# Predictive Model of Audience Rating of Franchise Films Using Sentiment Analysis

**Genevieve Ferguson**
Data Science Graduate Student
gferg020@fiu.edu

## Abstract

Sentiment analysis is a tool that can be used to examine any text and evaluate the level of intensity (strong/weak) and direction (positive/neutral/negative) of the sentiment of the speaker. This information can be very valuable to any person that can use the information about sentiment to make future decisions. In a business, understanding the consumer is a vital component of success. In this project, text data was scraped (using snscrape) from IMBD for each of the Marvel Movies and was used to create a total sentiment value that could be compared to that of the actual movie score from IMDB and RottenTomatoes. NLTK's VADER sentiment package (SentimentIntensityAnalyzer) was used to do the evaluation. To determine the efficacy of the sentiment analysis, 124 individual reviews were labeled and the accuracy of the predictions (Positive, Negative, and Neutral) were determined to be over 60 percent, slightly higher than some findings of around 56 percent. In addition, the compound scores were found to be drastically different from the numerical scores given on RottenTomatoes, but very similar to those given on IMDB. There can be many future improvements made to this study and to the sentiment analyzer, such as a higher accuracy measure through a better understanding of figurative language and slang.

## 1 Introduction

### 1.1 Social Media Discourse as a Marker for Product Performance

It is known that language, as it is an ever-changing and heavily irregular topic, can be difficult to accurately program due to ambiguity, syntax, semantics, data sparsity, and cultural context [3][5]. Several algorithms have been put in place to control for these obstacles, which allow us to process language at astounding speeds with greater accuracy than seen before. In this project, the goal was to implement that which we have learned in this natural language processing/computational linguistics course. In order to accomplish this, it was necessary to use some of the techniques and terms from the textbook, the lectures, and even the Watson papers. Extensive information about morphological processing, syntactical analysis, and semantic analysis had been taught in the course. Because of this, it was decided that the goal of the project was to take that a step further into a culmination of all that was learned. This paper looks into a topic that seems to be very popular lately due to the rise of social media platforms - sentiment analysis.

Sentiment analysis involves the use of machine learning algorithms to determine the intensity of a statement in relation to feeling [8]. For instance, a statement may be positive or negative, but it can also differ in intensity. The sentence "I hate..." is much more strongly negative than "I dislike...". Inversely, "I adore..." is a much more powerful statement than a simple "I like..." [7]. Because of the abundance of data that we are able to access using social media, sentiment analysis has become somewhat of a field on its own [9]. Businesses are able to create consumer profiles, track overall opinions of products and services, notice trends, and more using the easily accessible, public data via social media. As a result, a company is able to modify their business model and strategy in order to optimize production and revenue [9]. Knowing what a consumer wants and doesn't want to see is a good portion of what makes a product a total success or a complete flop. However, this is a major obstacle when you don't have an entire subdivision of a company devoting time, money, and energy into tackling this problem. To save on all of these resources, many businesses are looking toward technological advancements such as the aforementioned machine learning algorithms to discover the audience's opinion of their brand and products.

### 1.2 Obstacles for a Sentiment Analysis Model

Though it might be easy for a person to personally understand sentiment, the scale of the data is much too large for one person or even a whole team to tackle. And despite having the computing power to handle Big Data, it proves to be a challenging problem for a computer to solve for a number of different reasons [7]. Because all mechanisms of a computer are programmed, this means that there can be little room for error in mathematical equations and other straightforward factual problems. However, due to the several features of language that create abnormalities, sometimes the hyper intelligent computing devices can still struggle. Concepts such as ambiguity, syntax, semantics, data sparsity, and cultural context can prove to be hard for simple algorithms to understand.

## 2 Importance and Applications

As learned throughout the Natural Language Processing course, language is complex and can often be misinterpreted if the correct context is not identified. Although it would make the job much easier, a major problem with language is that on top of inherently being complicated, normal speech patterns - especially on social media - do not involve the use of full sentences and are full of spelling and grammar errors [12]. Therefore, teaching a computer to take all of these ambiguities into account is a major part of the task at hand. Many of the rules of social media have a psychological basis and feed into the idea of expressing oneself. It's important to also keep in mind that these online opinions may not represent the entire audience, as studies have shown that people are more likely to share their opinion when it is negative than when it is positive. In addition, the actions one takes as a user of social media serves as an extension of our opinion and can have meaning even without words.

### 2.1 Significance of Non-Text Sentiment

The nature of social media having 'posts' and 'comments', as well as 'likes', 'replies', and 'reposts' adds to the dimensions of meaning. Posts that share a certain opinion can have comments that are negative, meaning that they disagree with the original sentiment. On that same note, individual comments can convey an opinion that is disagreed with in a reply. Likes generally mean that an individual agrees with the original sentiment. Reposts usually means that an individual strongly agrees with the original sentiment. All of these accumulate to represent sentiment. Creating a model that correctly adjusts the effect of these parameters in order to get an accurate overall audience score would be the optimal result. However, due to the course having to do with language processing and not communication in general, these factors will be purposely ignored and only the text information from the social media content will be chosen to process through the program.

### 2.2 Potential Real-Life Applications

For large film franchises such as Star Wars, Marvel Studios, and Disney, it is extremely important to be able to know what the audience wants to see and what they enjoy the most. With millions and sometimes billions of dollars on the line, the stakes are high to pump out profitable content. As a result, sometimes the franchises become oversaturated or compromise quality according to the audience. The quality of latest movies from all of the three previously mentioned companies have been a point of contention between fans, especially on social media [2][11]. Though there seems to be some divide in each of these popular fandoms, by taking a look at the social media posts about the entirety of the Marvel franchise, perhaps we could determine if the audience's rating on popular websites match the sentiment of the text shown on social media platforms.

## 3 Methods

### 3.1 Data Retrieval

In order to retrieve the data, three methods were attempted but only two were implemented after one was found to be ineffective. Originally, the goal was to implement a popular Python library used to easily access the Twitter API, named tweepy. However, after applying for Twitter Developer access codes with no response, it was determined that a new method would need to be implemented in order to move forward with the project.

The first semi-successful method utilized the free "Twitter Scraper" program on Apify made by Quacker. The scraper was able to bypass the Twitter Developer requirements that had put the project on hold for almost an entire week. With this program, users are able to access information that exists on specific hashtags such as username, date of creation, the text of a tweet, as well as favorites, replies, and retweets. This is very similar to the functions of tweepy, with the main difference being that the data had to be saved to a file and then imported into the program rather than having a streamlined process. It was assumed that hashtags would have only tweets based on the topic in the hashtag, all of which would have some level of sentiment about what the hashtag represented. From the Twitter scraper, hashtags matching the names of the Marvel films were searched based on the most popular posts (ex:#BlackPanther). These were then saved to a csv file with the text from each tweet on each line.

The second method used was scrapy, a web scraper that extracts data from HTML and XML files [1]. By creating a parse tree of the entire document, we are able to find important features on specific webpages. The CSS method was used to select specific elements on the page. In this case, when the URL of an IMDB user review page was entered, it was then possible to retrieve the review title, the author username, the review date, the review text, and the number of 'helpfulness' likes that the review post received [1]. Then, the web drivers from selenium were imported to handle the HTML aspects of the project and automate the process of data retrieval. Because IMDB pages are standardized, the information was very easily accessed.

### 3.2 Sentiment Analysis Tools

To analyze the text from the tweets and reviews, several other packages were needed. For the tweet analysis, Numpy, matplotlib, and pandas were the first packages that were imported to handle many of the data handling and visualization portions of the program. The most essential part of the importing process was making sure that 'VADER' and 'TextBlob' were included, as these packages are what handled the actual sentiment analysis. VADER, or Valence Aware Dictionary and sEntiment Reasoner [10], uses a lexicon-based approach [12]. In this approach, dictionaries of words with pre-determined sentiment scores are used to match the words in a text, culminating into an overall sentiment score for the total block of text. In order to do this, the program first tokenizes the input text, breaking the words down into their base phrases and words, then evaluates and adds the lexicon values. TextBlob also uses tokenizers and a previously trained machine learning model to interpret the text [4]. A combination of the two packages can be used to possibly achieve a better result.

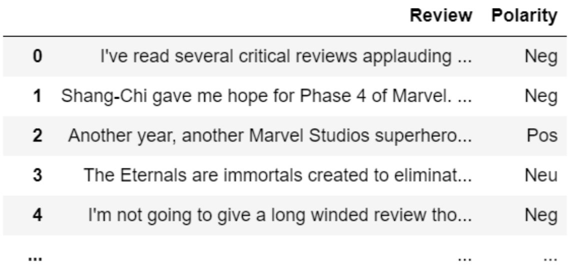
In this study, the analysis and tokenization was done by TextBlob and the scores were given by VADER's SentimentIntensityAnalyzer [10]. There are four scores that are returned with this package: positive, neutral, negative, and compound. The positive, neutral, and negative scores are all straightforward and self-explanatory. The compound score ranges from -1 to 1 and is a numerical representation of the entire text's sentiment, with -1 being strongly negative, 0 being neutral, and +1 being strongly positive. Both VADER and TextBlob's lexicon incorporates the intensity of individual words as well, as it gives stronger words higher sentiment scores [6]. For example, 'spectacular' would have a higher sentiment score than 'good'.

It is interesting to note that there are several features that are quite useful and make the analysis process much smoother and accurate. However, one major difference seems to make VADER the better lexicon for the problem at hand. This sentiment analysis tool, unlike many others, is able to process non-character symbols such as emojis, which are frequently used to convey emotions and opinions online. To add, the lexicon also accounts for repeated words and punctuations [6][12]. This results in a better understanding of the author's intention and opinion of the subject at hand, especially when it comes to social media.

A major issue with the first dataset was that there were several tweets that were unrelated to the topic in the hashtag, meaning that there was much extraneous information. In addition, there were several tweets that were in a different language. Though the sentiment analysis tool employed is able to understand and process text in different languages [10], there were several tweets in the dataset that were outside the compatible languages. This resulted in data that does not necessarily represent the reviews of the films that we were intending to compile.

Additionally, there were limited or zero tweets under the hashtags that had 'review' in the name (for example: '#IronManReview). Because of this, the methodology was forced to pivot and create a new dataset that had more to do with the task at hand, which was evaluating the sentiment of movie reviews. For this reason, IMDB was the natural target. After following a tutorial that demonstrated how to get a list of the reviews of the top 250 movies of all time on IMDB, a simple modification was implemented to get all of the audience reviews for each of the Marvel Cinematic Universe films [1]. By gathering the URLs for each film, it was then possible to allow the scraper program to gather each of the reviews from the webpage. This was done manually. It is essential to note that the functioning of this code largely depends on the host's connection to the internet and the ability to run a Google Chrome driver [1]. Modifications must be made if the file is being run in Google Colab.

The final piece of the project was the labeled data. Because there were no previously existing labeled datasets for these reviews, the task of evaluating the polarity of the texts was left to the research leader. All of the 124 reviews for the movie 'Eternals' were chosen for labeling. By reading each review, it was then determined whether each one was positive, neutral, or negative. Because there is such a great difference in individual perception of sentiment, both the VADER and TextBlob standardized sentiment tagging guidelines were the ones that were followed [4][10]. This ensured that there would be a limited amount of individual bias in the labeling of the data.



| | Review | Polarity |
|---|---|---|
| 0 | I've read several critical reviews applauding ... | Neg |
| 1 | Shang-Chi gave me hope for Phase 4 of Marvel. ... | Neg |
| 2 | Another year, another Marvel Studios superhero... | Pos |
| 3 | The Eternals are immortals created to eliminat... | Neu |
| 4 | I'm not going to give a long winded review tho... | Neg |
| ... | ... | ... |

Figure 1: First Five Examples of Labeled Data

In both packages, a neutral statement does not convey negativity or positivity. When a body of text is truly neutral, the author did not have any strongly positive or strongly negative feelings about the subject or had matching positive and negative emotions about the subject. This was all kept in mind when creating the labels for the review data. Each review was marked with either 'Pos', 'Neu', or 'Neg', all representing the positive, neutral, and negative overall sentiment classifications, respectively. This was then put into an Excel file named 'textEternals.csv'.

### 3.3 The Function

To execute the tasks described above, it was necessary to create a function to avoid complications with long, repetitive code. The function 'sentiment' was created to consolidate all of the information and instructions to a few lines of code, as there were over 30 movies that needed to be analyzed. The function took in the URL and keyword, as this was a modified version of the original code and necessitated the use of a keyword to proceed. The url was found by looking up the name of the movie on IMDB, then clicking on user reviews. The keyword was simply an identifying sequence of letters that corresponded to the movie name, usually condensed if the title was lengthy.

The function returned several different items. First, the full text of each review was written to a list variable named textIMDB. Next, lists of negative, neutral, and positive reviews were returned to lists named `negative_list`, `neutral_list`, and `positive_list`, respectively. Then, the final list named `comp_list` returned all of the compound scores from each of the text reviews. Finally, the list of polarity labels were assigned to a variable named polarityList. From there, a function called NormalizeData was used to normalize the compound scores on a scale from 0 to 1 that is more easily comparable to the IMDB scores given in a percentage. The mean of these scores was calculated and then printed. In addition, the score was added to a list named scoreAlgorithm that was used for comparison at a later point in the analysis.

The function also printed several different items to aid in understanding. First, the name of the film was printed as well as the name of the website title. As each review was processed, a bar showing progress and time elapsed was displayed on the print screen. After this, all variables from the first review were given: review title, author, review date, review text, and helpfulness count. All variables besides the text were ignored, but this was mostly done to see that the program was working as expected. Then, a second progress bar was printed to the screen, evaluating the text reviews that were not NoneType. Then, a pie chart was printed to the screen, showing the percentage of reviews labeled positive, neutral, and negative by the sentiment classifier. Finally, the aforementioned overall compound score was printed. This process was repeated for all films.

## 4 Goals

For this project, there were several different goals at each of the stages of the study. The first was to identify the relevant text reviews that would be able to represent the target group of Marvel

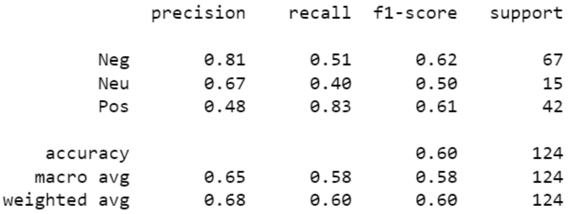
Cinematic Universe viewers. In the case of the modified study, the relevant information existed on each IMDB user review webpage that related to the movie. This method was very organized and resulted in a limited amount of extraneous data, as people are expected to write a review about the movie on this platform. The second goal was to be able to perform the sentiment analysis of the different text reviews without running into too many hiccups involving the misunderstanding of abstract terms, slang, or specific characters. Another goal was to make a comparison between the average sentiment and the overall rating of the movie using audience scores on rotten tomatoes. From this, it would be interesting to see whether audience members use a certain level of sentiment in their reviews that matches that of the numerical review. Lastly, the final goal was to achieve an accuracy that was greater than or at least equal to the average accuracy of the VADER sentiment package found in other experiments.

## 5 Deliverables and Metrics for Success

The difference between an overall rating of a particular film using my sentiment analysis project will be compared with different audience ratings on IMDB. Though sentiment is generally given in terms from -1 to +1, those values will be adjusted in order to match that of IMDB, which uses a percentage score. Therefore, a -1 will be a 0 percent, whereas a +1 will be a 100 percent. The average rating out of 100 will be compared with the true IMDB score on a 100 point scale. If sentiment is scaled on a three-point scale (negative, neutral, and positive), it is anticipated for the score to be within the same category as the audience score. As a supplement, the average difference between the sentiment of the review test data and the numerical movie rating will be tested.

To determine the efficacy of the program, several metrics were chosen. To get the accuracy, precision, recall, and F1 score of the sentiment analysis program, it was necessary to compare the labeled data with the results from the sentiment classification. Accuracy pertains to the total percentage of correct guesses of the sentiment made by the sentiment analyzer. Therefore, if there were 400 correct predictions out of a total of 500, the accuracy would be 80 percent. Recall is the percentage of correctly predicted responses out of all of the actual number of responses of that label in the pre-classified data. This means that if the program detected 10 positive labels and there were actually 20 positive labels in the pre-classified data, the Recall would be 50 percent. An F1 score combines the precision and recall scores of the model, which is particularly useful when there is an uneven class distribution in the data because it focuses on performance based on class. By using sklearn's metrics such as `accuracy_score` and `classification_report`, all of these model evaluation metrics were able to be determined.

## 6 Performance Metrics Results



```
              precision    recall  f1-score   support

         Neg       0.81      0.51      0.62        67
         Neu       0.67      0.40      0.50        15
         Pos       0.48      0.83      0.61        42

    accuracy                           0.60       124
   macro avg       0.65      0.58      0.58       124
weighted avg       0.68      0.60      0.60       124
```

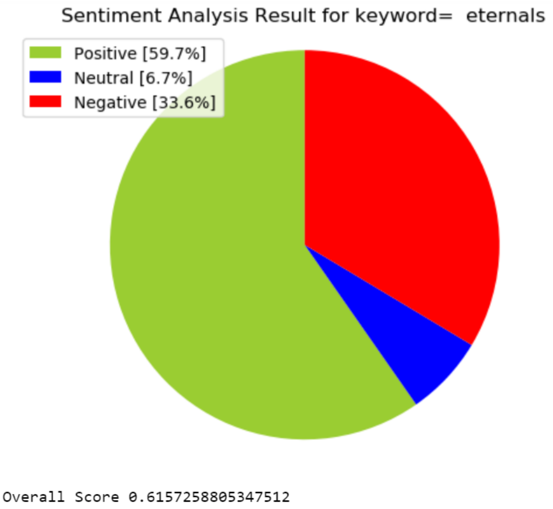
Figure 2: Sklearn's Classification Report Results.

Ultimately it was found that for each classification possibility (Neg, Neu, and Pos), there were differing measures of all metrics, some varying very little and some varying greatly. First, for the precision metric, the highest precision was found to be for the Negative case at 0.81. This means that this model was able to identify a high percentage of true Negative instances while minimizing the false Negative instances. Shortly following was the Neutral case at 0.67, and then finally the Positive case at 0.48. It seems that the classification algorithm was not very successful at detecting false positive instances and tended to misclassify other instances as positive.

Next, the Recall measure for each case was determined: 0.83 for Positive, 0.51 for Negative, and 0.40 for Neutral. A high recall value such as 0.83 indicates that the problem with the Positive case classification was in fact the misclassification of other instances as positive, though it did have a high percentage of detection of true positive cases. This also means that there is a low rate of false negatives for the Positive case. For the Negative case, the sentiment analyzer performed moderately well. The sentiment analyzer model seemed to struggle the most with the Neutral case, not being able to correctly identify most of the review texts. Recall proves to be important in cases such as cancer identification, where minimizing the false negatives can save someone's life and get them into treatment sooner. However, in this problem, recall may not be the most important metric, as companies may not be interested in the audience members that do not have a strong opinion about their product. It is much more important for them to correctly identify opinions that are very positive or very negative. So although this algorithm didn't perform well for the Neutral case, the results aren't the most important in the context of the problem at hand.

The accuracy measure for all of the data was determined to be about 0.60, meaning that about 60 percent of the text reviews were correctly labeled by the sentiment analysis model. Considering that some have recorded both TextBlob and VADER as having an accuracy of about 56 percent for social media data, 60 percent is a bit of an improvement. However, other studies have shown accuracies up to around 75 to 85 percent. Though this is not ideal, there are many factors that go into the accuracy, such as the medium and writing style of the author. There is much nuance that goes into making an accurate model, so having an accuracy that is fairly similar is an accomplishment in itself.

Finally, the last metric was the F1 score, which accounts for precision and recall. The values for these scores were 0.62 for the Negative class, 0.61 for the Positive class, and 0.50 for the Neutral class. As seen in the support category, there were 67 Negative reviews, 15 Neutral reviews, and 42 Positive Reviews. Since the class distribution seems to be unevenly distributed, with the Negative and Positive cases dominating the dataset, an F1 score may be a more appropriate measurement of accuracy than the standard accuracy calculation. However, there does not seem to be a major difference between the accuracy measure and the F1 scores for Negative and Positive classes. There does seem to be a lower F1 score for the Neutral class, perhaps with taking the limited support count into account.

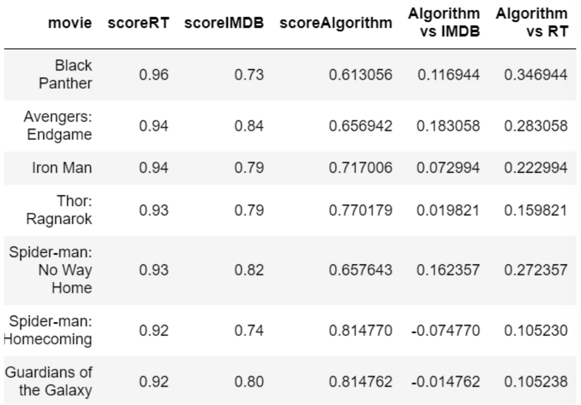


```
Overall Score 0.6157258805347512
```

Figure 3: Pie Chart of Classification Results and Reporting of Overall Compound Score.

## 6.1 Score Comparison

To provide the IMDB Scores of each of the movies, the names of all of the Marvel Cinematic Universe films were written down in string form and the scores were converted from a ten point scale to a one hundred point scale. These values were put into their own respective lists and later columns in a data frame named 'movie' and scoreIMDB. Each score was included in the same exact order to ensure that the correct values were being compared. In addition, out of curiosity, it was decided that a list for the RottenTomatoes scores would also be created [11]. These were already represented as a percentage, so no changes were made besides the new representation of the score in decimal form. These values comprise the column named scoreRT. Finally, the algorithm compound scores from scoreAlgorithm were added into the 'scoreAlgorithm' column.



| movie | scoreRT | scoreIMDB | scoreAlgorithm | Algorithm vs IMDB | Algorithm vs RT |
|---|---|---|---|---|---|
| Black Panther | 0.96 | 0.73 | 0.613056 | 0.116944 | 0.346944 |
| Avengers: Endgame | 0.94 | 0.84 | 0.656942 | 0.183058 | 0.283058 |
| Iron Man | 0.94 | 0.79 | 0.717006 | 0.072994 | 0.222994 |
| Thor: Ragnarok | 0.93 | 0.79 | 0.770179 | 0.019821 | 0.159821 |
| Spider-man: No Way Home | 0.93 | 0.82 | 0.657643 | 0.162357 | 0.272357 |
| Spider-man: Homecoming | 0.92 | 0.74 | 0.814770 | -0.074770 | 0.105230 |
| Guardians of the Galaxy | 0.92 | 0.80 | 0.814762 | -0.014762 | 0.105238 |

Figure 4: Each Film's Score on a scale of 0 to 100 (First Seven out of Thirty-One)

### 6.2 Similarity to IMDB and RottenTomatoes Scores

To determine the difference between each of the different scores, two new columns were made, named 'Algorithm vs IMDB' and 'Algorithm vs RT'. For each row, the established score given by the two websites was subtracted by the value of the score found with the sentiment analysis program. Subsequently, the values in each column were averaged to get an overall mean difference between the algorithm's score and two popular review websites' scores for the ratings of the films. In the end, it was found that the average difference between the RottenTomatoes scores and the sentiment algorithm scores was about 10.5 percentage points. In contrast, the average difference between the IMDB scores and the sentiment algorithm scores was about 1.9 percentage points. This means that the RottenTomatoes scores were about 10.5 percentage points higher than the average compound score found with this study's algorithm on average. For the IMDB scores, the sentiment score was only about 1.9 percentage points higher than that of the calculated compound sentiment score. This indicates that the algorithm gives sentiment that is most similar to that of the reviews on IMDB.

## 7 Discussion

With this study, the over-encompassing mission was to see the difference between the overall numerical rating of a film and the sentiment of that in the text reviews. By choosing a franchise with a large selection of movies, all with different ratings, it was thought that this may prove that the algorithm could handle the analyses all along the spectrum of very negative sentiment all the way to very positive sentiment. In the end, it was found that the algorithm performed better than average and was able to produce results that most related to the IMDB scores rather than the RottenTomatoes scores.

Several inferences were made about the performance of the sentiment analyzer, but what would this mean in a non-academic scenario? By looking into this problem, information was revealed about the authenticity of the reviews and the comparison between websites. Upon further digging and thought, there can be conclusions made about the consumers of both the websites and the contrast in the standards of each website. These findings could be used by a company to develop a better understanding of their market and the accuracy of the reviews posted on their website. This can indirectly contribute to the success of a company, as the market is what drives corporate production.

When making a review on IMDB, you are required to give a numerical review, but not necessarily a textual review. Because the text reviews were sourced from IMDB and the estimated scores of the sentiment were most similar to that of the IMDB numerical scores, it can be assumed that the reviewers on IMDB likely give textual reviews that are very similar to that of the numerical value that they also give the movie. This is important, as these two aspects don't necessarily have to match up. This may also indicate that IMDB is more resistant to a concept known as 'review bombing', which is an occurrence when a group of people collectively decide to give as many negative reviews as possible to a specific movie due to a personal grievance with one of the creators rather than a genuine opinion of the product. This highly skews the data towards a negative review, and is relatively common in popular franchises such as Marvel due to Marvel's occasional socio-political messages during the films. To ensure the lack of interference of conflicting variables, more studies would need to be performed. However, it is interesting to note this as a topic for further investigation.

While it was found that there was a small difference between the IMDB scores and the IMDB text sentiment, the difference between the RottenTomatoes scores and the IMDB text sentiment was much greater. This difference indicated that the RottenTomatoes scores were much higher than the IMDB scores. Why might this be the case? Though there are several potential reasonable candidates, some are business-oriented and others are consumer-oriented. It is possible that more positive reviewers like to post their opinions on RottenTomatoes or perhaps the RottenTomatoes environment is more conducive to positive comments. It is also possible that RottenTomatoes Reviewers genuinely enjoy the movies more. Nevertheless, the reason could be for capital gain as well. Though the website focuses on reviews, it does have a tab named 'Showtimes' that, when clicked on, leads the user to a webpage (Fandango) that sells movie tickets. It is conceivable that the RottenTomatoes could promote an environment that results in more positive comments in order to sell more tickets to movies because people are less likely to go to a movie that has bad reviews. It is important to note that though the review is off by more than 10 percent, this does not necessarily push the movie into a different sentiment category. Again, this subject would need to be investigated further in order to imply causation. For whatever reason the relationship exists, this study has provided a few more details about these two websites.

## 8 Future Improvements and Topics

Looking back on the performance of the model, the sentiment analysis seemed to perform at an average level of accuracy. But there were several conflicting variables that could have influenced the result. As mentioned previously, the labeled testing data was created out of necessity, but perhaps the guidelines were too ambiguous to give accurate labeling results. It did seem that there was a low representation of neutral cases in the dataset, which could be attributed to a difference in personal classification of what is 'neutral'. There is also the possibility that there simply were not many neutral cases that existed in the dataset. In future studies or replications, it would be a good idea to find a labeled dataset that has been reviewed and is considered standardized. This consistency would lead to more accurate results and a better representation of the intended subject.

The sentiment analyzer seems to struggle with certain types of phrases that deal with sarcasm, tone, context, irony, and figurative language in general. Although this is a general problem with NLP algorithms, it does still affect the studies associated with the task. Upon looking at individual reviews and some of the labels that were given to them, it was found that the sentiment analyzer did not understand when reviewers were being sarcastic or when they used humor to make fun of the film. In the future, it may be pertinent to find an updated algorithm that is suited to handle these types of responses better than the tools that are currently utilized.

One of the major issues with the study was the fact that both websites (IMDB and RottenTomatoes) allow users to leave a numerical review without creating a text response to accompany it. In essence, not everyone who does a review leaves a comment. By allowing this, the people in the text portion of the review may not be an accurate representation of all of those that have an opinion about the film. Not having a dataset that accurately represents the population of the individuals that are being studied can result in inaccurate or skewed data. For instance, the people who leave text reviews may be mostly people who have a strong opinion in either direction and will drown out those with neutral opinions. To fix this, it may be a better idea to just represent those that are writing reviews and exclude the scores that do not have text reviews. Additionally, in order to better represent the neutral class, it may be better to use a stratified sampling method that ensures that each subgroup is equally represented.

With a follow-up study, many aspects of the experiment that were not explored could be investigated. It may be fascinating to see how the reviews change over time. Did most of the positive reviews happen right after the movie came out, or did people leave more positive reviews after they were able to watch it another time and truly appreciate it? Furthermore, learning about when emphatic reviews happen may make it easier to create models that are able to detect spam reviews such as the 'review bombings'. With all of this information, better and more accurate conclusions about the audience, the websites, and the movies will be able to be made.

## 9 Conclusion

When embarking on a research-oriented journey, one aims to create a plan that is thorough and well-structured. This way, it is unlikely that mistakes will be made. But sometimes the best discoveries and ideas are generated from obstacles that were in the way. The goal in the beginning of the study was to learn more about twitter sentiment analysis and by the end had morphed into an completely different project out of necessity. This led to the use of community, open-source tools that greatly facilitated the learning process.

Through the combination of two different tools (VADER and TextBlob), the text was successfully analyzed. Though this project resulted in a strikingly average performance of about 60 percent accuracy, there were several valuable key points that were gathered. Firstly, IMDB textual reviews share about the same sentiment as all of the numerical reviews from IMDB with or without a text response included. Secondly, the numerical, non-textual RottenTomatoes' reviews for Marvel Cinematic Universe movies are generally higher on a 100 point scale than the IMDB reviews. Thirdly, the IMDB textual reviews have a greater average difference in sentiment with the RottenTomatoes numerical reviews than the IMDB numerical reviews. More research will need to be done to substantiate generalizations about films outside of the Marvel Cinematic Universe.

Though sentiment analysis may be simple for most human brains to comprehend, today's computing machines seem to struggle with the understanding of a constantly shifting and changing medium of communication. Language is such a complicated subject, but the models for it continue to improve to become more representative of the English language (among others) as a whole. Whether the sentiment towards social media is negative, neutral, or positive, one cannot deny that it has transformed the field of natural language processing by creating a constant influx of input data for researchers to use in order to improve language models. This study certainly would not have existed without the help of social media platforms providing the essential data needed to perform the sentiment analysis.

## References

[1] Aakash, M. (2022). *Scraping IMDB Reviews in Python Using Selenium*, Analytics Vidhya, https://towardsdatascience.com/social-media-sentiment-analysis-in-python-with-vader-no-training-required-4bc6a21e87b8

[2] Annlyel, J. (2020). *The Marvel Cinematic Universe Movies, Ranked by Imdb*, Annlyel Online, (2023). https://annlyelonline.com/2020/11/02/the-marvel-cinematic-universe-movies-ranked-by-imdb

[3] Balahur, A., Hermida, J. M., & Montoyo, A. (2012) *Detecting implicit expressions of emotion in text: A comparative analysis*, Decision Support Systems, 53(4), 742–753. https://publications.aston.ac.uk/id/eprint/37693/1/Orizu_O._2018.pdf

[4] Bandgar, Swapnil. (2021). *Sentiment Analysis Using TextBlob.*, Medium, Analytics Vidhya, (2021). https://medium.com/analytics-vidhya/sentiment-analysis-using-textblob-ecaaf0373dff

[5] Cambria, E., Schuller, B., Xia, Y., & Havasi, C. (2015). *New Avenues in Opinion Mining and Sentiment Analysis*, IEEE Intell. Syst. 28(2), 15-21 https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6468032

[6] GrabNGoInfo, Amy. (2023). *Textblob vs. Vader for Sentiment Analysis Using Python.*, Medium, Towards AI https://pub.towardsai.net/textblob-vs-vader-for-sentiment-analysis-using-python-76883d40f9ae

[7] MonkeyLearn. (2023). *Sentiment Analysis Guide*, MonkeyLearn, (2023). https://monkeylearn.com/sentiment-analysis

[8] B. Pang, L. Lee, and S. Vaithyanathan. (2002). *Thumbs up?: sentiment classification using machine learning techniques*, Proc. of Conf. Empir. Methods Nat. Lang. Process., pp. 79-86. https://www.cs.cornell.edu/home/llee/papers/sentiment.pdf

[9] Schwartz, H. A., & Ungar, L. H. (2015). *Data-Driven Content Analysis of Social Media A Systematic Overview of Automated Methods*, The ANNALS of the American Academy of Political and Social Science, 659(1), 78-94. https://journals.sagepub.com/doi/10.1177/0002716215569197

[10] Vadersentiment, (2020). *Vadersentiment*, PyPI, (2020). https://pypi.org/project/vaderSentiment

[11] Alex Vo. (2023). *All 31 Marvel Movies Ranked: See MCU Movies by Tomatometer*, Rotten Tomatoes Movie and TV News All 31 Marvel Movies Ranked See MCU Movies By Tomatometer Comments https://editorial.rottentomatoes.com/guide/all-marvel-cinematic-universe-movies-ranked

[12] Zoumana, K. (2022). *Social Media Sentiment Analysis in Python with Vader - No Training Required!* Medium, Towards Data Science, (2022). https://towardsdatascience.com/social-media-sentiment-analysis-in-python-with-vader-no-training-required-4bc6a21e87b8.