. T. Kim, B. Lee, S. Kim and H. Y. Youn, "A novel classification approach based on Naïve Bayes for Twitter sentiment analysis," KSII Transactions on Internet and Information Systems, vol. 11, no. 6, pp. 2996-3011, 2017. DOI: 10.3837/tiis.2017.06.011.

[8] Abdullah Al-Hashedi, Belal Al-Fuhaidi, Abdulqader M. Mohsen, Yousef Ali, Hasan Ali Gamal Al-Kaf, Wedad Al-Sorori, Naseebah Maqtary, "Ensemble Classifiers for Arabic Sentiment Analysis of Social Network (Twitter Data) towards COVID-19-Related Conspiracy Theories", Applied Computational Intelligence and Soft Computing, vol. 2022, Article ID 6614730, 10 pages, 2022. https://doi.org/10.1155/2022/6614730

[9] Alvinika, Y., Prasetyo, W., Mudjihartono, P. (2022). Analysis of Twitter User's Sentiment Against COVID-19 Vaccination Using the Lexicon Based Method. In: , et al. Hybrid Intelligent Systems. HIS 2021. Lecture Notes in Networks and Systems, vol 420. Springer, Cham. https://doi.org/10.1007/978-3-030-96305-7

[10] Dashtipour, K., Gogate, M., Gelbukh, A. et al. Extending persian sentiment lexicon with idiomatic expressions for sentiment analysis. Soc. Netw. Anal. Min. 12, 9 (2022). https://doi.org/10.1007/s13278-021-00840-1

[11] sklearn.NaiveBayes, "Naive Bayes

[12] Augustyniak, Ł.; Szymański, P.; Kajdanowicz, T.; Tuligłowicz, W. Comprehensive Study on Lexicon-based Ensemble Classification Sentiment Analysis. Entropy 2016, 18, 4. https://doi.org/10.3390/e18010004