International Journal of Science and Research (IJSR)
ISSN: 2319-7064
Impact Factor 2024: 7.101

Cross-Model Sentiment Analysis of Tweets on the Russia-Ukraine War: A Comparative Study of Lexicon-Based and Transformer Models

Aditya Arora[1], Anand Zutshi[2]

[1]Summit High School
Email: *adityaarora1007[at]gmail.com*

[2]Netaji Subhas University of Technology, Delhi
Email: *123anandji[at]gmail.com*

Abstract: ***This research explores sentiment analysis of social media discourse sur-rounding the Russia-Ukraine war by leveraging two distinct sentiment prediction models - Vader (lexicon-based) and Transformer (deep learning-based). A comprehensive pipeline was developed to extract, preprocess, and classify tweets into positive, negative, and neutral sentiments. Exploratory Data Analysis (EDA), visualization, and clustering techniques were employed to identify key patterns and features across sentiment categories. To enhance the robustness of sentiment classification, various machine learning models, including XGBoost, Random Forest, Support Vector Machine (SVM), Naive Bayes, and Logistic Regression, were trained on the Vader-labeled dataset and subsequently tested on the Transformer-labeled dataset. This cross-model evaluation approach provided insights into the generalizability and consistency of machine learning classifiers across different sentiment annotation techniques. The findings highlight disparities and alignments between lexicon-based and neural network-driven sentiment labeling, shedding light on the reliability and effectiveness of hybrid methodologies for social media sentiment analysis in dynamic geopolitical contexts.***

Keywords: Sentiment Analysis, Social Media, Russia-Ukraine Conflict, VADER, Transformer Models, Machine Learning, Cross-Model Evaluation

1. Introduction

Sentiment Analysis is defined as the process of analyzing and examining an expression or statement to identify the sentiment or feeling of the author (8). This process is an essential tool in natural language processing (NLP), allowing machines to understand human emotions in text. It can be applied in a wide variety of fields, ranging from uses in marketing and business to understand the attitude of consumers, to analyzing the thoughts surrounding certain topics, or to understand how the public feels about certain issues in today's modern world, which is what this research paper aims to do (1). People express their opinions in varying ways, but by far the easiest and most common method is through social media (10). Platforms like X (formerly Twitter) have become digital public squares, where users freely voice their thoughts, making them rich sources of unfiltered sentiment. Living in an age of controversies all the time every day after another, X is known for being a popular platform for people of all political views to express their opinions. We have constant debates online over political issues, and while not all of it is necessarily credible, it does give accurate insights into people's individual opinions. In this study, we took a large dataset of tweets regarding the Russia-Ukraine war with the "russiaukrainewar" hashtag from 2022. This hashtag was one of the most trending and widely used at the time, helping us ensure a substantial volume of data. The reason for choosing the Russia-Ukraine war out of all topics was the attention the situation got, since it was the topic of debate for many months, even today, and also that it is a very polarizing topic (5). With many people of the Democratic party in large support of Ukraine, and Republicans initially in support but now generally opposed to sending aid to countries like Ukraine, this topic provides a wide range of opinions and feelings, thus a holistic "sentiment" around this topic (20). This diversity in viewpoints makes it an ideal case study for comparing sentiment analysis methods. The timing of this dataset might yield slightly different results from what one might expect today, since the public's sentiment regarding the war would be presumably different when it started than after a couple of years. Temporal context is crucial in understanding sentiment trends, and this aspect is also explored in our study. The purpose of this research paper is to use models such as Vader and Transformer to not only analyze the public sentiment around this controversial event when it first took place, and also to take the data and compare and contrast it between Vader and Transformer to understand the differences in the two models. These two models represent fundamentally different approaches to sentiment analysis: Vader is a lexicon-based model, while Transformer relies on deep learning and contextual understanding. We apply the exact same code to both of these models, yet get different results, which help us find the similarities and differences in how these machine learning models work. Some of the biggest questions we aim to answer include how the two models interpret the same data, and what might cause divergence in their outputs. We do this mainly through training a variety of models including XGBoost, Random Forest, SVM, Naive Bayes, and Logistic Regression on the sentiment produced by one of the datasets, and applying it to Transformer to get results such as the precision, recall, f-1 score, and support. These are all metrics that help us understand how accurate these models are, and how accurate the Vader and Transformer models are relative to each other, since the goal is to get the most similar data possible. Our ultimate objective is to understand not just the public sentiment, but also the performance and behavior of different models when applied to complex and nuanced data.

2. Literature Survey

There are many different ways of sentiment analysis using computational algorithms, but the following are a couple of common ways (6):

1) Lexicon-Based Analysis-This type of sentiment analysis uses dictionaries of words associated with positive, negative, and neutral sentiments to generate the overall sentiment of a text or input. It is helpful with simplicity, but is not the most effective with sarcasm for example, or sometimes more in depth and nuanced language.
2) Machine Learning-Based Analysis-This type of sentiment analysis works by taking already existing machine learning models, and training them on datasets to determine sentiments. It is better than lexicon-based analysis given that it can handle more complexity, but requires a lot of data to be manageable.
3) Deep Learning-Based Analysis-These models pull sentiment from text with advanced neural networks. Semantic and Syntactic relationships from the text are learned by the models. It is very accurate, but is by far the most complex method of sentiment analysis.
4) Hybrid Techniques-These basically just combine two or sometimes even more of the above techniques. It allows for simplicity and a reduced level of complexity, but is also hard to implement. Current models like Vader are used in many industries, one example being companies' marketing, especially on social media. This research study uses Vader sentiment to analyze not just the satisfaction, but the level of satisfaction of consumers through taking and analyzing their social media posts online, similar to what we do in this research paper. This research paper, specifically from the Albanian language, created over 950 adjectives and verbs to have an in-depth analysis of sentiments, reaching an accuracy between 89% and 95% (7). This study shows the advancements of the Transformer model, and compared it to AI, and implies that Transformer based models are not just getting better and better, but are also the most effective today; more so that AI models. It analyzes sentiment analysis from almost 500 papers, and finds that the innovation in Transformer based models is often overlooked, which can be attributed to the fact that only 8 of those papers actually used Transformer based models for sentiment analysis in marketing, while most others used common modern AI (4). In other places, sentiment analysis techniques and models have been combined to yield results in areas such as information accessibility. For example, one paper developed TSLA, an incorporation of lexicon based sentiment analyzers to create a deep learning model that provides more accurate results in an easier and more accessible way to users (19). This has spread the use of models into many other industries such as the ones previously described.

3. Problem Definition

This paper investigates how the method of evaluating sentiment, whether that be lexicon based (Vader) or transformer based, impacts the accuracy and consistency of sentiment analysis. Using tweets about the Russia Ukraine conflict, this study compares these approaches by training models on one method and testing on the other to address the relationship between the two models.

4. Methodology

This was a dataset of over 30, 000 posts off of the popular social media app X, formerly "Twitter" (9). All of these tweets contained the keyword "russiaukrainewar" hashtag. There is no set filter for the kind of people this data was taken from, it was taken in a random and unbiased manner from people of every demographic, including people from various parts of the world, from various times, and in various languages to get a holistic look at the sentiment regarding this war. There is a vast timeframe ranging from around the end of 2008 to April of 2022. A large majority of posts would be post February 2022, after the war began, but the Russia Ukraine situation had garnered a lot of attention even before the war started. This has been a geopolitical topic of discussion for many years (15), ever since Ukraine got independence and as a result, our data also analyzes the public sentiment for years building up to the war as well.

All of our data was imported from a dataframe, where it was iterated through and put into a set of arrays for every category including the statement, account name, followers, how many people the account follows, account creation date, verified status, length of tweets, likes, language, retweets, and the date of tweets. Keep in mind that a few of these categories yielded no correlations and were not in any way used in our analysis.

From there, the first process came with the timestamps, where the time of both the tweet and the account creation was turned to a timestamp, a linear representation of time as a simple number. Through looping through the arrays for initial timestamps, a minimum timestamp was found and subtracted from every other timestamp. This was so that all of our timestamps could be shown in a relative manner, relative to the first tweets to make the data a lot easier to analyze. Now, our timestamps' range was a couple of years rather than the start of time.

For the length values, the length was already included in the dataset. It was written as " [0, length]", so all the data was looped through to basically remove the brackets, initial 0, and the comma and a space to follow so that the result would be a simple integer for the length. All the other data was included as it was from the original csv file into our arrays, with just one more category being for the sentiment.

For the Vader sentiment, we used the *analyzer. polarity_scores* command to generate a number as the score (17). This number represented a polarity score, basically showing the "polarity" or division around the statement of the analyzed tweet, and the result was yielded as a number. In the scenario that this number was less than-0.5, then it was considered a negative tweet. In the scenario that this was between-0.5 and 0.5, it was considered neutral. Lastly, a polarity score of above 0.5 meant that the sentiment around this statement or tweet was positive.

For the Transformer sentiment, we used the *sentiment_pipeline* command to generate sentiment like vader (17). Our study tested all scenarios of positive, negative, and neutral data to compare the results of these two datasets.

Transformer results were for now added into an array. All of the initial data taken, the modified data such as the timestamps, and the sentiments with Vader and Transformer were put and exported as two dataframes, one for Vader and another for Transformer, with the only difference being the sentiment column.

Exploratory Data Analysis

1) Reveals the general level of sentiment for positive, negative, and neutral feelings about the war. It proves that the majority of people have neutral posts, around 23 thousand. It is followed by negative tweets at just over 5 thousand, and finally positive tweets fall at just around or less than 2 thousand. It proves that there was a lack of positive sentiment around this war and conflict as a whole.

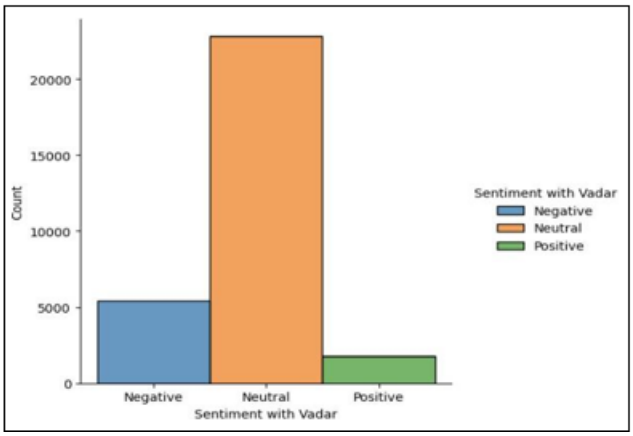

Figure 1: Sentiment distribution of tweets

2) reveals that when talking about the sentiment and the time of tweet, there is a clear decrease in the number of neutral tweets. They keep declining steadily, lower as time goes on. This is probably due to the fact that when the war actually began, it polarised opinions and they were not as neutral anymore. Some of them probably translated negatively, since at one point the negative tweets spike up while the positive one's spike down, but generally the negative and positive tweets say that same in quantity, any increases or decreases seem to be evened out over time.

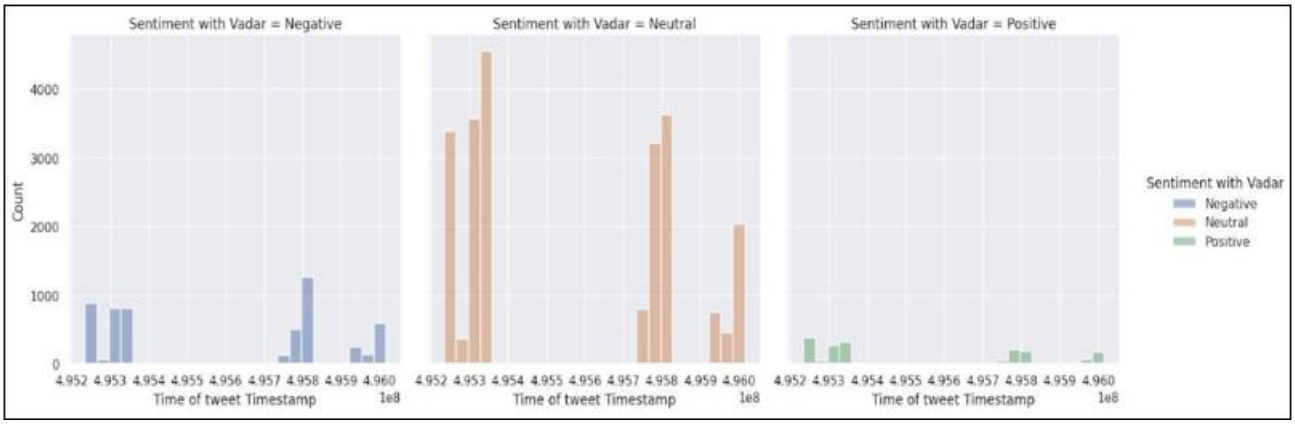

Figure 2: Sentiment distribution of tweets over time

It also shows us when the tweets were most active around this topic. The three periods when tweets were significantly more than at all other times had the relative timestamps of the follow:

- Period 1: 4.9525E8 to 4.9535E8
- Period 2: 4.9575E8 to 4.958E8
- Period 3: 4.9595E8 to 4.96E8

Through a code that reverses the original code of the timestamps, the minimum timestamp was added back to all of the others and then this was converted back to the global time format, yielding the following results.

One of the spikes was 2022 March 26 to March 27. Another spike was 2022 April 1 to April 2, and the other spike was from April 3 to April 4.

The main events that took place during that time were:

- Mach 26-Energy Infrastructure Attacks, Drone Strikes in Dnipro, and Kryvyi Rih Attack (13)
- April 1 to April 4-Discovery of Bucha Massacre (14)

3) reveals that the posts with the most followers have a strong correlation with neutrality. Authors with more followers are most likely to push out neutral tweets, followed by negative tweets. The difference between neutral and negative is definitive, but also minute, so it could be a result of outliers. The positive tweets however prove a very strong fact, and that is that most people with more followers did not put out positive tweets, presumably as a desire not to stand out against the crowd.

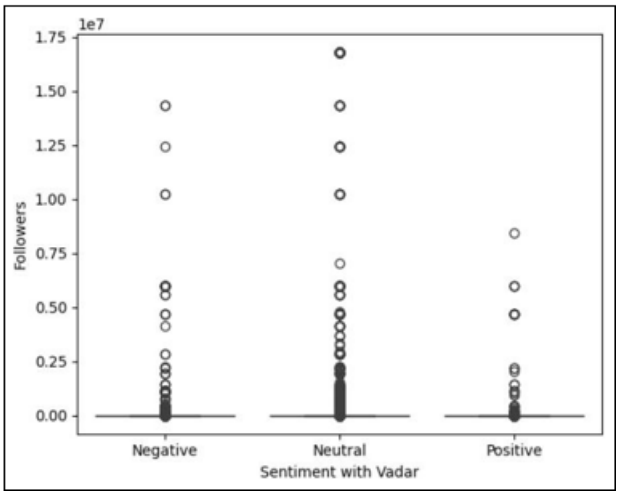

Figure 3: Distribution of sentiment of tweets with number of followers

4) reveals that the most recent sentiment is negative. There is a general equality between neutral and positive posts when it comes to the timeframe, but a noticeable increase in the timestamp for the negative tweets. This increase in the timestamp means that the timestamp is literally a larger number, which translates to recency, revealing that when the war began the sentiment was generally negative.

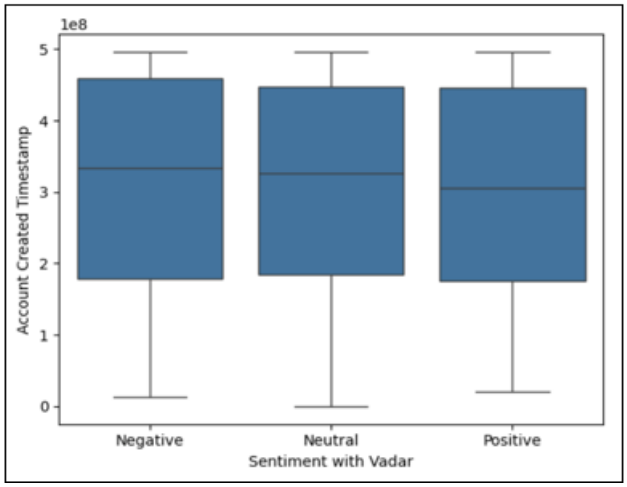

Figure 4: Distribution of sentiment with Account Created Timestamp

5) shows the relationship between sentiment and likes. While they may just be outliers, the graph shows that the most liked sentiments are the neutral ones, followed by the negative ones, and finally the positive ones. While this may just be the result of a few outliers since there are very few tweets that stand out in the first place as a result of likes, it does seem to go with the general trend in this study with the neutral tweets being the most in quantity and the most popular too, followed by the negative ones, and finally by the positive ones.

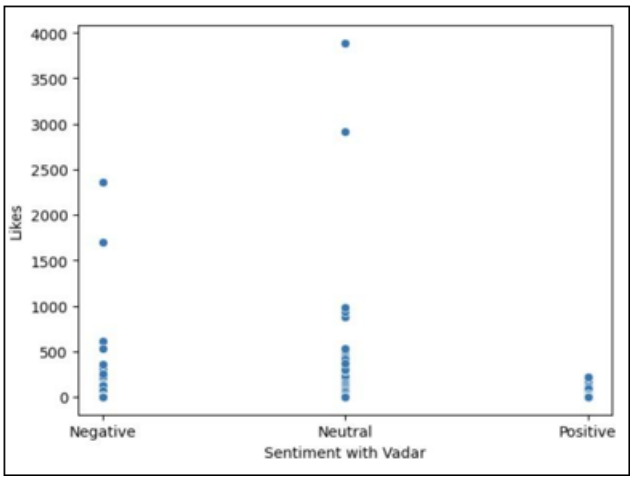

Figure 5: Distribution of sentiment with Likes

Some other factors were not taken into account as mentioned earlier, mainly because they yielded no correlations. When comparing length to likes in 6, it reveals that there was no real correlation between the popularity of the text and the length; rather that these two factors are completely omitted and not given much weight in the paper.

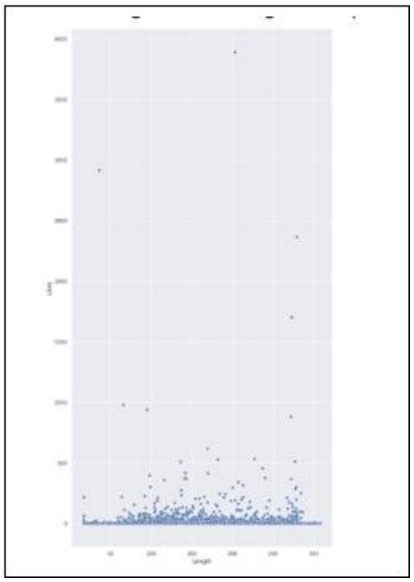

Figure 6: Distribution of Length with Likes

Some of the most common words in this dataset include 'russiaukrainewar', 'http', 'ukraine', 'russian', 'russia', 'war', 'kyiv', 'putin', 'ukrainian', 'euro- maidanpr', 'the', 'ukrainerussiawar', 'near', 'mariupol', 'day', 'march', 'april', 'resident', 'shelling', and 'biden'. 7 is a word cloud consisting of these words to show this visually.

Figure 7: Word cloud distribution of common words

Some of the most common negative words include ‘ukraine’, ‘russiaukrainewar’, ‘war’, ‘http’, ‘russian’, ‘russia’, ‘ukrainian’, ‘day’, ‘update’, ‘today’, ‘front’, ‘strategic’, ‘theater’, ‘focus’, ‘utw’, ‘bluf’, ‘jominiw’, ‘defe. . . ’, ‘putin’, and ‘euromaidanpr’ as shown in 8. Empirically, this shows the large proportion of negative tweets in this dataset, since most of these words are also some of the most common words overall in the dataset.

Figure 8: Word cloud distribution of common negative words

Some of the most common positive words include russiaukrainewar’, ’http’, ’ukraine’, ’russia’, ’putin’, ’russian’, ’ukrainian’, ’amp’, ’support’, ’euromaid-anpr’, ’please’, ’ukrainerussiawar’, ’peace’, ’win’, ’mariupol’, ’like’, ’zelensky’, ’air’, ’make’, and ’important’ as shown in 9.

Figure 9: Word cloud distribution of common positive words

In order to visualize and cluster this information we decided to go for three clusters. This was mainly a result of the fact that we wanted a couple of the more popular tweets to be separated clearly, creating a need for a few clusters. Furthermore, we also had to simplify the data down because we reduced this data down to 2 dimensions, and as a result the simplification was necessary.

For this clustering, we clustered negative and positive words separated by the popularity of their words; each cluster contained 20 common words in that sentiment cluster and were graphed. With K means, the closer the point to the origin the less extreme it tends to be. As a result, we tested out the data, and this was also followed by some revisioning of the data, where we removed most of the relatively meaningless words from the data and clustered again.

Clustering for negative tweets showcased majorly 3 clusters as can be seen in 10. Cluster 1 contained a lot of irrelevant words such as “http”, “near” and “burn”. After cleaning up these words, we plotted the clusters as can be seen in 10. This showcased a single major cluster with a lot of the tweets centralised together.

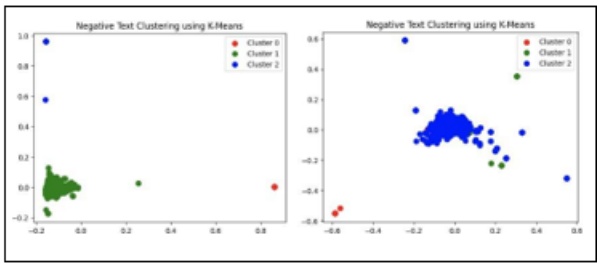

Figure 10: Negative tweet clustering

Clustering for positive tweets also showcased majorly 3 clusters at first with the one major cluster near the origin in 11. The Cluster 0 also had a lot of irrelevant words which needed to be cleaned up. After removal of those words, we got a clean cluster with majorly all the words centralised together in 11.

5. Results and Discussion

For the modelling, we trained different models for the sentiment classification problem. XGBoost is an example of a machine learning algorithm that uses a method called tree boosting at the same time as simplifying data and compressing it to provide an algorithm that gives similar results to other algorithms while requiring far fewer resources and being significantly more efficient in the process (12).

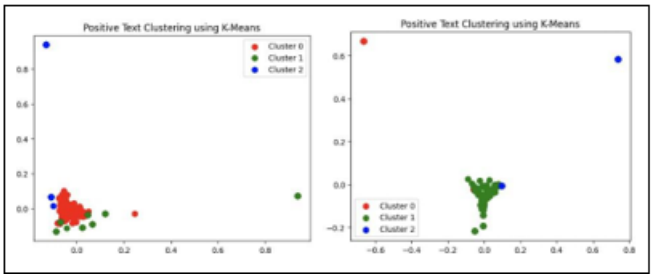

Figure 11: Positive tweet clustering

Random Forest is a machine learning algorithm that takes the use of tree predictors with independent vectors distributed across all the trees. This equal distribution gives a general error that is accounted for, and uses internal estimates to measure for example correlations, connections, accuracy, and more (21).

Naive Bayes came as a machine learning algorithm relatively recently, but the idea behind it comes from a famous formula called “Bayes Theorem” (11) that was actually discovered after the death of Thomas Bayes, who solved one of the most famous unsolved problems from The Doctrine of Chances, a revolutionary textbook on probability theory.

Logistic Regression is a model that observes binary sequences of 0s and 1s, and concludes that the chances of a 1 are determined by the values of independent variables, and then conducts trials with preassigned independent variables, and then with independent variables that are the sequences’ functions (13).

Linear Support Vector Classification-This model uses a nth dimension to determine the optimal hyperplane that differentiates points within differing classes, all while

augmenting the differences between the closest points of different classes (2).

One of the things that all of these machine learning models have in com-mon is that they compress and simplify data to maximize efficiency and reduce other factors such as power input and complexity. That is the main reason for choosing them over other deep-learning models. Because they simplify data and reduce complexity, they are not only extremely easy to use in contrast to most deep learning models, but they are also easy to interpret with the results being in simple terms such as precision, recall, f1-score, and support.

This part of the research paper is to cross apply vader and transformer sentiments with models such as XGBoost and Random Forest, and to find out how they compare. The way we do this is through training a model like Random Forest on a dataset such as Vader, and then test it on the Trans-former dataset, to yield results that we can compare to the original trans-former dataset, which we consider the real one. Taking the predicted and the real, we evaluate the results through metrics such as accuracy, precision, and more, that will be talked about soon.

Vader is a simple algorithm that is at its core a rule-based sentiment analysis algorithm. In Python, it makes use of the *polarity_scores ()* function to generate the polarity scores, which are basically "divisive scores" that revolve around sentiment to generate either a positive, negative, or neutral result. Transformer on the other hand, is a much more complex model that uses encryption and decryption algorithms and the concept of self attention to yield sentiment (17).

The reason we chose these two is because first of all, they are both sub-sets of Natural Language Processing, or NLP algorithms, that are specific to sentiment analysis. In the pre-testing process they also use the same procedures of cleaning, tokenization, and more. Furthermore, they yield the same results in terms of positive, negative, and/or neutral. As a result, they are very similar in the input and the output and the function, with the differences lying in the process. As described before, Vader is a simple algorithm while Transformer is a much more complex one. This difference in complexity and the difference in methods such as Vader using polarity scores while Transformer using encryption and decryption. In order to analyze the similarities and differences between models that have almost identical inputs and outputs but relatively different methods, we chose Vader and Transformer.

Put simply, we made a confusion matrix which is just the visual representation of the intersections between the data with the predicted value being the Vader sentiment and the actual value being the Transformer sentiment. This not only gives us a heatmap of which an example is attached below, but also prints out a classification report, which does all the processing behind the scenes and reveals some of the factors which we'll talk about later, such as precision and recall.

Some of the evaluation metrics we used—such as accuracy, loss, F1 score, and support—are calculated by comparing the predicted data with the actual results. These metrics derive values like true positives, true negatives, false positives, and false negatives, which are summarized in a confusion matrix.

Accuracy simply finds what percent of the predictions were true. In this case, it takes the true positives and negatives and divides it by the total to give us an accuracy score (18).

Precision is basically accuracy but only with the total positives. It takes the number of true positives and divides it by the number of total positives to give a score (18).

Recall finds the percent of positives that were correctly identified as pos-itive. Basically, it divides the number of true positives over the sum of true positives and false negatives, because a false negative is actually a positive (18).

F1 score finds a balance between the extreme precision and recall scores, that tend to sometimes have inaccuracies. Simply put, it multiplies the precision and accuracy scores, and divides this by the sum of the precision and accuracy scores, and multiplies this result by 2. The only thing to know is that if either the precision or the recall are 0 then the numerator in the equation would be 0 and thus the final F1 score would also come out to be 0 (18).

Loss uses binary and multi-class classification to calculate the difference in the predicted probability distribution and the true distribution of the sentiment (1).

After model training, the following results show the accuracy of the mod-els when training them on the Vader dataset. All but Naive Bayes have an accuracy score of over 90 percent. When looking at the classification report for Naive Bayes, the reason becomes apparent. The number 2 (positive) set of tweets has some outlier values in the Naive Bayes report, which pulls down its accuracy. Even some like XGBoost and Support Linear Vectorization have the highest accuracy scores, also have outlying values for the positive tweets, but even then the precision is still higher and so are other scores like f1 score, accounting for the higher accuracy.

1) SVC accuracy score: 0.932933
2) XGBoost accuracy score: 0.927333
3) Random Forest accuracy score: 0.925333
4) Logistic Regression accuracy score: 0.912267
5) Naive Bayes Multinomial Accuracy score: 0.866000

We did cross model testing on all of the models mentioned above: Linear Support Vector Classification, XGBoost, Random Forest, Logistic Regres-sion, and Naive Bayes. There was generally relatively good data, with all accuracy reports being over 90% with the exception of Naive Bayes, which fell to around 87%, with the average being around 0.9120526, or around 91%. One of the trends in these results is that number 2 (positive) tends to be a little less accurate. While certainly not the case in all models such as Random Forest where it had a precision of 0.99, it had a 0.67 precision with Naive Bayes, a 0.89 precision with XGBoost, and 0.87 precision with Linear Sup-port Vector Classification, showing that there is a trend with the transformer vs vader predictions regarding the positive sentiment tweets. Furthermore, looking at the positive sentiment tweets again, the recall was extremely low, no more than 0.61 at max. When looking at how recall works, it divides true positives over the

sum of true positives and false negatives, showing that there were an extremely high number of false negatives, or an extremely low number of true positives, which both reveal some discrepancies between the two models. At the end of the day however, while certainly not perfect data, it is generally well and correlative, while at the same time showing one or two of the discrepancies. 12 showcases the different classification reports for the models, showcasing SVM outperforming other models significantly when it comes to F1 score.

Model	Label	Precision	Recall	F1 score	Support
Random Forest	0	0.95	0.79	0.86	1377
	1	0.92	0.99	0.95	5661
	2	0.99	0.53	0.69	462
Naive Bayes	0	0.79	0.73	0.76	1377
	1	0.9	0.94	0.92	5661
	2	0.67	0.39	0.49	462
XGBoost	0	0.93	0.82	0.87	1377
	1	0.93	0.98	0.95	5661
	2	0.89	0.55	0.68	462
Logistic Regression	0	0.93	0.78	0.85	1377
	1	0.91	0.99	0.95	5661
	2	0.93	0.46	0.62	462
SVM	0	0.92	0.87	0.9	1377
	1	0.94	0.98	0.96	5661
	2	0.87	0.61	0.71	462

Figure 12: Classification reports for the models

6. Conclusion

The purpose of this study was to compare lexicon-based and transformer-based models for sentiment using a dataset of tweets from the Russia Ukraine War for analysis. Using over 30, 000 tweets with the #russiaukrainewar tag, we analyzed metadata including the dates, followers, and likes. Sentiments were generated from both Vader and Transformer, and cross model testing was done with the use of Random Forest, Naive Bayes, XGBoost, Logistic Regression, and SVM. Metrics such as accuracy, recall, f1 score, and more were used to compare and contrast the results of the different models.

The data shows that for the most part, negative and neutral sentiment largely outweighed the positive sentiment when it came to public opinion of the Russia Ukraine war, likely reflecting the distress caused by this geopolitical event, amplified by the scale of the event's discourse. The data also shows that with the war itself, neutral sentiment did start to decrease and negative sentiment did increase as the war went on, demonstrating how each and every day shifted public opinion criticizing the events of the war. The EDA also reveals how those with more followers were most likely to put out neutral sentiment, most likely because of the apolitical stance most people with larger fan bases would want to take because of the risk of losing viewership. When it came to the model specific results, one of the main findings was that transformer based models demonstrated instability, especially when it came to the positive sentiments. This was shown by the lower recall scores consistent throughout all of the cross model testing, sometimes as low as 37%, despite the overall scores shining, with all the models having an overall accuracy of over 90% with the exception of Naive Bayes. The models to perform best were SVC and XGBoost, with accuracies of 93.3% and 92.7% respectively.

This study contributes to sentiment analysis through applying identical analysis to different models, allowing for clean cross evaluation. It shows sentiment analysis under conditions where the process of determining the sentiment stays the same while the raw data remains the same. Furthermore, the study consists of machine learning models like Random Forest and Naive Bayes that offers a balanced framework of interpretability and computational efficiency. This approach is valuable for large-scale use of sentiment analysis, where real-time processing and explainability are often as critical as accuracy.

7. Future Scope

A key limitation of this research is the limited scope of the data, specifically the singular event of the Russia Ukraine conflict, on a single platform X. This narrow focus potentially harms the generalizability to other topics or platforms. In addition, the use of models without specific tuning and optimization also hinders accuracy, mostly when it comes to nuanced and circumstantial situations.

Another limitation lies in the use of pre-trained sentiment models, without fine tuning them to the specific tone and mood of the political discourse. This potentially contributed to observed inconsistencies, especially in classifying positive sentiments. Historically, sentiment analysis has often relied solely on either lexicon based models or deep learning based models without comparing them in controlled setups. Our work breaks this barrier between the two methods by comparing both with a steady evaluation metric; same dataset, same preprocessing, same classifier setup. This comparison shows that the type of model used to label sentiment, whether it's lexicon-based or transformer-based, can significantly affect how well machine learning classifiers perform.

Future studies could build on this approach by incorporating fine tuned and optimized transformers trained on specific discourse about topics such as war or politics. This could also be an opportunity to explore hybrid sentiment systems, which combine the flexibility and interpretability of lexicon methods with the contextual depth and knowledge of transformers based models.

References

[1] GeeksforGeeks. (2024, January 24). *What is Sentiment Anal-ysis?* Retrieved from https: //www.geeksforgeeks. org/ what-is-sentiment-analysis
[2] GeeksforGeeks. (2025, January 27). *Support Vector Machine (SVM) Algorithm*. Retrieved from https: //www.geeksforgeeks. org/ support-vector-machine-algorithm/
[3] GeeksforGeeks. (2024, December 18). *Loss Functions in Deep Learning*. Retrieved from https: //www.geeksforgeeks. org/ loss-functions-in-deep-learning/
[4] Giannoula, M. (2024, July 26). *Transformer-based AI for Sentiment Analysis in Marketing*. Retrieved from https: //dl. acm. org/doi/pdf/ 10.1145/3690771.3690795
[5] Gaillot, A. (2022, December 28). *What Topics Were Trending on Social Media in 2022*. Retrieved from https: //www.meltwater. com/en/blog/ trending-social-media-topics
[6] Hassan, M. (2024, March 25). *Sentiment Analysis – Tools, Tech-niques and Examples*. Retrieved from https: //researchmethod. net/ sentiment-analysis/
[7] Hoti, M., & Ajdari, J. (2023, December 28). *Sentiment Analysis Using the Vader Model for Assessing Company Services Based on Posts on So-cial Media*. Retrieved from https: //sciendo. com/article/10.2478/ seeur-2023-0043
[8] IBM. (2023, August 23). *What is sentiment analysis?* Retrieved from https: //www.ibm. com/think/topics/sentiment-analysis
[9] Dutta, A. (2022). *Twitter Sentiment Analysis, Russia Ukraine Conflict*. Retrieved from https: //www.kaggle. com/code/avinandandutta/ twitter-sentiment-analysis-russia-ukraine-conflict? scriptVersionId=92269032&cellId=11
[10] Openr. (2023, February 24). *Why People Use Social Media To Share Their Opinions*. Retrieved from https: //openr. co/ why-people-use-social-media-to-share-their-opinions
[11] Prince, R. (1763, November 10). *An Essay towards solving a Problem in the Doctrine of Chances*. Retrieved from https: //bayes. wustl. edu/ Manual/an. essay. pdf
[12] Chen, T. (2016, August 13). *XGBoost: A Scalable Tree Boosting System*. Retrieved from https: //arxiv. org/pdf/1603.02754
[13] Cox, D. (1958, November 2). *The Regression Analysis of Binary Se-quences*. Retrieved from https: //www.jstor. org/stable/2983890? read-now=1&seq=1#page_scan_tab_contents
[14] Didenko, M. (2022, March 27). *Fire put out at Lviv oil deport attacked by Russian Missiles*. Retrieved from https: //english. nv. ua/nation/fire-put-out-at-lviv-oil-depot-attacked-by-russian-missiles-50228734. html
[15] The Federal. (2022, December 19). *Top 10 events of 2022 that shaped, shook or brightened the world*. Retrieved from https: //thefederal. com/international/top-10-events-of-2022-that-shaped-shook-or-brightened-the-world
[16] NPR Staff. (2022, April 4). *Russia-Ukraine war: What happened to-day (April 4)*. Retrieved from https: //www.npr. org/2022/04/04/ 1090851859/russia-ukraine-war-what-happened-today-april-4
[17] Suvrat, A. (2025, April 4). *Sentiment Analysis using Python*. Retrieved from https: //www.analyticsvidhya. com/blog/2022/07/sentiment-analysis-using-python/#h-ways-to-perform-sentiment-analysis-in-python
[18] Tigerschiold, T. (2022, November 7). *What is Accuracy, Precision, Recall and F1 Score?*. Retrieved from https: //www.labelf. ai/blog/ what-is-accuracy-precision-recall-and-f1-score
[19] Xhao, X. (2022, December 22). *Automated Measures of Sentiment via Transformer-and Lexicon-Based Sentiment Analysis (TLSA)*. Retrieved from https: //www.researchgate. net/publication/ 366717516_Automated_Measures_of_Sentiment_via_Transformer-_and_Lexicon-Based_Sentiment_Analysis_TLSA
[20] Zadorozhnyy, T. (2025, April 7). *US increasingly polarized over Ukraine support as Trump's 'America First' deepens party divide*. Retrieved from https: //kyivindependent. com/83-of-democrats-back-ukraine-aid-79-of-republicans-against-poll-shows
[21] Breiman, L. (2001, January). *Random Forests*. Retrieved from https: //www.stat. berkeley. edu/~breiman/randomforest2001. pdf

Author Profile

Aditya Arora: Aditya Arora is a student at Summit High School with a strong interest in machine learning, data science, and natural language processing. Outside the classroom he is also an experienced debater, having competed at prestigious national tournaments including Yale, Harvard, Princeton, and Stanford.

Anand Zutshi: Mr. Anand is a Software Development Manager at Amazon, leading engineering teams to build impactful AmazonPay solutions, with prior experience as an SDE, multiple research publications, and a track record of innovation in deep learning and software development.