**Journal of Computing and Electronic Information Management**
ISSN: 2413-1660 | Vol. 15, No. 3, 2024

# A comparative study of e-commerce review sentiment analysis models based on VADER and RoBERTa

**Yongli Bao***

Chongqing University of Posts and Telecommunications, Chongqing, 400065, China

* **Corresponding author Email:** 2021215000@stu.cqupt.edu.cn

**Abstract:** The study compares the performance of two sentiment analysis models - VADER and RoBERTa - in analyzing Amazon product reviews. Using a dataset of reviews containing different star ratings, the study performed sentiment analysis on reviews using the VADER and RoBERTa models, respectively, and analyzed the performance of both on positive, neutral, and negative sentiment scores. The results show that the RoBERTa model performs well in capturing complex sentiment and contextual information, and especially has stronger recognition ability on extreme sentiments (e.g., 1-star and 5-star reviews.) The VADER model, although lightweight and fast, predicts weak correlation between sentiment scores and actual user ratings when dealing with complex semantics and long texts. This study provides reference value for sentiment analysis of user reviews on e-commerce platforms.

**Keywords:** Sentiment Analysis, VADER, RoBERTa, NLP.

## 1. Introduction

Sentiment analysis belongs to a natural language processing technology derived from the field of cross-development of linguistics and computer science, and the main purpose of sentiment analysis in the mainstream view is divided into three stages, which are to judge the subjectivity of the text, to judge the polarity of the text, and to determine the intensity of the text to express emotion. Emotional tendencies in sentiment analysis can be generalized into positive, neutral and negative [1].

Sentiment analysis has a wide range of real-life applications including, but not limited to, feedback evaluation, marketing, and regulating social platform discourse, such as companies attempting to gain public opinion on their products by using sentiment analysis models to identify and mine relevant posts and comments on their products to gain different opinions on their products. Shopping sites can analyze users' preferences based on their historical comments to better surmise which products the user should be targeting. Through sentiment analysis, social networking site administrators can more easily monitor the verbal aggressiveness of some posts or comments to create a favorable social networking environment [2].

There are many researches for sentiment analysis, for example, early Joscha tried Bag of words models, n-gram and other methods for sentiment analysis, but these methods do not take into account the semantic correlation between sentences, so they are less effective [3].Ahmad Kamal proposed a framework to help in sentiment analysis, feature extraction or summarizing comments by using supervised machine learning methods such as Naive Bayes, Decision Tree, Multilayer Perceptron, etc.[4].Kumara et al. utilized Support Vector Machines (SVMs), logistic regression and k-nearest neighbor machine learning algorithms, count vectors and TF-IDF mechanism to determine the sentiment of Twitter posts and obtained that Logistic regression with count Vectorizer is the most accurate model vectorization combination with 88.26% accuracy [5].Chiorrini et al. used BERT and defined for the task two independent classifiers to perform sentiment analysis of Twitter text [6]. Another study proposed a seven-layer framework for analyzing sentence sentiment. The framework utilizes Convolutional Neural Network (CNN) and Word2vec to compute the vector representation and perform sentiment analysis (SA), respectively. In order to improve the accuracy and generalization of the model, the researchers also used techniques such as dropout, normalization, and Parameterized Rectified Linear Units (PReLU), which achieved relatively good results [7].

The aim of this study is to compare VADER and RoBERTa models by performing sentiment analysis on the same dataset and comparing the two models in terms of applicability, accuracy, and judgment of extreme sentiment.

## 2. Material and Method

### 2.1. Data source

The dataset chosen for this study is Amazon product reviews [8]. This dataset consists of more than 568000 consumer reviews of different Amazon products. This dataset consists of Id, ProductId, UserId (User ID who is reviewing), ProfileName (User Profile Name who is reviewing), HelpfulnessNumerator (Numerator for Helpfulness of review), HelpfulnessDenominator (Denominator for Helpfulness of review), Score, Time, Summary and text for a total of ten columns.Due to the large amount of data in the original dataset, this study only takes the first 10000 entries for sentiment analysis.

After intercepting the dataset, the bar chart shown in Figure 1 is used to visualize the distribution of the number of reviews with each score among the first 10,000 reviews, where five scores represent the positive reviews with the strongest sentiments, and one score represents the poor reviews with the strongest sentiments.

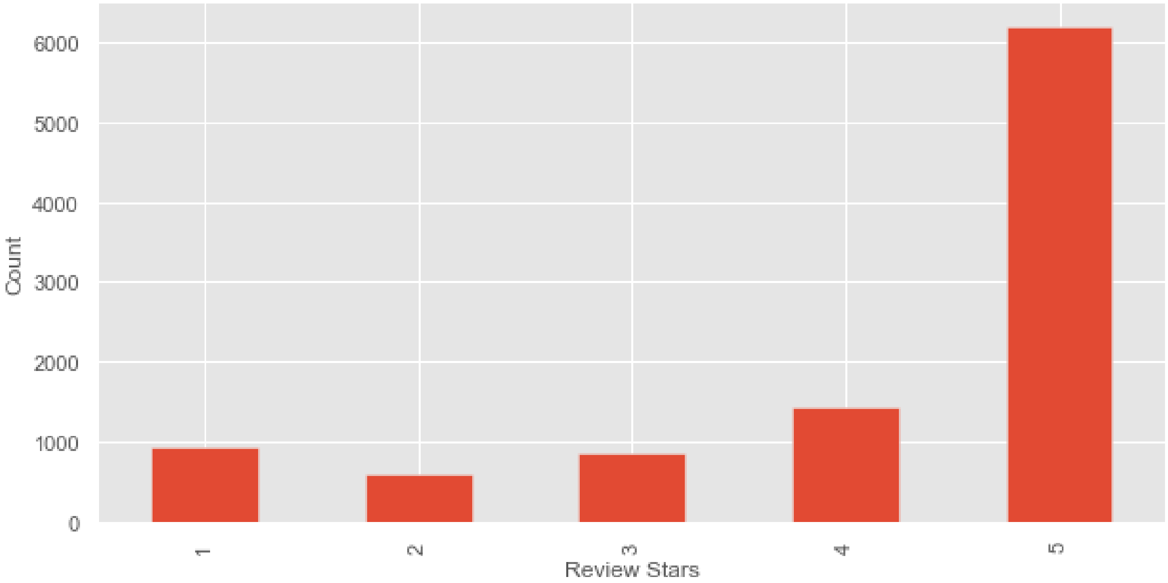

**Fig. 1.** Distribution of dataset ratings

## 2.2. Method

### 2.2.1. VADER

The first model in this study is SentimentIntensityAnalyzer provided by VADER (Valence Aware Dictionary for sEntiment Reasoning) sentiment analysis toolkit.SentimentIntensityAnalyzer is a dictionary-based and rule-based sentiment analysis model that has been trained by its authors specifically for social media messages, so it can be said to be designed specifically for social media text sentiment analysis [9]. Since SentimentIntensityAnalyzer is a dictionary-based model that does not require a large amount of training data, can quickly calculate sentiment scores, and has the advantages of being lightweight and easy to use, it is considered suitable for real-time sentiment analysis tasks.

SentimentIntensityAnalyzer features a lexicon-based scoring model, where each word in SentimentIntensityAnalyzer's sentiment lexicon is associated with a sentiment intensity score (typically -4 to +4), which represents the intensity of the sentiment expressed by the word. In addition to this, SentimentIntensityAnalyzer has a number of expressions that are judged to be sentiment-enhancing, such as capital letters, punctuation marks-especially exclamation points, emoticons, and various negatives. In addition, as mentioned above, the author of SentimentIntensityAnalyzer has specifically built his lexicon based on social media data, so SentimentIntensityAnalyzer recognizes and understands a large number of slang, Internet terms and abbreviations, which makes it ideal for dealing with texts in informal language such as product reviews.

SentimentIntensityAnalyzer works in a very simple way, it first segments the recognized text into individual words or phrases, and then looks up the corresponding sentiment scores in its sentiment dictionary. Before calculating the sentiment score, SentimentIntensityAnalyzer also recognizes the context of the text and adjusts the sentiment score accordingly, e.g., the exclamation point mentioned above may strengthen the sentiment score, while the negative word may lower or reverse the sentiment score. After these operations, SentimentIntensityAnalyzer calculates a composite score that represents the overall sentiment tendency of the text. The composite score usually ranges from -1 to +1, where a composite score greater than 0.05 indicates that the text conveys a positive sentiment. A composite score between -0.05 and 0.05 indicates that the text is neutral in sentiment. A composite score of less than -0.05 indicates that the text conveys a negative sentiment. In addition to the composite score, SentimentIntensityAnalyzer provides three separate sentiment scores - “pos”, “neu “ and “neg”. These scores indicate the relative strength of the text on different sentiment dimensions, thus helping people to better understand the sentiment of the text.

Steps of Sentiment Analysis using SentimentIntensityAnalyzer in the experiment,First initialize the SentimentIntensityAnalyzer of VADER. After this initialize an empty dictionary to store the subsequent sentiment scores. Then traverse the processed dataset and store the id of each text and its corresponding sentiment score in the dictionary. By doing this it helps to differentiate between different texts and their corresponding sentiment scores.

In order to visualize the distribution of different sentiment texts in the data set, the dictionary with sentiment scores is firstly converted to Data Frame and transposed to id as the row index, and finally the scoring results of SentimentIntensityAnalyzer are merged with the original data set to retain the original texts.

Finally, the results of SentimentIntensityAnalyzer are visualized using icons. Figure 2 shows the relationship between different star ratings and compound sentiment scores. The higher the rating, the higher the compound score accordingly, and the figure illustrates that the VADER model successfully captures positive sentiment for high star ratings and negative sentiment for low star ratings.

Figure 3 details the distribution of positive, neutral, and negative sentiment scores, with the first subplot (Positive) showing the relationship between star ratings and positive sentiment scores. Typically, reviews with higher ratings also have higher positive sentiment scores, indicating that the more satisfied the user is, the stronger the positive sentiment in the review. The second subplot (Neutral) shows neutral sentiment scores, which do not usually change significantly with star ratings. The third subplot (Negative) shows the relationship between star ratings and negative sentiment scores. Lower star ratings (e.g., 1 and 2 stars) typically exhibit higher negative affective scores, while higher star ratings (e.g., 4 and 5 stars) have lower negative affective scores.

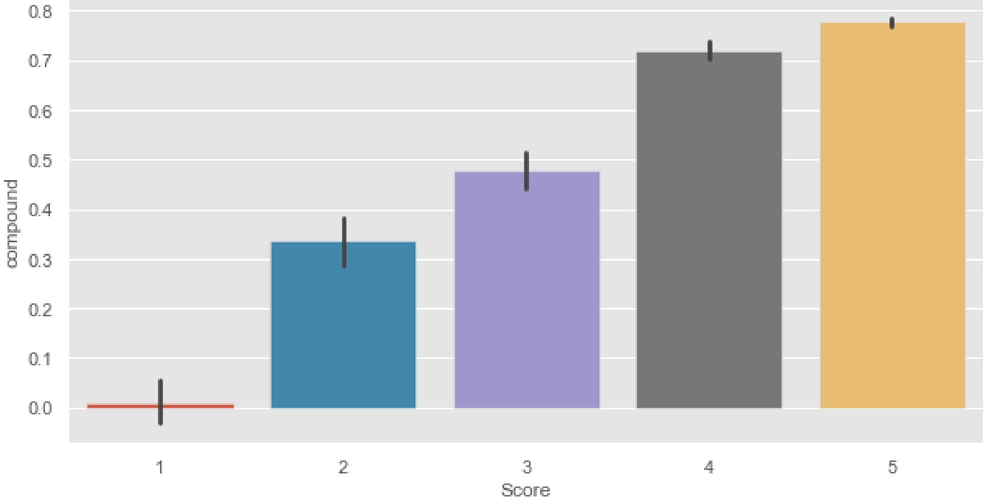

**Fig. 2.** Relationship between star ratings and compound sentiment scores

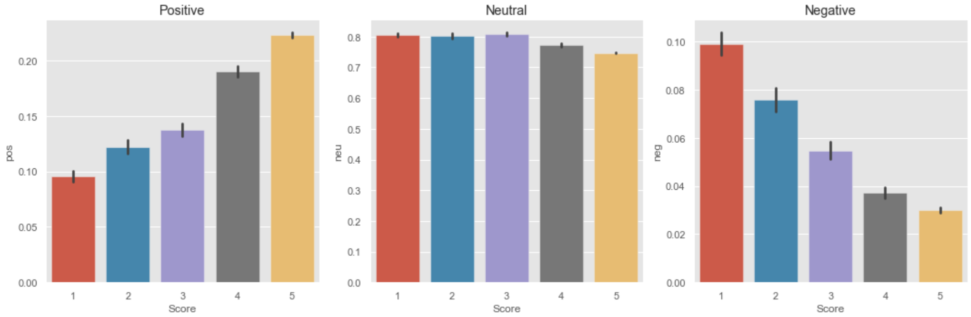

**Fig. 3.** Distribution of positive, neutral and negative sentiment scores

### 2.2.2. RoBERTa

RoBERTa (Robustly optimized BERT approach) is an improved version of BERT model introduced by Facebook & University of Washington [10].BERT model is a natural language processing model introduced by Google based on the Transformer architecture.BERT model has powerful semantic understanding capabilities. It is capable of capturing the bidirectional meaning of words from context through the self-attention mechanism of bidirectional encoder, which has better performance.RoBERTa has several important improvements over BERT, the first is that RoBERTa uses a much larger dataset, a total of 160 GB of textual data, which includes books, news, Wikipedia, and so on, and the training data is nearly ten times that of BERT . Larger training data also brings longer training time. Therefore RoBERTa also uses a larger Batch and learning rate to further improve the efficiency of model learning. Second, RoBERTa also removes the Next Sentence Prediction (NSP) task, which was added to BERT retraining to allow the model to predict whether two sentences before and after are related. RoBERTa removed this task because the researchers concluded that NSP would provide little improvement to the model when RoBERTa was trained on a large amount of data.

In addition, RoBERTa changed to use dynamic masks.When BERT is trained, a mask is performed once during data preprocessing to get a static mask. Whereas RoBERTa uses dynamic masking during training, RoBERTa regenerates the masked words in each epoch. In this way, the model gradually adapts to different masking strategies and learns different linguistic representations during the continuous input of large amounts of data.

RoBERTa still works on the BERT-based Transformer Encoder architecture, which relies on a “self-attention mechanism” to understand the context. First, there is an input embedding layer, where the input text is sliced into words and transformed into word embedding representations, which are fed into the encoder layer of the model. Then there is a multi-layer bi-directional Transformer encoder. With the bi-directional encoder, the model is able to learn the meanings of the words simultaneously from the context of the text, which is crucial for accurately understanding the semantics of the language. At each training stage, RoBERTa randomly masks some words and trains the model to predict these masked words. Finally, at the output layer, the model predicts the masked words based on information from the preceding and following text.

In this experiment, the pre-trained RoBERTa model is first loaded and then the input text is converted into a format acceptable to the model using AutoTokenizer. For sentiment analysis, the comment text is first encoded into tensor format using tokenizer(), and then the encoded text is subjected to sentiment prediction to obtain the un-normalized scores. Finally, the scores are then converted into probability distributions using softmax(scores) to obtain probability values for three categories: negative, neutral, and positive. After completing the sentiment analysis, a dictionary was created to store the three sentiment scores (negative, neutral, and positive). Afterwards, the RoBERTa sentiment analysis logic is encapsulated into the polarity_scores_roberta function, which is conveniently called for each comment, and then batch sentiment analysis is performed.

## 3. Results Analysis

The results from the previous SentimentIntensityAnalyzer with RoBERTa batch sentiment analysis were re-indexed and integrated into the original dataset for further analysis. In order to facilitate our comparison of the two models, this study uses Seaborn's pairplot function to plot the correspondence between the sentiment score variables, on top of which the data are colored according to Score to help observe the distribution of different star ratings on the sentiment scores.

Figure 4 shows the relationship between different sentiment scores in sentiment analysis for both VADER and RoBERTa models, and how these scores vary with the star rating (Score) of the product. The horizontal and vertical coordinates represent the VADER sentiment scores: vader_neg (negative), vader_neu (neutral), vader_pos (positive) and RoBERTa sentiment scores: roberta_neg (negative), roberta_neu (neutral), roberta_pos (positive), respectively. As can be seen in Figure 4, RoBERTa's pos and neg scores are more pronounced with star rating, and usually more accurately reflect the distribution of emotions in high (positive) and low (negative) star ratings. In contrast, VADER tends to perform less well than RoBERTa in complex sentiment and satirical texts. for example, in some 1- or 5-star reviews, VADER may give sentiment scores that do not match the star rating. This suggests that RoBERTa outperforms VADER in extreme sentiment detection.

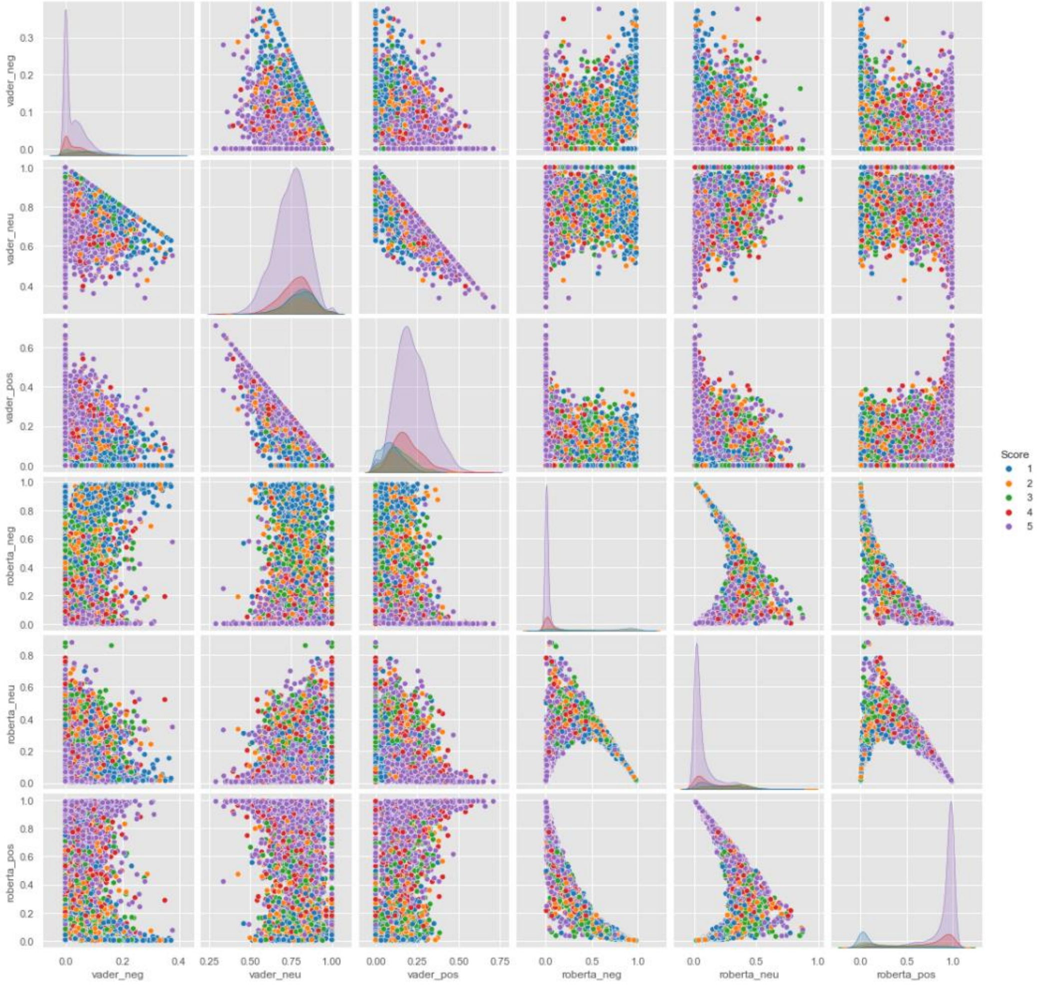

**Fig. 4.** Plot of the relationship between VADER and RoBERTa sentiment scores

## 4. Conclusion

This study reveals the difference in performance between two sentiment analysis models, VADER and RoBERTa, by comparing their performance across different sentiment scores and star ratings. the RoBERTa model performs better overall: RoBERTa is better able to capture complex sentiments in text, especially in complex utterances with long comments. And it performs well on sentiment analysis of texts with extreme ratings (e.g., 1 and 5 stars), accurately identifying both positive and negative sentiment. And although VADER performs well on short texts and informal texts with lower application costs, it has a more limited understanding of complex syntax and context. This is especially evident in sentiment analysis of long comments or complex emotional expressions, where the sentiment score matches the actual star rating relatively poorly. Therefore, RoBERTa is a more suitable choice for application scenarios that require high-precision sentiment analysis, such as the sentiment analysis of complex product reviews.VADER, on the other hand, is suitable for scenarios that have limited computational resources or for sentiment analysis tasks that require fast processing of a large number of short texts.

This study is useful in real-world applications of sentiment analysis, for example, due to RoBERTa's excellent comprehension of long texts, it can be applied to relevant ai Q&A systems, as well as to the sentiment analysis of complex reviews. VADER's lightweight features can support some real-time sentiment analysis. The application should be based on the specific task to choose the appropriate model. Subsequent research can compare other big models, such as llama and other big models, so as to compare the different characteristics of different sentiment analysis models more perfectly.

# References

[1] Taboada, M. (2016). Sentiment analysis: An overview from linguistics. Annual Review of Linguistics, 2(1), 325-347.

[2] Baid, P., Gupta, A., & Chaplot, N. (2017). Sentiment analysis of movie reviews using machine learning techniques. International Journal of Computer Applications, 179(7), 45-49.

[3] Märkle-Huß, J., Feuerriegel, S., & Prendinger, H. (2017). Improving sentiment analysis with document-level semantic relationships from rhetoric discourse structures.

[4] Kamal, A. (2015). Review mining for feature based opinion summarization and visualization. arXiv preprint arXiv: 1504. 03068.

[5] Jayakody, J. P. U. S. D., & Kumara, B. T. G. S. (2021, December). Sentiment analysis on product reviews on twitter using Machine Learning Approaches. In 2021 International Conference on Decision Aid Sciences and Application (DASA) (pp. 1056-1061). IEEE.

[6] Chiorrini, A., Diamantini, C., Mircoli, A., & Potena, D. (2021, March). Emotion and sentiment analysis of tweets using BERT. In Edbt/icdt workshops (Vol. 3, pp. 1-7).

[7] 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.

[8] Amazon Product Reviews, Kaggle (2021).

https://www.kaggle.com/datasets/arhamrumi/amazon-product-reviews

[9] Hutto, C., & Gilbert, E. (2014, May). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In Proceedings of the international AAAI conference on web and social media (Vol. 8, No. 1, pp. 216-225).

[10] Liu, Y. (2019). Roberta: A robustly optimized bert pretraining approach. arXiv preprint arXiv:1907.11692.