ISSN: 2994-6433

Research Article

Journal of Sensor Networks and Data Communications

Unraveling Emotions: Contemporary Approaches in Sentiment Analysis

Eberechukwu Q. Chinedu[1], Emmanuel Chibuogu Asogwa[2], Belonwu Tochukwu Sunday[3], Ndukaku Macdonald Onyeizu[4] and Okechukwu J. Obulezi[5*]

[1,5]*Department of Statistics, Faculty of Physical Sciences, Nnamdi Azikiwe University, Awka, Nigeria*

[2,3,4]*Department of Computer Science, Faculty of Physical Sciences, Nnamdi Azikiwe University, Awka, Nigeria*

[*]Corresponding Author:
Okechukwu J. Obulezi, Department of Statistics, Faculty of Physical Sciences, Nnamdi Azikiwe University, Awka, Nigeria.

Submitted: 2023, Nov 19; **Accepted**: 2023, Dec 26; **Published**: 2023, Dec 29

Citation: Chinedu, E. Q., Asogwa, E. C., Sunday, B. T., Onyeizu, N. M., Obulezi, J. O. (2023). Unraveling Emotions: Contemporary Approaches in Sentiment Analysis. *J Sen Net Data Comm, 3*(1), 223-230.

Abstract

Sentiment analysis, an essential component of natural language processing, plays a pivotal role in deciphering public opinion and emotional cues within the vast sea of user-generated content on social media platforms. This paper presents a focused analysis on sentiment analysis leveraging two different techniques, namely VADER (Bag of Words approach) and the RoBERTa model, an extension of the BERT architecture, known for its outstanding performance in a wide range of Natural Language Processing (NLP) tasks. Model A achieved a 47% accuracy in sentiment classification, while Model B demonstrated a higher accuracy of 73%. The findings not only highlight the performance disparities between the two models but also offer insights into the factors contributing to their varying degrees of accuracy. This research underscores the significance of model selection in sentiment analysis tasks and contributes to a better understanding of their applicability in real-world scenarios.

Keywords: Sentiment Analysis, Natural Language Processing (NLP), NLP Models, Text Mining, Sentiment Classification, Computational Linguistics, Text Sentiment Prediction.

1. Introduction

Sentiment analysis, also known as opinion mining, is a flourishing field in natural language processing and data analytics. In an era where digital communication and user-generated content have become pervasive, the ability to automatically gauge and interpret sentiments expressed in text has profound implications for various sectors, including business, marketing, politics, and customer service. Sentiment analysis, at its core, is the process of determining the emotional tone or attitude conveyed within textual data, be it social media posts, customer reviews, news articles, or any form of written communication. This technology allows for the automated classification of text as positive, negative, or neutral, and even more nuanced sentiments such as happiness, anger, or sadness, enabling organizations and individuals to make data-driven decisions, improve customer satisfaction, manage their brand reputation, and gain insights into public opinion.

In their seminal work, Pang and Lee laid the groundwork by introducing a groundbreaking lexicon-based method known as the Vader model, which efficiently captures sentiment polarity in diverse texts [1]. Building upon this foundation, the study by Liu et al. delves into the realm of deep learning, presenting the robustness of the Roberta model in understanding intricate nuances and context within sentiment-laden content as well as the effectiveness/importance of large-scale pretraining for sentiment analysis across diverse domains [2].

Recent advancements in sentiment analysis have been marked by the emergence of transformer-based models. Devlin et al. introduced BERT (Bidirectional Encoder Representations from Transformers), a contextualized language model that achieved state-of-the-art performance on various NLP tasks, including sentiment analysis and by Wang et al. who extended the discourse by introducing a hybrid sentiment analysis framework, amalgamating machine learning and lexicon-based strategies for enhanced accuracy [3,4].

In 2020, Adoma et al. compared RoBERTa model with 3 other models to determine which model performed best on text-based emotion recognition [5]. They concluded that the RoBERTa model outperforms the other three models not only in specific emotions but in a wide range of emotions ranging from anger to fear, sadness to joy, love to hate, etc. In 2023, Yani et all were able to use the RoBERTa model to detect cyber-bullying content on Twitter with an accuracy of 89.6% [6]. [7] in their review paper, summarized fifty-four articles on Sentiment Analysis. The articles were categorized according to their contributions to the various Sentiment Analysis (SA) techniques. They are divided into six categories namely Sentiment Analysis, Emotion Detection, Sentiment Classification, Feature Selection, Transfer Learning, and Building Resources. [8] in their comparative study on different approaches to sentiment analysis identified machine learning-based approaches as Support Vector Machine, N-gram Sentiment Analysis, Naïve Bayes Method, Maximum Entropy Classifier, K-NN and Weighted K-NN, Multilingual Sentiment Analysis, Feature Driven Sentiment Analysis other approaches include rule-and lexical-based approaches. [9] took their perspective of sentiment analysis from the linguistic overview. [10] explored sentiment analysis in the news. According to the article, the main difference texts have with news articles is that their target is clearly defined and unique across the text. Following different annotation efforts and the analysis of the issues encountered, they concluded that news opinion mining is not the same as the other text types. Three sub-tasks were identified; definition of the target; separation of the good and bad news content from the good and bad sentiment expressed on the target; and analysis of clearly marked opinion that is expressed explicitly, not needing interpretation or the use of world knowledge. In this same development, three different possible views on newspaper articles were distinguished namely author, reader, and text, which have to be addressed differently at the time of analyzing sentiment. Given these definitions, work on mining opinions about entities in English language news was deployed, in which (i) a test on the relative suitability of various sentiment dictionaries was carried out and (ii) an attempt to separate positive or negative opinions from good or bad news. It was tested whether or not subject domain-defining vocabulary should be ignored. Results showed that this idea is more appropriate in the context of news opinion mining and that the approaches considering this produce a better performance. Other significant contributions include [11-15].

While the field has witnessed remarkable progress, challenges persist in effectively addressing sentiment variations across different domains and languages, as highlighted in the work of Gupta and Dasgupta [16]. The ongoing exploration of transformer architectures, as seen in models like GPT-3, signals a continued evolution in sentiment analysis methodologies [17]. The review of related works seeks to contextualize our work within this vibrant landscape, contributing insights that advance our understanding of sentiment analysis methodologies. By juxtaposing the Vader and Roberta models, our study adds a nuanced perspective to the ongoing discourse, striving to guide future research toward more effective sentiment analysis solutions.

The organization of the remainder of this article appears in this sequence. Section 2 considers the methodology. In section 3, we dwell on the analysis and results. The article is concluded in section 4 with the discussion of results and conclusion.

2. Methodology

The dataset utilized in this study was sourced from Kaggle, a popular platform for sharing datasets and machine learning resources. The dataset consists of 10 columns and approximately 30000 observations. The columns are:

- textID
- text
- selected text
- sentiment
- Time of Tweet
- Age of User
- Country
- Population (as at 2020)
- Land Area (in square kilometers)
- Density (Population per square kilometer)

The dataset provides valuable information regarding sentiments, tweet text, user details, and geographical statistics, facilitating analysis and research within these domains. However only two columns are relevant to this study. Those columns are 'text' and 'sentiment'.

In this study, python was used to run both the VADER analysis and the RoBERTa analysis.

2.1 Steps Involved in Natural Language Processing (NLP)

Natural Language Processing (NLP) encompasses several vital steps, with significant emphasis placed on the text processing phase. The initial text preprocessing stage involves several key procedures aimed at refining raw textual data for effective analysis. This phase typically consists of 4 - 7 steps namely:
1. Tokenization: Here, strings (text) are broken into tokens (words).
2. Stemming/Lemmatization: In this step, each word is converted to its root form. Eg: running to run, going/went/gone to go, etc.
3. Stop Words: Stop words are common words that are frequently used in natural language but often do not carry significant meaning in the context of a specific text. This includes articles (e.g., "the," "a," "an"), conjunctions (e.g., "and," "but," "or"), prepositions (e.g., "on," "in," "at"). Removing stopwords, is crucial for streamlining the data.
4. Parts of Speech (POS): This is the task of labelling the words in the text according to their part of speech.
5. Names Entity Recognition (NER): These are noun phrases that refer to specific locations, people, facility, etc. Eg: World Health Organisation (WHO), Nnamdi Azikiwe University, His Excellency Peter Obi, etc.
6. Chunking: Here, individual pieces of information are picked up and grouped into larger pieces.

The models carry out the aforementioned steps themselves. This description is provided to offer a level of insight into the workings of these models.

2.2 Metrics

The metrics used for the evaluation of the models will be discussed in this section.

To determine how well a model performs, classification metrics such as precision, recall, f1 score, accuracy score amongst others are used. For this analysis, the aforementioned metrics will be used to evaluate the performance of the models.

1. Precision: This measures the accuracy of positive predictions. It is the ratio of correctly predicted positive observations to the total predicted positive observations.

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{1}$$

2. Recall (Sensitivity): This measures the ability of the model to identify all relevant instances. It is the ratio of correctly predicted positive observations to the actual positive observations in the data.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{2}$$

3. F1 Score: The F1 score considers both precision and recall and is calculated using as

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3}$$

4. Accuracy Score: This measures the ratio of correctly predicted instances to the total instances in the dataset. The accuracy score is calculated using the following formula:

$$Accuracy = \frac{\text{True Positives} + \text{True Negatives}}{\text{True Positives} + \text{False Positives} + \text{True Negatives} + \text{False Negatives}} \tag{4}$$

2.3 VADER Technique (Bag of Words Approach)

VADER stands for Valence Aware Dictionary and sEntiment Reasoner. It is a lexicon and simple rule-based model for sentiment analysis. VADER has the advantage of assessing the sentiment of any given text without the need for previous training as we might have to do for Machine Learning models.

The result generated by VADER returns 4 columns. Negative, neutral, positive and compound. Their sum of the first 3 should be equal to 1 or close to it with float operation. While compound corresponds to the sum of the valence score of each word in the lexicon and determines the degree of the sentiment rather than the actual value as opposed to the previous ones. Its value is between -1 (most extreme negative sentiment) and +1 (most extreme positive sentiment). Using the compound score can be enough to determine the underlying sentiment of a text, because for:

• A positive sentiment: compound ≥ 0.05
• A negative sentiment: compound ≤ -0.05
• A neutral sentiment, the compound is between (-0.05, 0.05)
Now that we understand the main concepts, let's dive into the implementation.

2.4 RoBERTa Technique

In the context of the transformer architecture used in models like RoBERTa, there are a few key mathematical formulations involved. Here are some fundamental equations:
1. Self-Attention Mechanism: The self-attention mechanism is a critical component in transformers. Given a sequence of words represented as vectors, it computes the weighted sum of these vectors to capture the relationships between words in the sequence. The equations for self-attention are as follows:-

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{QK}^T}{\sqrt{d_k}}\right)\mathbf{V} \tag{5}$$

• Q, K, and V are the Query, Key, and Value matrices, respectively.
• d_k represents the dimension of the key.
• Softmax is used to normalize the attention scores, providing a weight for each word in relation to others.

2. Positional Encoding: In transformer models, positional encoding is added to the input embeddings to provide information about the position/order of words in the sequence. The formula for positional encoding can be represented as:

$$\mathrm{PE}(pos, 2i) = \sin\left(\frac{pos}{10000^{2i/d_{\mathrm{model}}}}\right) \quad (6)$$

$$\mathrm{PE}(pos, 2i+1) = \cos\left(\frac{pos}{10000^{2i/d_{\mathrm{model}}}}\right) \quad (7)$$

These formulas play a crucial role in the functioning of transformer-based models like RoBERTa. They enable the model to capture the relationships between words and encode positional information, facilitating better language understanding and sequence processing.

3. Analysis and Results

In this section, certain stages of the analysis process will be explored as well as the results obtained at the end of the analysis.

3.1 Data Cleaning

During this phase, the data was loaded into a Pandas dataframe, followed by the removal of undesirable rows from the dataset as well as the creation of needed columns, as seen in the code below.

```
def wrangle(file):
  df = pd.read_csv(file, encoding='ISO-8859-1')

  # Select needed columns
  df = df[['text','sentiment']]

  # Drop missing rows
  df.dropna(inplace = True)

  # Create a new index and add an id column for the models
  df['id'] = range(0,len(df))
  df['myID'] = range(0,len(df))
  df.set_index('id', inplace = True)

  return df
```

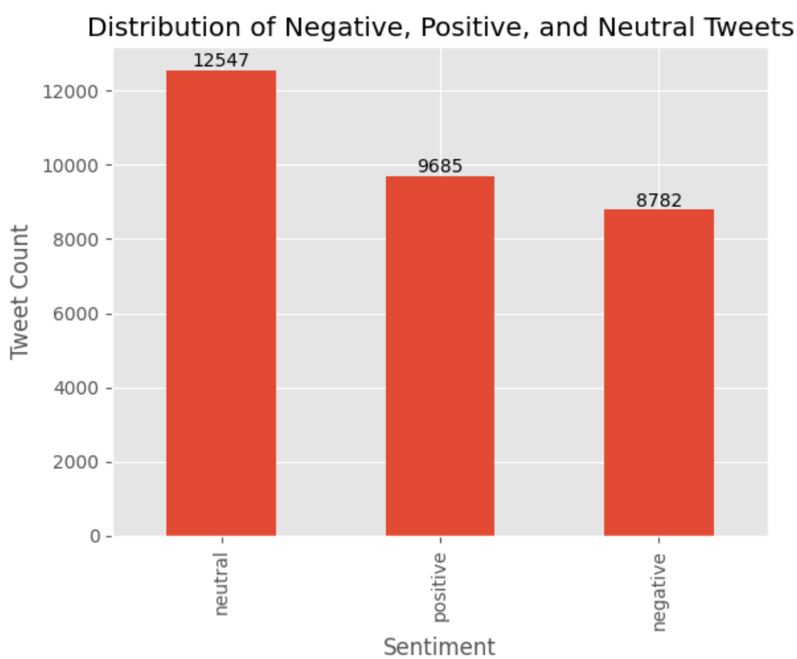

Figure 1: Your Plot Caption

Observing the distribution of our sentiment classes, it's clear that we have some class imbalance. While some might resample the data for analysis, we deliberately skipped this step to evaluate the models' performance on the raw, unaltered dataset.

3.2 Baseline

In the context of model building and evaluation, a baseline serves as a reference point or starting point against which the performance of more complex or sophisticated models is compared. It's a simple and often straightforward model or heuristic that provides a minimum level of performance, giving insight into the expected results without applying complex methods. There are various reasons why one might want to determine the baseline for a given dataset. Such reasons include but are not limited to:

1. Performance Evaluation: It offers a benchmark against which the performance of more advanced models can be measured. If a complex model cannot outperform the baseline, it indicates that the model might not be adding much value or needs further refinement.
2. Understanding Model Complexity: Baselines help in understanding how complex a problem is. If a simple baseline performs reasonably well, it might suggest that the problem itself is not very complex and might not require sophisticated models.
3. Establishing a Starting Point: They offer a starting point for developing and improving models. Once the baseline performance is established, it becomes a reference for further enhancements or modifications to achieve better results.

```
df.sentiment.value_counts(normalize = True).max()
```

Consequently, we proceeded to establish the dataset's baseline, using the code below:
df.sentiment.value_counts(normalize = True).max()
resulting in a value of 0.404, which equates to 40.4%. That is to say, if the model predicts 'neutral' for every tweet in this dataset, it will be right 40% of the time.

3.3 VADER Model

The VADER model gave an accuracy of 47% which is 7% above the baseline for this dataset. Looking at the confusion matrix for this model 2, it can be noted that it (the model) is more inclined to read tweets as neutral. I would have mentioned that this could be attributed to the training dataset, but considering that this is a pre-trained model, that can't be the case. Also, the model performed better at classifying positive tweets compared to its classification of negative tweets. It could be that the VADER model is better at classifying neutral and positive textual sentiment.

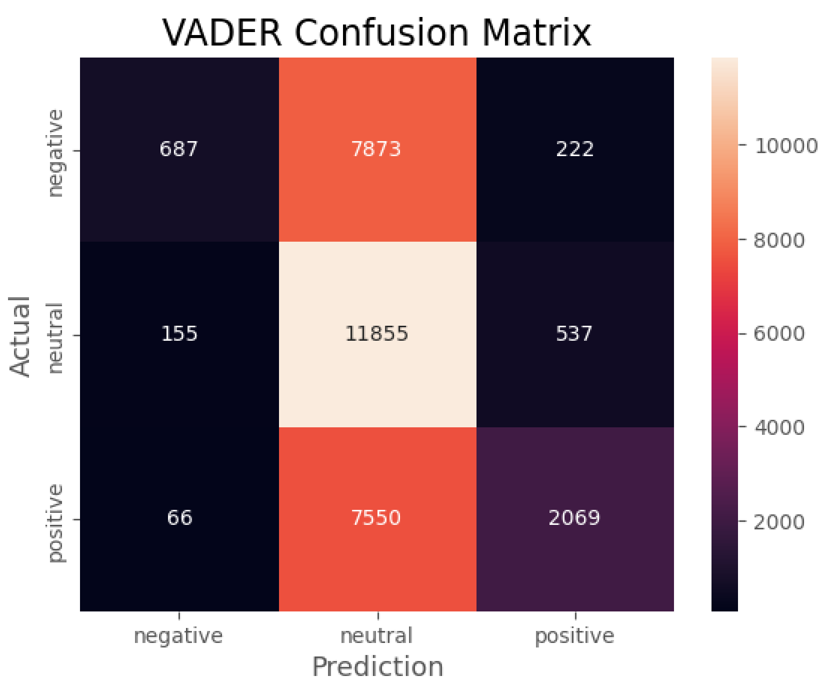

Figure 2: Confusion Matrix for VADER

The f1 scores of both the negative and positive tweets are poor as seen in the classification report of this model. This means that the model performed terribly at classifying negative and positive tweets. It is noteworthy that this model took 10 seconds to train.

3.4 RoBERTa Model

The RoBERTa model outperformed the VADER model by obtaining an accuracy score of 73%, that is a difference of approximately 26%. Looking at the confusion matrix of the RoBERTa model, it performed least when it came to the prediction of neutral tweets.

The f1 score for each class is fairly high (above 50%) with the positive class having the highest f1 score. This value of 0.71 for the positive class indicates that there exists a good balance between precision and recall.

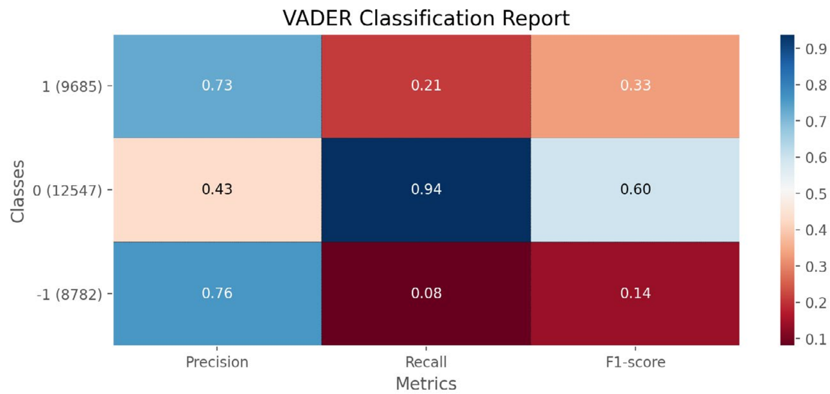

Figure 3: Classification Report for VADER

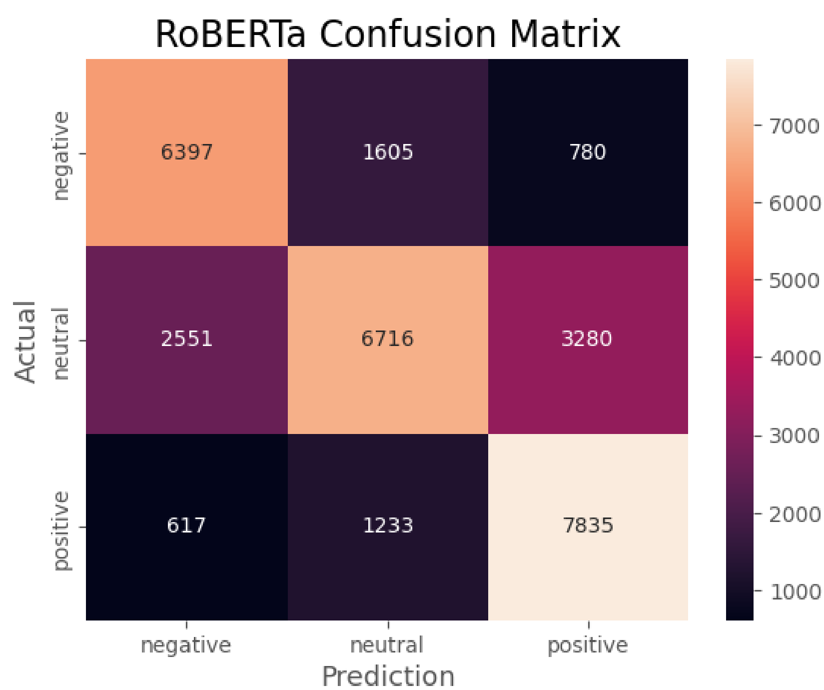

Figure 4: Confusion Matrix for RoBERTa

It means the model has high precision and high recall for the classification of positive tweets. This informs us on how beneficial this model can be when making crucial decisions especially in scenarios where both false positives and false negatives are equally important.

Having seen how robust the RoBERTa model is, it should be noted that this model is computationally expensive given that it took it 4687.38 seconds (78 minutes) to run on the dataset as opposed to the VADER model that took 10 seconds to compute.

4. Discussion and Conclusions

Considering that the F1 score considers both precision and recall, let's now examine and contrast the F1 scores for each sentiment in both models.

Sentiment	Vader	Roberta
Negative	0.14	0.70
Neutral	0.60	0.61
Positive	0.33	0.73

Table 1: F1 Score Comparison

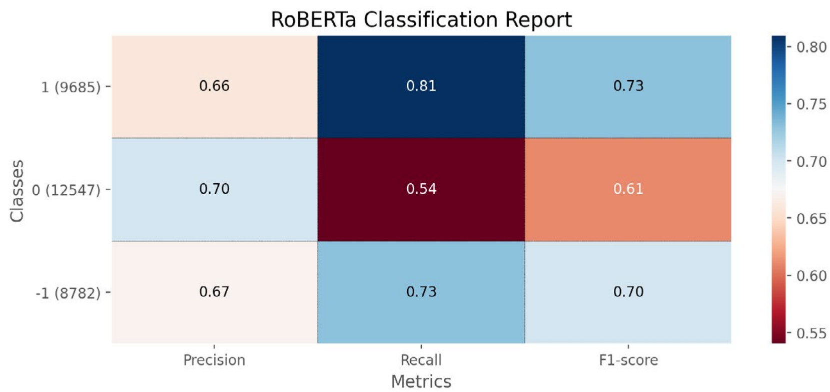

Figure 5: Classification Report for RoBERTa

VADER faces challenges in handling context-dependent sentiments and struggles with nuanced expressions. The model's reliance on predefined rules and sentiment scores from a lexicon can limit its adaptability to diverse domains and evolving language use.

RoBERTa's transformer architecture, with its attention mechanism and bidirectional context understanding, addresses many of the limitations of models like VADER. Its robust performance across multiple benchmarks and datasets underscores its efficacy in sentiment analysis tasks, even in challenging scenarios.

Comparing VADER and RoBERTa reveals a trade-off between simplicity and adaptability. VADER's simplicity makes it a pragmatic choice for certain applications, particularly those with real-time processing requirements. However, its limited adaptability to diverse contexts may hinder its performance in domains with complex linguistic expressions.

On the other hand, RoBERTa's deep learning approach empowers it to understand intricate relationships in context, making it suitable for applications demanding a higher level of sophistication. However, this comes at the cost of increased computational requirements and potential challenges in real-time deployment.

In conclusion, our comparative analysis of the Roberta and Vader models has shed light on their respective strengths and limitations in sentiment analysis tasks. While Roberta demonstrates superior performance in capturing nuanced sentiments and context, Vader proves efficient for quick, rule-based analyses. The choice between these models depends on the specific requirements of the task at hand.

Our study contributes to the ongoing discourse on sentiment analysis methodologies, providing insights into the nuanced performance trade-offs between state-of-the-art deep learning models like Roberta and rule-based approaches like Vader. As the field continues to evolve, further research could explore hybrid approaches that harness the strengths of both paradigms.

References

1. Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. Foundations and Trends® in information retrieval, 2(1–2), 1-135.
2. Ouyang, X., Zhou, P., Li, C. H., & Liu, L. (2015, October). Sentiment analysis using convolutional neural network. In 2015 IEEE international conference on computer and information technology; ubiquitous computing and communications; dependable, autonomic and secure computing; pervasive intelligence and computing (pp. 2359-2364). IEEE.
3. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
4. Zhou, J., Lu, Y., Dai, H. N., Wang, H., & Xiao, H. (2019). Sentiment analysis of Chinese microblog based on stacked bidirectional LSTM. IEEE Access, 7, 38856-38866.
5. Adoma, A. F., Henry, N. M., & Chen, W. (2020, December). Comparative analyses of bert, roberta, distilbert, and xlnet for text-based emotion recognition. In 2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP) (pp. 117-121). IEEE.
6. Yani, M. A. A., & Maharani, W. (2023). Analyzing

Cyberbullying Negative Content on Twitter Social Media with the RoBERTa Method. JINAV: Journal of Information and Visualization, 4(1), 61-69.
7. Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. Ain Shams engineering journal, 5(4), 1093-1113.
8. Devika, M. D., Sunitha, C., & Ganesh, A. (2016). Sentiment analysis: a comparative study on different approaches. Procedia Computer Science, 87, 44-49.
9. Taboada, M. (2016). Sentiment analysis: An overview from linguistics. Annual Review of Linguistics, 2, 325-347.
10. Balahur, A., Steinberger, R., Kabadjov, M., Zavarella, V., Van Der Goot, E., Halkia, M., ... & Belyaeva, J. (2013). Sentiment analysis in the news. arXiv preprint arXiv:1309.6202.
11. Yelena, M. (2009). Sentiment Analysis: An Overview Comprehensive Exam Paper. Computer Science Department, University of Iowa.
12. Liu, B. (2010). Sentiment analysis and subjectivity. Handbook of natural language processing, 2(2010), 627-666.
13. Hussein, D. M. E. D. M. (2018). A survey on sentiment analysis challenges. Journal of King Saud University-Engineering Sciences, 30(4), 330-338.
14. Obulezi, O. J., Chinedu, E. Q., Oramulu, D. O., Etaga, H. O., Onyeizu, N. M., & Ejike, C. O. (2023). Machine Learning Models for Predicting Transportation Costs Inflated by Fuel Subsidy Removal Policy in Nigeria. Int. Res. J. Mod. Eng. Technol. Sci, 5, 1053-1070.
15. Nwankwo, M. P., Onyeizu, N. M., Asogwa, E. C., Ejike, C. O., & Obulezi, O. J. (2023). Prediction of House Prices in Lagos-Nigeria Using Machine Learning Models. European Journal of Theoretical and Applied Sciences, 1(5), 313-326.
16. Gupta, P., Mehlawat, M. K., Khaitan, A., & Pedrycz, W. (2020). Sentiment analysis for driver selection in fuzzy capacitated vehicle routing problem with simultaneous pick-up and drop in shared transportation. IEEE Transactions on Fuzzy Systems, 29(5), 1198-1211.
17. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. Advances in neural information processing systems, 33, 1877-1901.