Social Media Sentiment Analysis of Electric Vehicles:

Leveraging BERT, RoBERTa, Naive Bayes, & VADER

Members: Loo Si Min (Lucy)

Rishika Randev

Eric Ortega Rodriguez

Fan Xu

Group: TMNT

11 Dec 2024

Introduction

Transitioning to sustainable energy and transportation has become a priority in combating climate change, reflecting its growing importance on the American political agenda. As part of this movement, the Biden administration introduced plans to accelerate the adoption of electric vehicles. Specifically, the Build Back Better agenda and the Bipartisan Infrastructure Deal aim to cleaner transportation – with a target of achieving a 50% electric vehicle sales share by 2030.

Public sentiment plays a critical role in the success of such initiatives, as it reflects the views of the American population. Moreover, it influences consumer adoption, market trends, and societal acceptance of electric vehicles. Platforms like Twitter (now known as X) provide an opportunity to analyze public discourse and opinion on these topics.

This study examines shifts in public sentiment on Twitter from January 2020 to December 2023, marked by significant policy announcements and growing discourse around EV adoption. By applying *BERT*, *roBERTa*, Complement Naive Bayes, and VADER models side by side, this study conducts sentiment analysis on tweets related to electric vehicles.

The primary goal of this research is to compare the performance of these four models when applied to the same problem: exploring how public opinion on EVs has evolved in response to political and technological developments between 2020 and 2023. The results of this kind of sentiment analysis have the potential to contribute to a deeper understanding of the interplay between government policies, public opinion, and the transition to sustainable energy and transportation, offering valuable implications for policymakers and industry stakeholders.

Data Collection

Real Data

Twitter was the primary data source for our analysis of public sentiment on EVs. Data was extracted using Octoparse, a versatile online web scraping tool, by doing a keyword search for tweets associated with the phrase "Electric Vehicles" and posted between January 2020 and December 2023. This method enabled us to gather 83,415 tweets, capturing a wide range of public sentiments and opinions regarding EVs over four years.

Synthetic Data

We incorporated synthetic data into our analysis by asking chatGPT to create 400 tweets related to electric vehicles (300 positive, 50 negative, and 50 neutral). These observations were labeled by chatGPT and then used to train our models in order to add an additional layer to our comparison of the models' performances.

Data Processing & Cleaning

Labeling

All 400 of the synthetic tweets were generated and labeled by chatGPT. 300 of the real tweets were randomly sampled from different years of our scraped dataset and manually labeled by a member of the team as having either positive, negative, or neutral sentiments. Prior to the labeling process, a common set of rules was discussed to reduce variability in how the tweets would be labeled. These rules were determined based on the focus of the study, which was to see how the opinions of the common public on EVs have changed over time. Because of this, the following rules were agreed upon in order to ensure that new findings about EVs and tweets from news outlets did not skew the representation of public opinion.

1. Any tweets with positive or negative language within the tweet should be labeled as such, even if it is an article title from a neutral news source.
2. If the tweets have a similar amount of positive and negative language or only neutral language, it should be labeled as neutral.

Text Cleaning

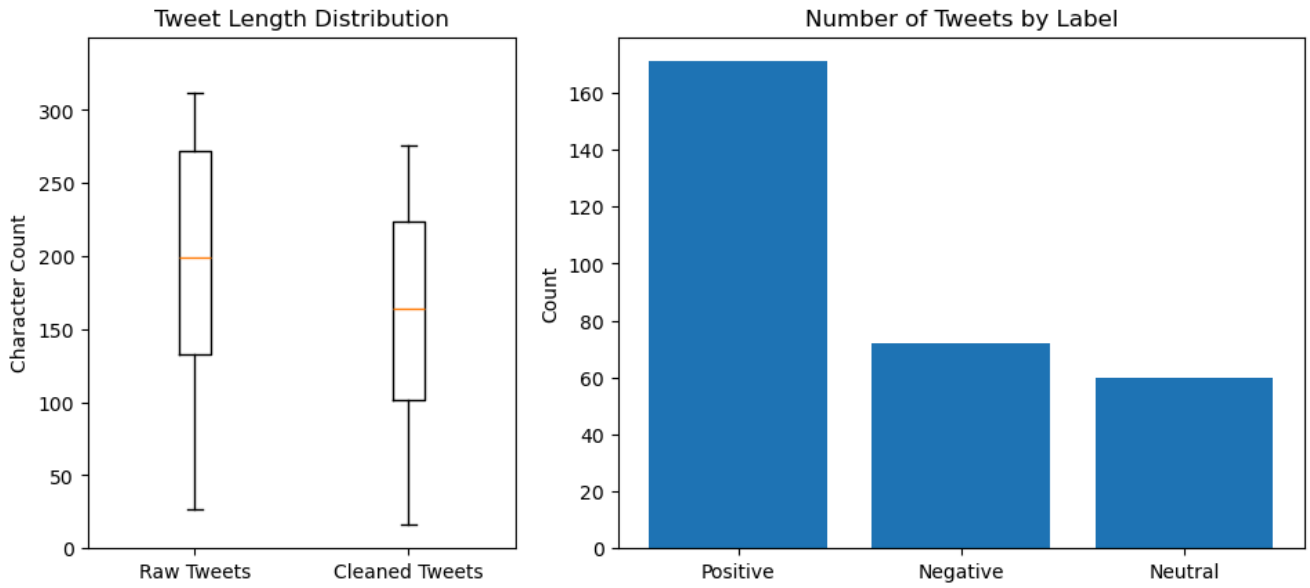
The synthetic dataset contained 400 tweets and 2 columns, tweet content, and label. The combined raw dataset with real tweets scraped using Octoparse from Jan 2020 - Dec 2023 contained 83,415 observations and 24 columns. Key columns of interest in this dataset included the tweet content, language, and tweet date. The raw tweets from both datasets were unsuitable for analysis due to significant noise, such as mentions, blank entries, emojis, and various irrelevant elements. Hence, the datasets were refined to ensure suitability for subsequent modeling and analysis: irrelevant columns were dropped from the real dataset, leaving only the tweet content and date, and preprocessing was performed on both datasets. Entities such as URLs, new line characters, non-ASCII characters, mentions, hashtags, emojis, and special characters (e.g., $ and &) were removed to reduce noise and standardize the text. Tweet normalization was also performed by eliminating extra spaces and converting text to lowercase to ensure consistent formatting across the two datasets. Furthermore, the language of each tweet was verified to ensure that only English-language tweets were included in the analysis.

Tokenization

After processing the text data to a suitable format, the next step was to prepare the text for the models. Tokenization is the process of breaking down text data into smaller units (tokens) such as words, phrases, or subwords, and this is critical for transforming raw text into a format that machine learning models can understand. For the Complement Naive Bayes model, cleaned TF-IDF tokenization was used. For the *BERT* and *roBERTa* models, the cleaned tweets were tokenized using the *BERT* Tokenizer and the *roBERTa* Tokenizer, respectively; each tokenized sequence was padded or truncated to a maximum length of 128.

Exploratory Data Analysis

Within our labeled dataset, our tweets had a mean length of 197.2 characters. This decreased to 161.2 characters after cleaning and removing extraneous content. The median tweet lengths for the raw and cleaned data were 199 and 164, respectively. Due to the means being close to the medians and the boxplots below showing our data to be fairly symmetrical, we can assume our data is relatively unskewed in terms of length.



If we look at our tweets by the distribution of labels, we can see the majority of real tweets on EVs that our team labeled manually are positive in sentiment with 171 total tweets, followed by 72 negative tweets and lastly 60 neutral tweets.

Naive Bayes

Model Overview

When using Complement Naive Bayes (CNB), we classified tweets into three sentiment categories: Positive, Neutral, and Negative. The Complement Naive Bayes algorithm was specifically chosen due to its ability to address skewness in data distribution. In our dataset, the tweets were not equally distributed among the three categories, leading to class imbalance. CNB improves upon traditional Naive Bayes by focusing on the complement of each class during training, thereby mitigating the bias towards majority classes and enhancing performance on the minority categories. This approach was well-suited for sentiment analysis tasks given our imbalanced tweet dataset. Overall, the simplicity, speed, and robustness of Complement Naive Bayes allow for a reliable baseline for classifying tweet sentiments while addressing the challenges posed by an imbalance in classes.

Training Process

To classify the sentiment for electric vehicles, we employed a systematic training process. First, the dataset was split into training and testing subsets, ensuring a balanced distribution of the

three sentiment categories. Next, we transformed the textual data into numerical features suitable for machine learning algorithms. To begin, a count vectorizer converted the text into a sparse matrix of token counts, capturing the frequency of terms in the tweets. After this, a TF-IDF Transformer was applied to the token counts to compute term weights based on their importance across the dataset. The TF-IDF transformation normalized the counts and reduced the influence of commonly occurring terms, thereby enhancing the model's ability to differentiate sentiment. CNB calculates probabilities based on the complement of each class, mitigating bias toward majority classes and improving the classification of minority sentiments. The model was trained using the TF-IDF-transformed training data and corresponding sentiment labels, and predictions were generated on the test data to evaluate the model's performance. This process enabled us to classify tweets about electric vehicles into Positive, Neutral, or Negative sentiments with efficiency and accuracy.

Model Performance

Though we have attempted to address the class imbalance by opting for complement Naive Bayes, this class imbalance still influenced the model's performance, as the recall for the negative and neutral classes was very low, at 0.19 and 0.07 respectively, while the positive class achieved a much higher recall of 0.97. These results suggest the model struggles to identify negative and neutral examples accurately, which is understandable given the limited number of negative samples and the inherent difficulty in clearly identifying sentiments when manually labeling the neutral class. The model achieved an accuracy of 0.56, with precision being highest for the negative class (0.75), indicating fewer false positives, and the F1-score being highest for the positive class (0.70), making it the most reliably classified category.

Class	Precision	Recall	F1-Score
Class 0 (Negative)	0.75	0.19	0.30
Class 1 (Neutral)	0.50	0.07	0.12
Class 2 (Positive)	0.55	0.97	0.70
Macro Average	0.60	0.41	0.37
Weighted Average	0.59	0.56	0.46
Overall Accuracy	0.56		

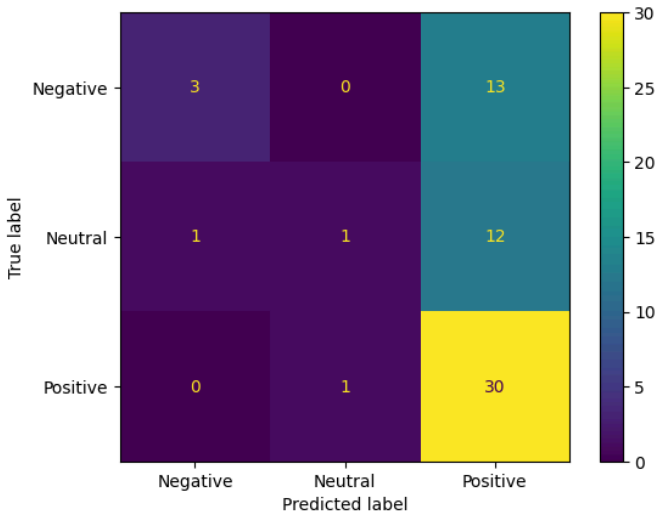


Figure 1: Confusion matrix for Complement Naive Bayes

Naive Bayes Training With Synthetic Data

Data & Model Overview

400 tweets were generated and labeled by ChatGPT. The distribution of these synthetic tweets differed significantly from what we observed in the real data, with 300 tweets labeled as positive, 50 as neutral, and 50 as negative. The Complement Naive Bayes model, along with the training process and hyperparameters, was consistent with those used during real-world training.

Complement Naive Bayes Model Performance

Overall, the model performed better in most regards when trained and tested on the synthetic data than on the real data. However, this is probably due to the increase in sample size, as well as less nuanced tweet sentiments generated by AI. While synthetic data helped boost accuracy and Positive sentiment classification, it lacked the nuanced variability of real-world data, limiting its effectiveness in training a balanced model.

The use of synthetic data generated by ChatGPT significantly influenced the performance of the Complement Naive Bayes model. The dataset was highly imbalanced, favoring Positive tweets (300 Positive, 50 Neutral, and 50 Negative). This skewness caused the model to perform exceptionally well on Positive tweets and achieved a F1-score of 90% and a recall of 96%. The overall accuracy also improved from 56% with real-world data to 76% with synthetic data. On the other hand, the model's performance on Neutral and Negative tweets remained poor, with F1-scores was low and remained at 32%. Examples of mislabeled tweets are shown in Table 1, and indicate the model may not have been able to interpret the influence of succeeding context on the sentiment of phrases like “longer drives,” and the relative importance of surrounding context (for example, “reduce” and “dependency” together contribute the most to making the third example positive). Context around certain words such as “fossil fuels” could have also been misinterpreted.

Class	Precision	Recall	F1-Score
Class 0 (Negative)	*0.5*	*0.23*	*0.32*
Class 1 (Neutral)	*0.33*	*0.3*	*0.32*
Class 2 (Positive)	*0.85*	*0.96*	*0.9*
Macro Average	*0.56*	*0.5*	*0.51*
Weighted Average	*0.73*	*0.76*	*0.73*
Overall Accuracy	*0.76*		

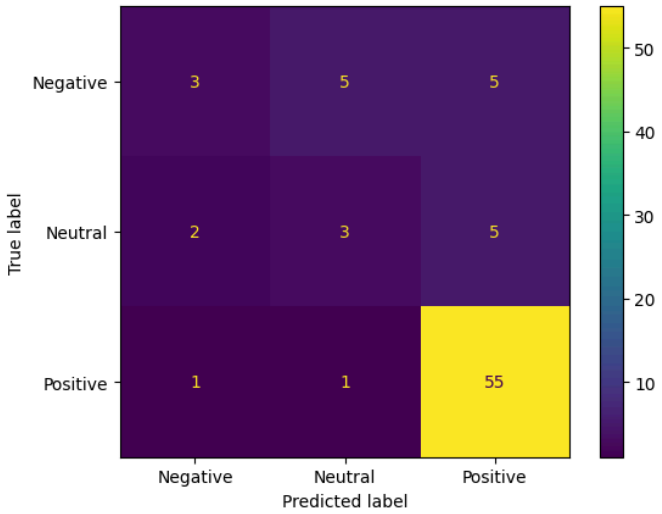


Figure 2: Confusion matrix for Complement Naive Bayes trained and tested on synthetic data

Cleaned Tweet	True Label	Predicted Label
rural ev owners often face longer drives to find a public charging station accessgap	Negative	Neutral
evs are costeffective even in rural areas thanks to fewer fuelrelated expenses broadsavings	Positive	Negative
ev adoption helps reduce dependency on fossil fuels energyindependence	Positive	Negative

Table 1: Examples of synthetic tweets mislabeled by CNB

BERT & RoBERTa

Bidirectional Encoder Representations from Transformers (BERT)

BERT is a pre-trained language model introduced by Devlin et al. (2019) that utilizes a transformer-based architecture to process text bi-directionally. Therefore, unlike traditional unidirectional models, *BERT* considers both the preceding and succeeding context of words in a sentence, allowing it to capture the context and semantic information more effectively. The bidirectional transformer architecture is achieved with the self-attention mechanism (Vaswani et al, 2017). It is pre-trained on large corpora i.e. English Wikipedia.

BERT is pre-trained with two key tasks: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP). MLM is a pre-training task where a percentage (typically 15%) of input tokens in a text sequence are randomly "masked" (replaced with a special [MASK] token) then predicted based on the surrounding context, which trains the model's understanding of relationships between words. In NSP, *BERT* learns how to capture context across sentence pairs which is relevant for tasks like question answering.

Robustly Optimized BERT Pre-Training Approach (roBERTa)

RoBERTa builds upon *BERT* by optimizing its pretraining strategy (Liu et al, 2019). *RoBERTa* retains the same fundamental transformer-based architecture as *BERT*, but it introduces several improvements that enhance its performance.

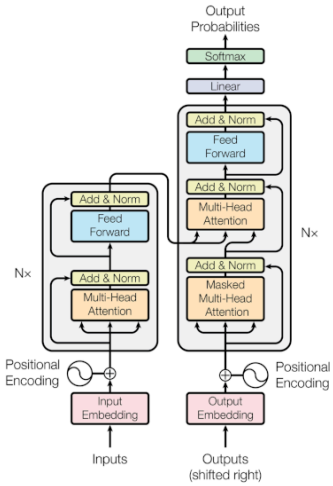


Figure 3: BERT & roBERTa Architecture

Firstly, it removes the NSP task as it was found to contribute minimally to downstream performance. Secondly, *roBERTa* is also trained on a significantly wider and larger dataset. Thirdly, *roBERTa* introduces dynamic masking in the context of the MLM task which addresses a limitation in *BERT*'s static masking approach. In *BERT*, the masked tokens are determined only once during data preprocessing. This means that for every epoch (pass over the training dataset), the model sees the same tokens masked in the same positions. Meanwhile, tokens

are masked on the fly during each training epoch in *roBERTa*. Each time a sequence is passed through the model during training, a new random subset of tokens is masked. Over multiple epochs, the model encounters different combinations of masked tokens, leading to richer training examples.

In implementing *BERT* and *roBERTa* for the task of sentiment analysis, we fine-tuned *BERTforSequenceClassifcation* and *RobertaforSequenceClassifcation* models for our context.These models contain a classification head on top of the base models that outputs logits, and are pre-trained for classification tasks.

Training Process

Train-Test Split

The labeled portion of the real-world dataset consisted of 300 tweets, and this was then split 80:20 into 240 tweets for training and 60 tweets for testing. Sentiment classes were mapped numerically: *Negative = 0, Neutral = 1, and Positive = 2.*

Class Balancing

Since the real-world data was overrepresented with positive sentiment labels as compared to negative and neutral sentiments, class weights were calculated and added during training; this was done to ensure that a misclassification of the minority classes would be penalized more harshly.

Class	Count in Training Data
Negative	72
Neutral	60
Positive	171

One Hot Encoding

The sentiment labels were converted into a one-hot encoded format to represent each sentiment class as a binary vector (e.g., [1, 0, 0] for neutral, [0, 1, 0] for positive, [0, 0, 1] for negative). This format is required for the CategoricalCrossentropy loss function we opted to use during training.

Fine Tuning for Sentiment Analysis

Firstly, pre-trained tokenizers (BertTokenizerFast & RobertaTokenizerFast) were used to convert the training and testing tweets into token IDs and attention masks. After tokenization, the pre-trained *BERTforSequenceClassifcation* and *RobertaforSequenceClassifcation* models were loaded to provide contextual embeddings. Both models were then trained to classify sentiments into the respective categories - neutral, positive, and negative. In terms of hyperparameters,

batch size was set to 16 and the learning rate was set to 5e-5, based on recommendations from the original *BERT* paper (Devlin et al, 2019).

Model Performance

In the first round, both *BERT* and *RoBERTa* performed relatively similarly on the testing dataset. Class 2 had the best performance metrics overall, which makes sense given that a majority of the training and testing set belonged to class 2 (positive sentiment). Given the class imbalance, it is especially important to look not only at overall accuracy but at F-1 scores, which indicate the balance between precision (when the model labels a tweet with a particular sentiment, how often is it actually correct) and recall (how often does the model accurately identify a particular sentiment). Both models performed quite poorly on class 1 (neutral sentiment), but a comparison of precision, recall, and F-1 scores between the models indicated that *roBERTa* was able to identify class 0 (negative sentiment) much better than *BERT*. Additionally, there was a tendency across both models to misclassify neutral sentiment as positive and vice versa, as seen in both confusion matrices. A few tweets that were mislabeled in these ways by roBERTa are shown in Table 1 below; these examples suggest that the models may not have been distinguishing positive opinions about EVs ("key to beating climate change") from general news about EV growth ("sales..in Europe overtake") in the same way that the manual labelers did.

There is clearly significant room for improvement in the models, especially when it comes to identifying neutral sentiment, which makes sense given the fact that the neutral sentiment class was the least represented in both the training and testing data.

BERT

	Precision	**Recall**	**F-1 Score**
Class 0 (Negative)	0.53	0.60	0.56
Class 1 (Neutral)	0.33	0.33	0.33
Class 2 (Positive)	0.79	0.74	0.77
Macro Average	0.55	0.56	0.55
Weighted Average	0.62	0.61	0.61
Overall Accuracy	0.61		

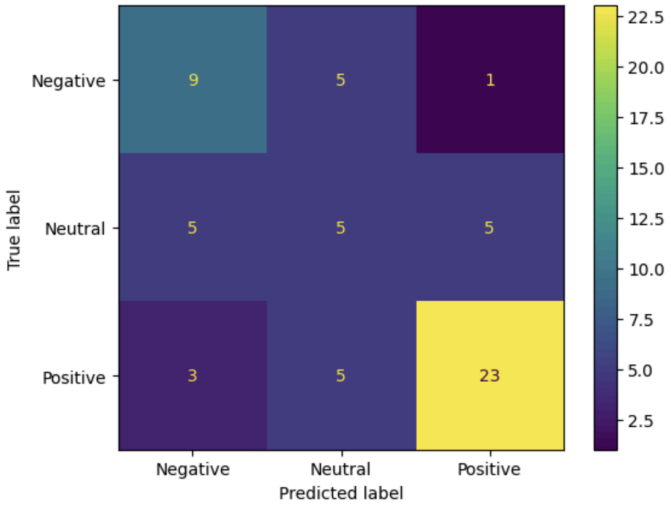


Figure 4: Confusion matrix for BERT

RoBERTa

	Precision	**Recall**	**F-1 Score**
Class 0 (Negative)	0.80	0.80	0.80
Class 1 (Neutral)	0.20	0.18	0.19
Class 2 (Positive)	0.78	0.80	0.79
Macro Average	0.59	0.59	0.59
Weighted Average	0.68	0.69	0.68
Overall Accuracy	0.69		

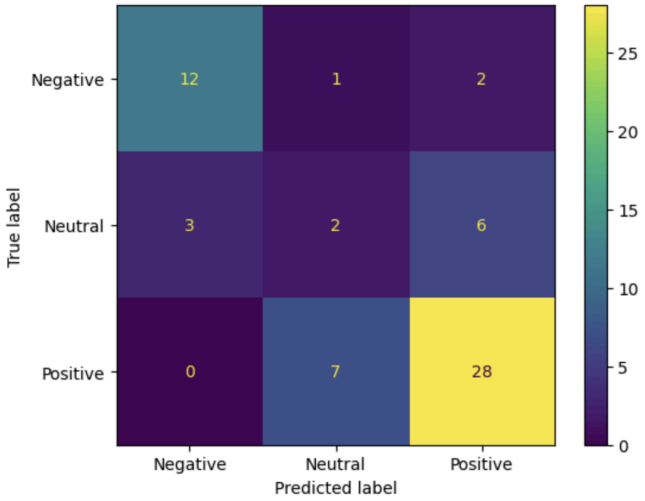


Figure 5: Confusion matrix for roBERTa

Cleaned Tweet	True Label	Predicted Label
clean energy grids and electric vehicles key to beating climate change and air pollution	Positive	Neutral
sales of electric cars in europe overtake those in china	Neutral	Positive
the transition to zero emission vehicles over the next decade will see a dramatic change in the skills required with apprentices and existing technicians needing the training experience and infrastructure to service and repair electric vehicles	Neutral	Positive

Table 2: Example tweets mislabeled by roBERTa

BERT & RoBERTa Training With Synthetic Data

Data & Model Overview

BERT and *roBERTa* were fine-tuned on the 400 synthetic tweets generated by ChatGPT as well. The pre-trained models, training process, and hyperparameters were the same as those used in real-world training.

BERT & RoBERTa Model Performance

Overall, both *BERT* and *roBERTa* performed better in most regards when trained and tested on the synthetic data than on the real data. *BERT* specifically had high precision, recall, F-1 scores, and accuracy across negative and positive classes, and high precision on the neutral class. roBERTa did not perform as well on the negative and neutral classes. In general, however, because the testing data size was quite small (only 80 samples) and positive sentiment was overrepresented (making up around 60 of those), it is likely that the models were able to get

away with overfitting on the positive class. Examples of neutral tweets that were mislabeled as positive are shown in table 3 below.

BERT

	Precision	**Recall**	**F-1 Score**
Class 0 (Negative)	0.80	0.80	0.80
Class 1 (Neutral)	0.82	0.69	0.75
Class 2 (Positive)	0.95	0.98	0.97
Macro Average	0.86	0.82	0.84
Weighted Average	0.91	0.91	0.91
Overall Accuracy	0.91		

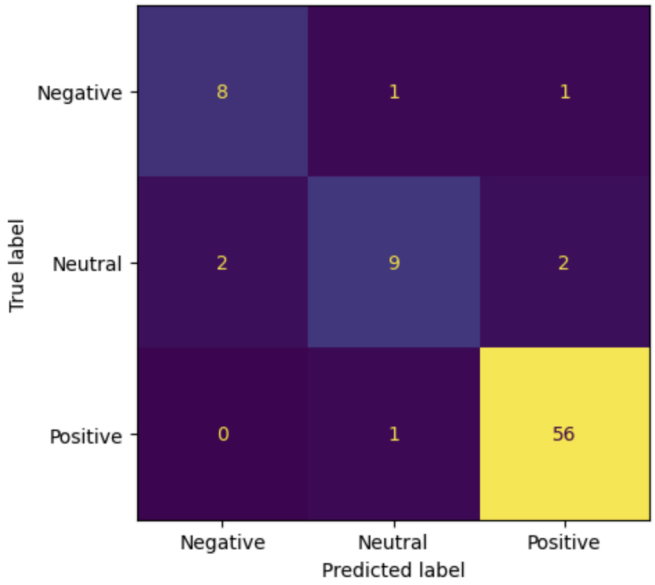


Figure 6: Confusion matrix for BERT with synthetic testing data

RoBERTa

	Precision	Recall	F-1 Score
Class 0 (Negative)	0.62	0.56	0.59
Class 1 (Neutral)	0.62	0.45	0.53
Class 2 (Positive)	0.94	1.00	0.97
Macro Average	0.73	0.67	0.69
Weighted Average	0.86	0.88	0.86
Overall Accuracy	0.88		

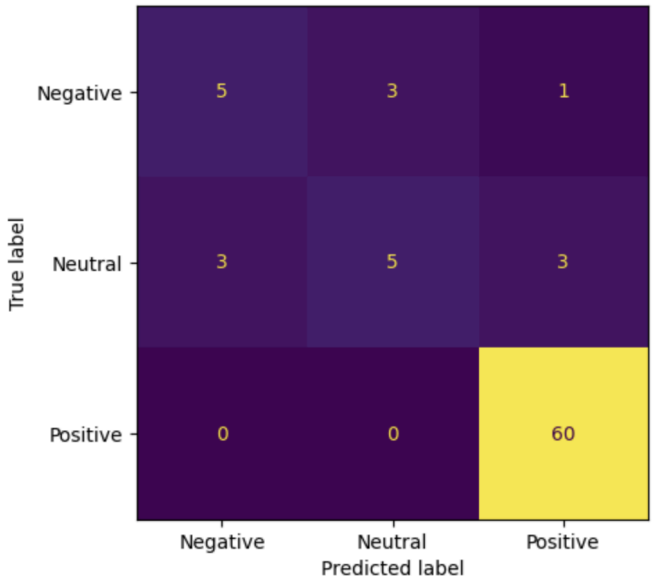


Figure 7: Confusion matrix for roBERTa with synthetic testing data

Cleaned Tweet	True Label	Predicted Label
higher insurance costs for evs can offset savings on fuel and maintenance hiddencosts	Neutral	Positive
ev range estimates are just thatestimates realworld range depends on driving habits realitycheck	Neutral	Positive

Table 3: Examples of synthetic tweets mislabeled by roBERTa

Vader

Description of Model

VADER (Valence Aware Dictionary and sEntiment Reasoner) is a lexicon and rule-based sentiment analysis tool created in 2014 that was specifically designed to analyze sentiment in social media. It is particularly effective for tasks involving texts, emojis, comments, and informal language. VADER has been shown to perform as well as or even better than human raters at classifying the sentiment of tweets, which is why we chose it as a point of comparison for our other models. VADER classifies text into 3 categories, positive, neutral, and negative by calculating 3 scores that indicate the proportion of the text that falls into each category. It is not only sensitive to the polarity of sentiments but also their intensity which VADER computes as a compound score, derived from the lexicon valence score sum, adjusted according to the rules, and then normalized.

The compound score will be the main metric we use to evaluate sentiment since it takes into account sentiment intensity. The score ranges from -1 being the most negative to 1 being the most positive. The typical thresholds set for classifying text are positive for scores greater than or equal to 0.5, negative for scores less than or equal to -0.5, and neutral for scores between -0.5 and 0.5.

Model Limitations

Given that VADER is a pre-trained model, its main purpose is to serve as a baseline for comparison with more advanced models like *RoBERTa* and *BERT* in sentiment analysis. While VADER has the capability to handle unusual characters and patterns within social media text, the model architecture is less complex and extensive when compared to *RoBERTa* and *BERT*. This is because models such as the latter leverage deep learning and transformer architectures which allow a better incorporation of certain nuances, especially given the niche and complex topic – electric vehicle sentiment. VADER on the other hand simply evaluates scores for words that are contained within its lexicon that were generated by human raters and then calculates sentiment based on simple rules from language modifiers and qualifiers. Because VADER is lexicon-based and contains no smoothing, it can only analyze text that contains words within its lexicon. If the text does not contain any words that VADER can recognize, it will simply output a compound score of 0.0, classifying the text as neutral even though it may not be. Due to these reasons, VADER is much more lightweight and efficient than ML models, but also less flexible.

Model Performance

In comparison to *BERT* and *RoBERTa*, VADER performed considerably worse on real tweets. Its overall accuracy was 0.11 points lower than *BERT* and 0.16 points lower than *RoBERTa*. In terms of precision VADER's scores were lower across the board, and it especially struggled to predict positive sentiment, often conflating it with neutral sentiment. This overpredicting of neutral tweets could be due to words not being present in the lexicon as explained in the model limitations. The category VADER was the worst at was negative sentiment, only predicting 4 out of 20 negative tweets correctly. Normally this could be attributed to the lack of negative tweets in the training data, but because VADER was not trained on that data, underfitting does not seem

like a plausible explanation. Overall, our fitted models were able to classify the tweets with their training much more accurately than the pre-trained VADER model, indicating that specializing in the EV topic was important.

VADER

	Precision	**Recall**	**F-1 Score**
Class 0 (Negative)	0.67	0.20	0.31
Class 1 (Neutral)	0.55	0.70	0.62
Class 2 (Positive)	0.40	0.40	0.40
Macro Average	0.54	0.43	0.44
Weighted Average	0.53	0.51	0.49
Overall Accuracy	0.51		

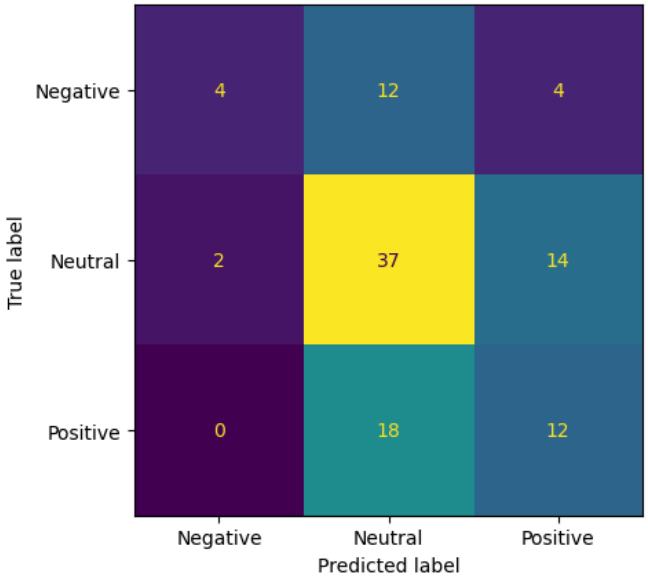
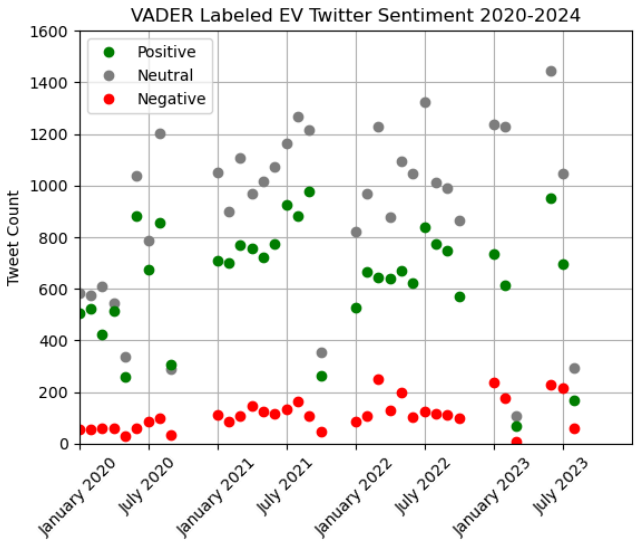
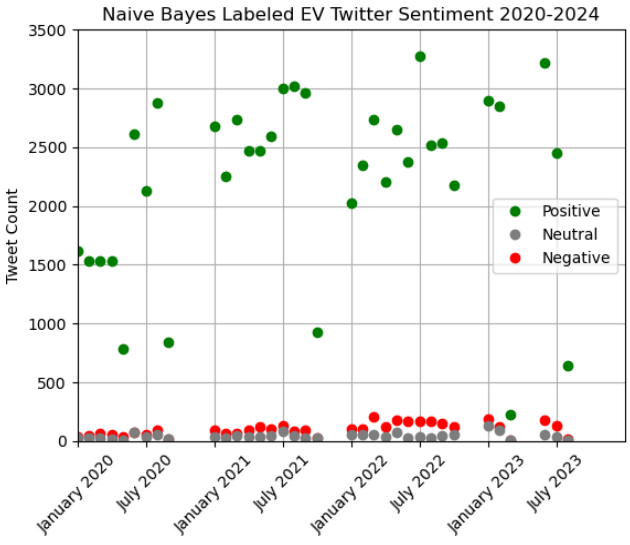


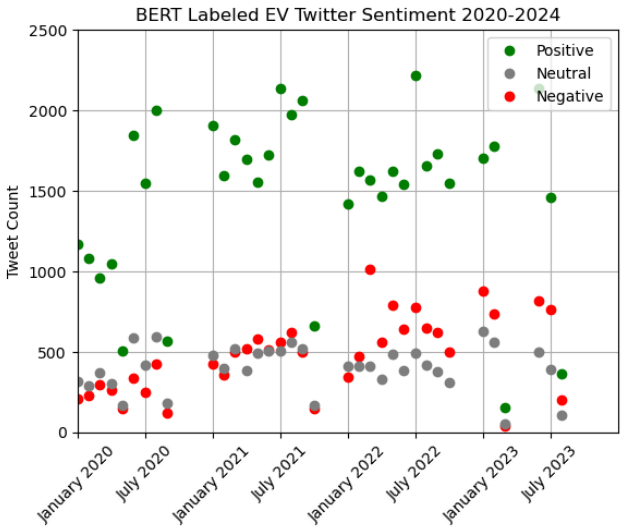
Figure 8: Confusion matrix for VADER

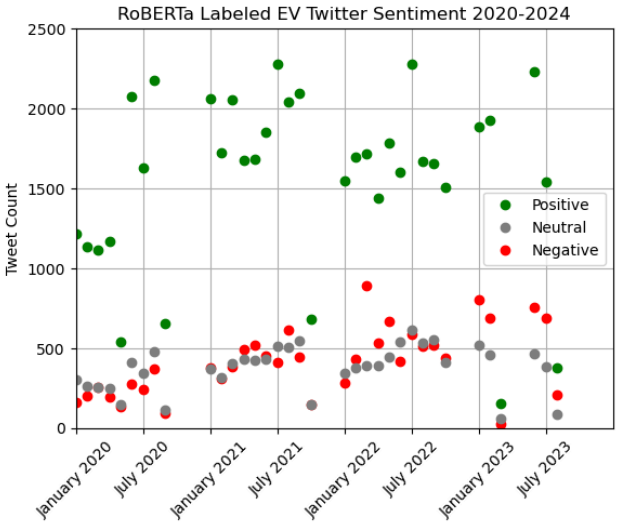
Sentiment Analysis Over Time





In order to add an extra dimension to our analysis, we decided to see if the sentiment on EVs has changed over time. We looked at our full data unlabeled data in the time period beginning from January 2020 until December 2023 using each of our models. VADER is the only model that predicts most of the tweets to be neutral while the Naive Bayes model has the largest proportion of positive tweets by far, with barely any neutral or negative tweets. These seem in line with the accuracy issues that we've seen with each model.





BERT and *RoBERTa* have similar predictions, with the ratios being much more similar to our labeled tweet proportions. Overall we see a fairly constant sentiment across this period with some occasional blips, which could be explained by some data collection issues.

Conclusion

Overall, there are several limitations to this study which suggest a potential scope for improvement in future iterations. The most significant limitation is that due to time and resource constraints, only a small subset of collected real tweets were able to be labeled, meaning our training and testing datasets were limited to 300 tweets altogether. Ideally, increasing the size of this dataset would substantially improve model performance, especially given the fact that neutral and negative tweets were poorly represented in our dataset as a whole. Additionally, each tweet was only labeled by one person, which could lead to bias and inconsistencies in how the confines of the three different classes were approached by each labeler, despite the fact that rules were established prior to the labeling process. It would be more optimal to have multiple trained labelers label each tweet with a score and then aggregate the scores by averaging. It is also possible that the models had trouble “learning” the rule for marking tweets containing news and findings about EVs as positive instead of neutral, and vice versa. The fact that CNB, *BERT*, and roBERTa all performed much better overall on the synthetic data than on real data supports this conclusion, as that data was labeled more consistently and by adhering more rigorously to a ruleset. Therefore, further defining the manual labeling rules and process in order to ensure that training examples accurately reflect labels is pivotal. Only then can we expect the models to appropriately distinguish the relationships between news with a positive slant and a general presentation of findings.

Given that *BERT* and *roBERTa* are not highly interpretable models, a comparison with the simpler Complement Naive Bayes model is warranted. CNB demonstrated effectiveness as a baseline for sentiment classification, particularly when class imbalance was an issue. However, its performance was slightly worse than *BERT* and *RoBERTa* due to challenges in capturing the nuances of the dataset, especially in cases involving sarcasm or ambiguous sentiment. Future efforts should focus on expanding our dataset, improving consistency when labeling, and even exploring hybrid approaches that combine the interpretability and speed of classical models like CNB with the adaptability of transformer-based models.

References

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805*. Retrieved from https://arxiv.org/abs/1810.04805

Hutto, C.J. & Gilbert, E.E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. *Eighth International Conference on Weblogs and Social Media (ICWSM-14)*. Ann Arbor, MI, June 2014. Retrieved from https://ojs.aaai.org/index.php/ICWSM/article/view/14550/14399

Kamaldeep. (2024, October 20). How to Improve Class Imbalance using Class Weights in Machine Learning? Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/10/improve-class-imbalance-class-weights/#:~:text=Class%20weights%20are%20a%20technique,bias%20towards%20the%20majority%20class.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*. Retrieved from https://arxiv.org/abs/1907.11692

Neptune AI. (n.d.). *BERT and the transformer architecture*. Retrieved November 26, 2024, from https://neptune.ai/blog/bert-and-the-transformer-architecture

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is All You Need. *Advances in Neural Information Processing Systems, 30*. Retrieved from https://arxiv.org/abs/1706.03762