# Sentiment Analysis: The better one?

**Harsh Sharma**
**Purdue University**
**Fort Wayne**
*sharh02@pfw.edu*

## Abstract

As more and more data is generated each day by millions of users online, classifying that data has become more of an urgent task than ever. One way of classifying this humongous of a data is sentiment analysis. It is one of the tasks under the paradigm of Natural Language Processing. There are multiple ways/models which are utilised to achieve sentiment classification based on the task at hand. The purpose of this paper is to compare the various models available at the disposable and analyse their results. And to answer which model, for sentiment analysis, is the **better one**?

## 1 Introduction

If you were to buy anything online, the one thing that can't be missed is reading the reviews. One would like to know about other people's opinions before buying that new Cinderella cup or deciding to buy a house in which locality. And so with the help of the internet, we can know about millions of people's opinions and this has newly created the need for sentiment analysis.

Sentiment Analysis, or Opinion Mining, is a field which unites fields like, computational linguistics, data mining, machine learning, and even psychology[1]. The objective is to find the opinion or the sentiment behind the text. [1]

## 2 Motivation

Humans are complicated, their emotions more so. Even people have trouble understanding the emotions of people around them. And hence delegating the task of understanding emotions to a machine can be very helpful, but also very challenging. And therefore, the personal motivation was being able to comprehend as to how a machine is understanding human emotions.

But sentiment analysis also has industry use cases and effect real businesses and people in more than one way.

### 2.1 Sentiment Analysis for Businesses

It is believed that people buy emotions, not products. And so, from a standpoint of a business, it becomes necessary to determine what is the view of the general masses.

SA helps businesses in numerous ways:

- Companies can monitor customer sentiments in real-time. It aids in identifying business problems before they escalate, especially at times of a new product launch.

- Companies can estimate the ROI of marketing campaigns by evaluating negative and positive conversations and opinions shared among customers. And hence plan accordingly for the next campaign.

### 2.2 Sentiment Analysis for Preventing Hate Speech

Hate speech plays a consequential role in promoting violence, from violent extremism to atrocity crimes [2]. Hate speech has evolved in complexity and scope. The widespread use of social media has given hate speech a larger audience. It's considered as a form of cultural violence, which pushed many ordinary people to glorify and commit violent action against other people.

Using the concepts of sentiment analysis, toxic comments spread through social media and online can be detected. By flagging them, the platform can take proper actions against the comments/or even against the user. The platforms can ban the user if they are repeatedly spreading hate speech.

## 3 Related Work

*Sentiment Analysis: An Overview* is a paper authored by *Yelena Mejova* which provides a good

[1] The code for the project is available in this GitHub Repository

starting point for understanding sentiment analysis[3]. Whereas, *Sentiment analysis algorithms and applications: A survey*[4] co-authred bby *Walaa Medhat* explains the architecture of the process. And, *Sentiment Analysis: It's Complicated!* gives a good understanding about the models, co-authred by *Kian Kenyon-Dean*[5]

## 4 Methodology

### 4.1 Approach

To perform sentiment analysis there are multiple approaches:

- **Lexicon Based**
- **Rule Based**
- **Hybrid Based**
- **Machine Learning Based**
  - Pre-Trained Model
  - Fine-Tuning Model

Each methodology uses a different approaches to the same problem of sentiment analysis. All the above mentioned approaches have been tested and compared using their respective models.

### 4.2 Dataset

I've used iMDb sentiment analysis dataset. This dataset contains 49582 unique values of movies review. For labels, it has two possible values, 'positive' and 'negative'. To train the model, I'm converting the sentiment labels into numerical values of 1s and 0s. Also, I tried Amazon product review dataset but in the end I was getting better results for iMDb in case of fine-tuning the RoBERTa model. So the final performance metric is calculated for the iMDb dataset.

## 5 Experiments

### 5.1 AFINN Model

Affective Norms for English Words contains list of English words with associated sentiment scores, ranging from -5 (very negative) to +5 (very positive).

This model uses **Lexicon based approach**. Therefore, it relies on pre-defined lists of words and their associated sentiment scores to determine the sentiment of a piece of text.

Each word in the input text is looked up in the AFINN lexicon, and the sentiment score of the word is retrieved and then calculated to find the final score. Below here, the table showcases how the AFINN model will assign the sentiment scores. Ex: "This is a bad movie."

| **Word** | **Score** |
|---|---:|
| This | 0 |
| Is | 0 |
| A | 0 |
| Bad | -3 |
| Movie | 0 |

Table 1: This table represents how the model is assigning sentiment score to each of the respective words.

At the end, the model will calculate each score to get the final sentiment score for the sentence. Here,

**Final score: 0 + 0 + (-3) + 0 = -3**

Here, zero is considered as the threshold. Therefore, since the final score is negative, the entire sentence is given a negative prediction. Which is also accurate in our case.

#### 5.1.1 Performance Metric

| **Accuracy** | **F1-score** | **Precision** | **Recall** |
|---|---|---|---|
| 70.26% | 74.45% | 65.37% | 86.19% |

Table 2: Performance of AFINN Model

Table 3: AFINN Classification Report

| | precision | recall | f1-score | support |
|---|---:|---:|---:|---:|
| negative | 0.80 | 0.54 | 0.65 | 24747 |
| positive | 0.65 | 0.86 | 0.74 | 24753 |
| accuracy | | | 0.70 | 49500 |
| macro avg | 0.73 | 0.70 | 0.69 | 49500 |
| weighted avg | 0.73 | 0.70 | 0.69 | 49500 |

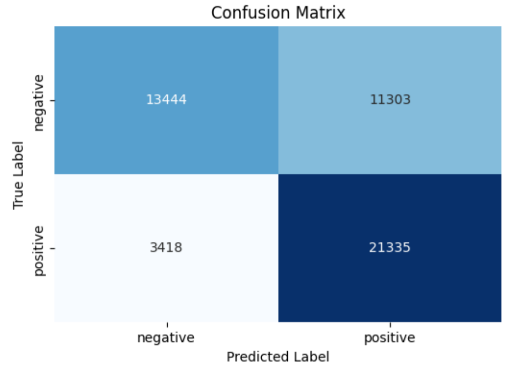

Figure 1: AFINN Confusion Matrix Distribution

### 5.1.2 Limitations

This is a very simple approach towards sentiment analysis. Since it gives a score to each word, relative relation is completely ignored. For example, it is not able to handle double negatives. To demonstrate, consider the same example but with double negative: The movie is **not bad**.

The model predicts it to be negative, with a final sentiment score of -3. Clearly, it is not able to handle double negatives. **This showcases that the word relation is not taken into consideration, which is a big drawback.**

## 5.2 VADER Model

Valence Aware Dictionary and sEntiment Reasoner is a lexicon and rule-based sentiment analysis tool that is specifically designed to handle social media text.

In this **hybrid approach**, a sentiment lexicon is used as a base, but additional rules and heuristics are added to handle exceptions, ambiguities, and context-dependent sentiment. We also need rules to have specialised knowledge and so they can be domain specific.

Social Media contains slangs, emojis, highly informal, and are context-dependent. And since VADER is social-media domain specific, it can handle the above exceptions.

VADER consists of a sentiment lexicon that contains over 7,500 lexical features. It also has rules for intesifiers and negations.

Here is the same input from the last model: The movie is bad.

**Model Prediction Score: 'neg': 0.538, 'neu': 0.462, 'pos': 0.0, 'compound': -0.5423**

The model predicts correctly, as expected from a hybrid model. Now the example where the last model failed.

Test Statement: The movie is not bad. **Score: 'neg': 0.0, 'neu': 0.584, 'pos': 0.416, 'compound': 0.431**

Here the **neg** is **0.0** which shows that the sentence is not negative, whereas the **pos** is **0.416** and **neu** is **0.584** which represents the sentence is neutral to positive.

### 5.2.1 Performance Metric

| **Accuracy** | **F1-score** | **Precision** | **Recall** |
|---|---|---|---|
| 69.64% | 73.82% | 64.89% | 85.60% |

Table 4: Performance of VADER Model

The performance metric helps to analyse that the VADER Model is **not any better performing than AFINN**. In fact, it has worse scores. It wasn't expected, as it's a more sophisticated model than AFINN. One reason for this unusual result could be the use case VADER is designed for. It contains rules for social media and is designed to be **social media domain specific**. And here, we are using the iMDb dataset which is slightly different than the VADER-targeted dataset.

Table 5: VADER Classification Report

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| negative | 0.79 | 0.54 | 0.64 | 24747 |
| positive | 0.65 | 0.86 | 0.74 | 24753 |
| accuracy | | | 0.70 | 49500 |
| macro avg | 0.72 | 0.70 | 0.69 | 49500 |
| weighted avg | 0.72 | 0.70 | 0.69 | 49500 |

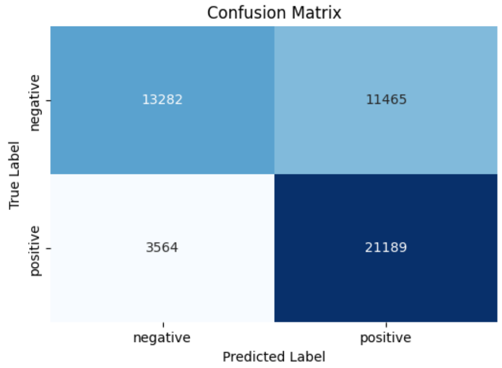

Figure 2: VADER Confusion Matrix Distribution

### 5.2.2 Limitations

Clearly, VADER performs better than AFINN, but it also has its limitations. The sentences which don't have emotionally strong words/lexicons, the VADER faces problem in giving proper scores to them. For ex, consider this sentence:

**Test Sentence: Movie's director used a Christopher Nolan's approach, which you don't find much in Hollywood.**

Here the user is comparing the direction of a movie to be similar of Christopher Nolan. Which is seemingly a top most compliment a director could get. But it's very niched to movie world. The VADER gives us following sentiment analysis,

**Score: 'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0**

Since the sentence doesn't contain any strong emotional word, VADER treats the test sentence to be neutral. But it isn't the case. VADER fails in task where very specialised/niche knowledge is required. And hence, VADER also fails.

## 5.3 Naïve Bayes Model

In Naïve Bayes model we try to find the probability of a given text document belonging to a particular class, such as positive or negative sentiment. The model uses the features of the text, such as the occurrence of specific words or phrases, to calculate the conditional probability of the document belonging to each class (positive/negative).

The Naïve Bayes uses a bag of words approach which means it doesn't count in for relative word relation. The frequency of the word is also not considered which makes it a little faster as we strip off all the repetitions.

### 5.3.1 Performance Metric

| Accuracy | F1-score | Precision | Recall |
|---|---|---|---|
| 85.00% | 84.49% | 87.55% | 81.64% |

Table 6: Performance of Naïve Model

Table 7: Naïve Classification Report

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| negative | 0.83 | 0.88 | 0.85 | 4994 |
| positive | 0.88 | 0.82 | 0.84 | 5006 |
| accuracy | | | 0.85 | 10000 |
| macro avg | 0.85 | 0.85 | 0.85 | 10000 |
| weighted avg | 0.85 | 0.85 | 0.85 | 10000 |

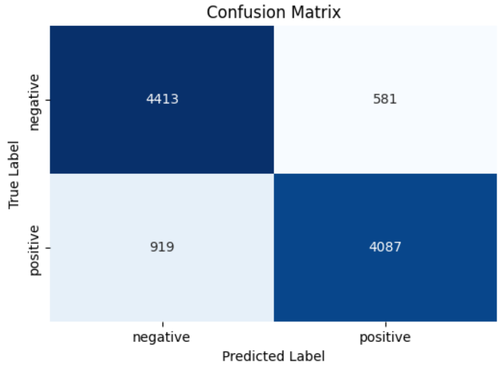

Figure 3: Naïve Confusion Matrix Distribution

### 5.3.2 Limitations

Although 85 per cent accuracy is remarkable, but it can be better(as we'll see with RoBERTa). There are some limitations to Naïve Bayes Model.

- It assumes that the features are independent of each other, and therefore it can't capture the complex relationships and doesn't perform well in sarcastic text.
- Takes stop words and other fillers into consideration
- Can't handle out-of-vocabulary words

## 5.4 RoBERTa Model

Training a model from scratch has its limitations, like smaller datasets, limited resources, etc. A pre-trained model helps to overcome such issues. One such model is RoBERTa.

Based on BERT architecture, the Robustly Optimized BERT Pretraining Approach(RoBERTa) is a pre-trained open-source model developed by hugging face(Facebook AI Research). RoBERTa model is trained on a large, diverse corpus of text data, including books, articles, and web pages.

But to have better performance out of this pre-trained model, I've fine-tuned it using the same dataset. It also helps to narrow the model to have better accuracy for our dataset.

### 5.4.1 Performance Metric

| Accuracy | F1-score | Precision | Recall |
|---|---|---|---|
| 93.80% | 93.32% | 94.54% | 92.13% |

Table 8: Performance of RoBERTa Model

Table 9: RoBERTa Classification Report

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| negative | 0.93 | 0.95 | 0.94 | 530 |
| positive | 0.95 | 0.92 | 0.93 | 470 |
| accuracy | | | 0.94 | 1000 |
| macro avg | 0.94 | 0.94 | 0.94 | 1000 |
| weighted avg | 0.94 | 0.94 | 0.94 | 1000 |

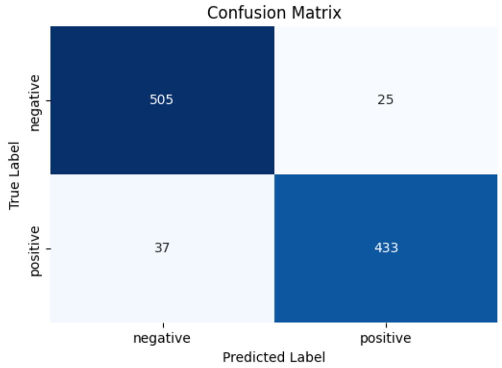

Figure 4: RoBERTa Confusion Matrix Distribution

### 5.4.2 Limitations

- It's a very large model, with over 300 million parameters. And so while fine-tuning it took very long even with good resources (4 GB NVIDIA Graphic card).
- Also, since it's already pre-trained we can't be clear with what datasets were used to train it, and hence harder to determine if it's better for a specific usecase.
- As it's trained to be generalised, it performs bad in any domain specificity tasks.

## 6 Results

The above experiments demonstrate that more complex models such as Naïve Bayes and RoBERTa can outperform simpler models like AFINN and VADER in terms of accuracy, F1 score, precision, and recall.

The performance metric for VADER and AFINN is almost similar, which wasn't the expected outcome. Since VADER uses a **hybrid** approach and is more sophisticated but performs worse than AFINN which only uses a much simpler approach, **lexicon**. It could be because of the dataset as it's not heavily social-media based which VADER is trained for. **So for non-social media sentiment analysis, AFINN is better than VADER**

Naïve Bayes performed pretty standard with 85 per cent accuracy. This result was expected from a standard complex model (with respect to the previous two).

The RoBERTa model performed very well after it was fine-tuned. Its accuracy is above 93 per cent and hence could be considered highly accurate compared to the rest of the models discussed. **Note that**, before it was fine-tuned it wasn't as accurate. So to answer the initial query, **fine-tuned RoBERTa model is better**.

The code for the project is uploaded in this GitHub Repository

## 7 References

The following are the references:

1. Sentiment Analysis: An Overview
2. United Nations
3. Paper Authored by Yelena Mejova
4. Sentiment Analysis: A Survey
5. Sentiment Analysis: It's Complicated!