Sentiment Analysis and Collocation extraction report

By Gareth Madoc-Jones ID: 25646648

Edge Hill University

CIS2705 Data Analytics 2024

Table of Contents

[**Table of Figures** 2](#_Toc196731456)

[**Introduction** 3](#_Toc196731457)

[**Methodology** 3](#_Toc196731458)

[**Sentiment Analysis** 3](#_Toc196731459)

[**Strengths of Method** 5](#_Toc196731460)

[**Drawbacks of Method** 5](#_Toc196731461)

[**Collocation Extraction Method** 6](#_Toc196731462)

[**Strengths of Method** 7](#_Toc196731463)

[**Drawbacks of Method** 7](#_Toc196731464)

[**Collocation Extraction with Pos-Tagging** 7](#_Toc196731465)

[**Strengths of Method** 8](#_Toc196731466)

[**Drawbacks of Method** 8](#_Toc196731467)

[**Results** 8](#_Toc196731468)

[**Visualisation of Results** 9](#_Toc196731469)

[**Bar Chart** 9](#_Toc196731470)

[**Scatter Graph** 9](#_Toc196731471)

[**Pie Chart** 10](#_Toc196731472)

[**Errors and Mistakes in Sentiment Analysis** 11](#_Toc196731473)

[**Collocation Extraction Results** 13](#_Toc196731474)

[**Without POS-Tagging** 13](#_Toc196731475)

[**Positive Reviews Filtered** 13](#_Toc196731476)

[**Positive Reviews Unfiltered** 14](#_Toc196731477)

[**Negative Reviews Filtered** 15](#_Toc196731478)

[**Negative Reviews Unfiltered** 16](#_Toc196731479)

[**With POS Tagging** 17](#_Toc196731480)

[**Positive Reviews** 17](#_Toc196731481)

[**Positive Reviews with NN Pos Tagging** 18](#_Toc196731482)

[**Negative Reviews with Pos Tagging** 19](#_Toc196731483)

[**Negative Reviews with NN Pos Tagging** 20](#_Toc196731484)

[**Conclusion** 20](#_Toc196731485)

[**References** 21](#_Toc196731486)

[**Appendix** 22](#_Toc196731487)

[**Student Contribution Table** 26](#_Toc196731488)

# **Table of Figures**

[Figure 1- Imported Libraries 2](#_Toc196666497)

[Figure 2- Chunk Creation for Sentiment Analysis 3](#_Toc196666498)

[Figure 3 - Parameters to aid Data Frame calling 3](#_Toc196666499)

[Figure 4- Column Assigning and List Creation 3](#_Toc196666500)

[Figure 5- Final Data Frame Creation for Gathering Results 3](#_Toc196666501)

[Figure 6- Application of Sentiment Analysis and Label Generation 4](#_Toc196666502)

[Figure 7- Functions for Sentiment Analysis and Label Creation 4](#_Toc196666503)

[Figure 8 Creation of Lists to Separate Positive and Negative Reviews. 5](#_Toc196666504)

[Figure 9 Non-POS Tag Collocation Extraction. 5](#_Toc196666505)

[Figure 10 POS Tag Collocation Extraction 6](#_Toc196666506)

[Figure 11 Sentiment Analysis Results 7](#_Toc196666507)

[Figure 12 Bar Chart Code 8](#_Toc196666508)

[Figure 13 Bar Chart Results 8](#_Toc196666509)

[Figure 14 Scatter Graph Code 9](#_Toc196666510)

[Figure 15 Scatter Graph Results 9](#_Toc196666511)

[Figure 16 Pie Chart Code 9](#_Toc196666512)

[Figure 17 Pie Chart Results 10](#_Toc196666513)

[Figure 18 Error in Sentiment Analysis 1 11](#_Toc196666514)

[Figure 19 Error in Sentiment Analysis 2 11](#_Toc196666515)

[Figure 20 Error in Sentiment Analysis 3 11](#_Toc196666516)

[Figure 21 Positive Filtered Collocation Extraction Results 1 12](#_Toc196666517)

[Figure 22 Positive Filtered Collocation Extraction Results 2 12](#_Toc196666518)

[Figure 23 Positive Unfiltered Collocation Extraction Results 1 13](#_Toc196666519)

[Figure 24 Positive Unfiltered Collocation Extraction Results 2 13](#_Toc196666520)

[Figure 25 Negative Filtered Collocation Extraction Results 1 14](#_Toc196666521)

[Figure 26 Negative Filtered Collocation Extraction Results 2 14](#_Toc196666522)

[Figure 27 Negative Unfiltered Collocation Extraction Results 1 15](#_Toc196666523)

[Figure 28 Negative Unfiltered Collocation Extraction Results 2 15](#_Toc196666524)

[Figure 29 Positive POS Tagged Collocation Extraction Results 1 16](#_Toc196666525)

[Figure 30 Positive POS Tagged Collocation Extraction Results 2 16](#_Toc196666526)

[Figure 31 Positive NN POS Tagged Collocation Extraction Results 1 17](#_Toc196666527)

[Figure 32 Positive NN POS Tagged Collocation Extraction Results 2 17](#_Toc196666528)

[Figure 33 Negative POS Tagged Collocation Extraction Results 1 18](#_Toc196666529)

[Figure 34 Negative POS Tagged Collocation Extraction Results 2 18](#_Toc196666530)

[Figure 35 Negative NN POS Tagged Collocation Extraction Results 1 19](#_Toc196666531)

[Figure 36 Negative NN POS Tagged Collocation Extraction Results 2 19](#_Toc196666532)

# **Introduction**

For this project a dataset of 10,000 movie reviews was analysed using Sentiment Analysis. Arumugam et al (2018) suggests that Sentiment Analysis is then process of deriving values from text that can be obtained through different data sources such as social networks, product reviews and news articles. For this project, the Movie Reviews dataset will be analysed to conclude which reviews are positive, negative and neutral. The Sentiment Value needs to be given a label based on its value. If the sentiment value is above 0 then the review is positive, equal to 0 and the review is neutral and below zero and the review is negative. Once the Sentiment Analysis is complete using collocation extraction the 40 most important collocations from both the positive and negative reviews will be analysed. They will be analysed with POS tagging and without POS tagging. Cambridge Dictionary (2025) argues that a collocation is ‘the combination of words formed when two or more words are often used together in a way that sounds correct’. This shows that pairs of words in the dataset will be gathered and analysed. To further expand on this Lulu, Yue and Pengyuan (2021) suggest that collocation extraction refers to the automatic extraction of collocations through the computing power of a computer. This as mentioned above will be analysed with and without POS tagging. In accordance with Geeks for Geeks (2024) Part-Of-Speech tagging is where each word in a document is given a particular part of speech such as adverbs, adjectives, verb or as part of the grammar. With POS tagging the results of the Collocation Extraction can be narrowed down to make them more accurate.

# **Methodology**

As mentioned above, the process of this dataset being analysed is split into a sentiment analysis and then a collocation extraction. The sentiment analysis was the first way the dataset was analysed.

# **Sentiment Analysis**

For this method to work multiple libraries were required to be downloaded and/or imported. These libraries are all imported to aid in the Sentiment Analysis and Collocation Extraction Methods.

A screen shot of a computer code

AI-generated content may be incorrect.

Figure - Imported Libraries

A computer screen shot of a program code

AI-generated content may be incorrect.For the Dataset to be analysed due to the large size of the dataset the data was split into chunks to allow for efficient processing of the reviews in a for loop. Medium (2024) suggests that the way to do this is through splitting the data into chunks, allowing part of the data to be loaded at a time preventing Python from running out of RAM and stopping the program. For this, chunks of 1000 reviews were loaded at a time to prevent any Memory Issues and make the loading of the dataset more efficient

Figure - Chunk Creation for Sentiment Analysis

A computer screen shot of a program code

AI-generated content may be incorrect. While the data is being loaded in chunks there are Parameters in place to help the loading of the data. Geeks for Geeks (2021) suggests that to read a file without a header the code header = None needs to be used to show the interpreter that there is no header row in the file. There is also the case that the contents of the file are separated by tab spaces, so the code delimiter = ‘\t’ is needed to show the interpreter (Geeks for Geeks, 2024). There also needs to be factors in place preventing lines that are incomplete from being loaded in the chunk and potentially stopping the program from running. In accordance with Saturncloud (2023) to do this the code needed is the line on\_bad\_lines = ‘skip’, this tells the interpreter to skip any lines that do not contain the required data.

Figure - Parameters to aid Data Frame calling

A computer screen shot of a program code

AI-generated content may be incorrect.

Figure - Column Assigning and List Creation

A computer screen shot of a program code

AI-generated content may be incorrect.Once this is done the chunk is assigned columns to help apply the sentiment analysis to only the review text. This is done using the .columns code to manually give the chunk columns. Once the columns are identified there is another function applied. The sentiment\_value is applied to the ReviewText column to gather a sentiment value using the get\_sentiment function. This function uses the Text Blob libraries to analyse the text. Once the value is gathered it is run through the sentiment\_label function to assign it a Boolean label of Positive if the value is above 0, neutral if the value is 0 and negative if the value is less than 0. The Sentiment Values, Labels and Review Texts are added to preestablished lists, so that they can be added to a final dataframe.

When this is finished, the lists are added to a final data frame called final\_df. This Data Frame is used to show the review texts, with there sentiment values and labels.

Figure - Final Data Frame Creation for Gathering Results

This Data Frame is printed to show the overall number of times that certain outcomes occur. This is through the Value counts method. SaturnCloud (2023) argues that the value\_counts () method is a convenient way to count the number of times a unique value in a DataFrame column occurs. By doing this it is easy to distinguish the amount of positive, negative and neutral reviews there are.

A computer screen shot of text

AI-generated content may be incorrect.A computer screen shot of a program code

AI-generated content may be incorrect.During the processing of the reviews in each chunk the functions get\_sentiment(text) and sentiment\_label(polarity) are called to run the Sentiment Analysis of each review and give each review the corresponding label. These two functions work together as the get\_sentiment function gets the sentiment value using Text Blob, and the sentiment\_label function assigns a Boolean value to the sentiment value by calling the gathered Sentiment Value from the get\_sentiment function and running it through an if else statement checking which Boolean criteria it meets.

Figure - Application of Sentiment Analysis and Label Generation

Figure - Functions for Sentiment Analysis and Label Creation

## **Strengths of Method**

This method has many strengths to it. Firstly, by applying the Sentiment Analysis Method into a for loop and chunking the data so it runs per 1000 reviews, the speed and efficiency of the code is improved allowing for a quicker Sentiment Analysis of the reviews. Also, the parameters in place preventing any errors such as on\_bad\_lines = ‘skip’ allows the method to be run smoothly and have a more robust feel. Having the first 5 reviews of each chunk being show can act as checkpoints showing how much of the sentiment analysis has been performed.

Having the value counts be performed at the end allows for an easy way of showing the overall results of the sentiment analysis. As well as this the creation of a new data frame to store the results of the sentiment analysis including the review test, sentiment value and label ensures that the original data file is kept original since it is needed every time the code is executed.

## **Drawbacks of Method**

However, there are some drawbacks to this method. Firstly, the method is unable to pickup on Sarcastic tones in user reviews. This could be reviews that use positive words in a negative way. An example of this would be “I thought this movie would be great, but I was clearly wrong”. The method would pick up on the positive word “great” but not pick up on the clear tone of the sentence which is that the film was bad. This will be explored more deeply in the Results section along with explanations as to why this can occur.

Another Drawback of this Method is a weakness of the Text Blob Sentiment Analysis method itself. Text Blob Sentiment Analysis will skip over any words that it is unfamiliar with. This can be a problem with such a huge dataset, there will be cases where positive/negative words are skipped. Why this is an issue is explained in depth in the Errors and Mistakes section.

# **Collocation Extraction Method**

The Collocation Extraction method used requires functions to be run that call the positive and negative review texts. For this to occur the positive and negative review texts need to be extracted from the review texts list.

This was done by creating new lists that could extract the review texts which contained either a Positive Label or a negative Label.



Figure Creation of Lists to Separate Positive and Negative Reviews.

Once this was done the Collocation Extraction could be undertaken.

For the Collocation Extraction a function was defined with the Positive or Negative review text lists used as a parameter depending on what type of review was being analysed. Having these reviews be in the lists made it easy to define which reviews were to be analysed.

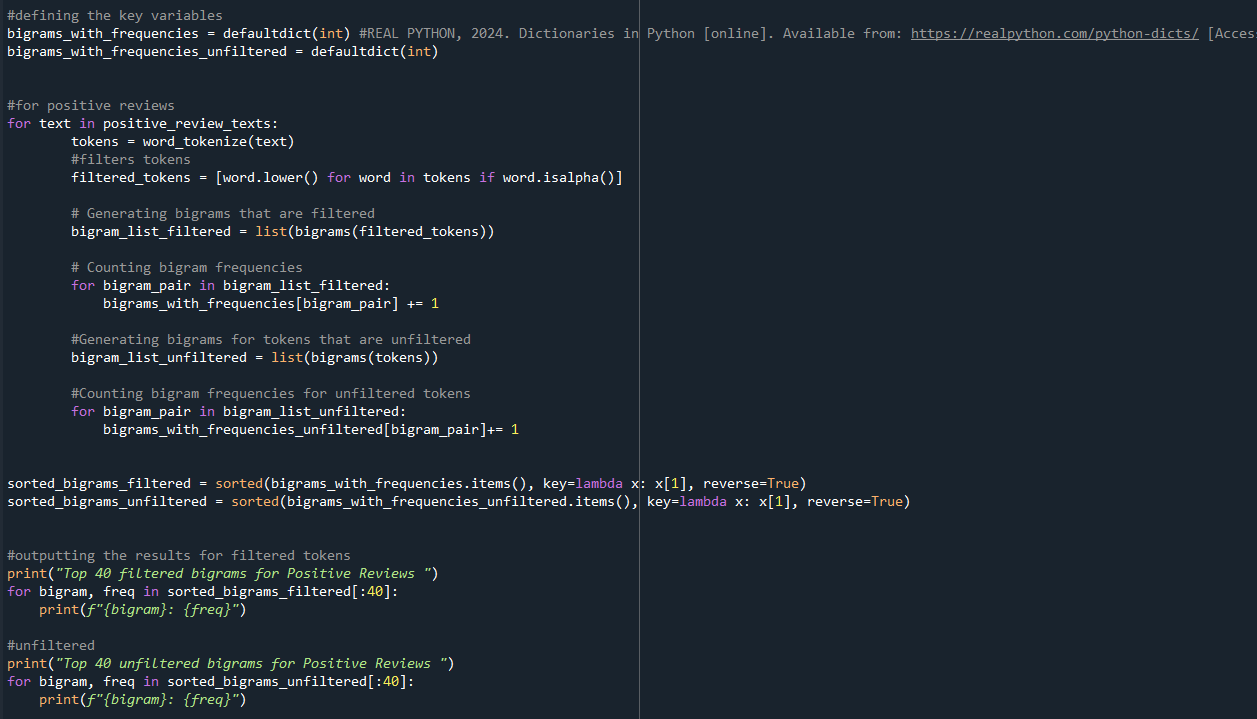


Figure Non-POS Tag Collocation Extraction.

The Collocation Extraction method starts by defining some key values that will be used in the collocation extraction. The collocation extraction is then performed using a for loop. For the collocation extraction, the results are filtered and unfiltered without the use of POS tagging. The filtered results have punctuation and special characters removed to have only words show up. The unfiltered results will have punctuation and special characters in the results.

For the filtered results, results are filtered to show only alphanumeric results which will display only words. For the unfiltered results this is not applied. For both ways the text is tokenised to ensure better results. According to Awan (2024) tokenisation is the process of converting sequences of text into smaller parts aiding machines to understand human language by making it easier to analyse. Once this is done the review texts are split into bigram pairs and the frequency of the pairs are counted. These frequencies are then sorted to show the results in frequency order.

## **Strengths of Method**

This method has many strengths to it. By applying tokenisation to the results, the analysis of the results and makes them more accurate. This ensures that the results from the collocation extraction are useful. Also, having a method which shows results with punctuation and special characters, and having a method that removes these from the results allows for a more in-depth look into the dataset as well as implementing a broader range of results.

## **Drawbacks of Method**

However, there are some drawbacks to this method. This method does not nave stop words removed meaning that the results will have a lot of common words in them such as “the” and “and” meaning that usefulness from the result is limited. Also, this method could have been implemented using a function to make it overall be more robust and prevent any errors in the code.

# **Collocation Extraction with Pos-Tagging**

A screen shot of a computer code

AI-generated content may be incorrect.

Figure POS Tag Collocation Extraction

For POS-Tagging the Collocation Extraction Method also implemented the ability to find POS tagged bigrams. POS tagging adds what type of word the word is to the output. The exact same code structure has been used for both positive and negative reviews with a few adjustments based on the situation. The method starts with bigrams\_with\_frequencies variable being created using the defaultdict(int) dictionary. A python dictionary is used to store data values in pairs (W3 Schools, 2025). The defaultdict behaves almost the same as a regular dictionary, but if it has not got any values inside then it will generate a key preventing errors such as Key Error (Real Python, 2025). In this case if a bigram pair cannot be given a frequency value, then the default value of 1 will be applied.

Once this is done then the reviews are loaded into a for loop that will go through them individually and apply the collocation extraction and the POS tagging. The words are first tokenised putting them into lower case because this helps with the collocation extraction by preventing duplicate values. For example, if in the review sets the word movie was used in the upper-case form Movie and the lower-case form movie, then the system would define them as two separate values when they are in fact the same. Once this is performed the words are pos-tagged using the pos-tag method. The words are then further filtered by ensuring that each word has a corresponding tag ([word, tag] for word, tag in pos\_tagged\_words) and then there is a check to ensure that only words are passed (word. isalpha ()) and that the words do not appear in the stopwords library (and word not in stop words). As explained in the results section stop words are words that have little to no meaning such as “the” and “and”.

As used in the base collocation extraction method the words are added to the bigram\_list variable. There is then an embedded for loop extracting pairs from this list and incrementing the frequency of each pair when they appear. These pairs are then sorted using the sorted () function to appear by highest frequency. Then there is a print statement showing what is happening and a final for loop running through the 40 most frequent bigram pairs.

For the NN POS-tagging the method used is the same for both positive and negative reviews with the addition of the code (tag.startswith('NN')) to tell the system what to look for.

## **Strengths of Method**

This method has many strengths to it. Applying stop words as well as pos-tagging really narrows down what will appear meaning that the outputted results can provide important information about the dataset. Stop Words will remove unnecessary words such as “the” that have no meaning and therefore cannot give a better understanding of the dataset and viewers compliments/issues with the films they have reviewed. Having the POS tagging applied as well helps gather an understanding of the overall structure of the reviews written and focusing on the Nouns with the NN pos-tagging allows for a deeper understanding of the names of films/actors that were hated or loved depending on the review.

## **Drawbacks of Method**

However, there are some drawbacks to this method. Having stop words removed can take some context out of the results. For example, the phrase, “the movie is bad” would be cut down to “movie” and “bad” and it would make little sense grammatically. Also, having POS-tagging be part of the results can make the results far less readable and for some people the results could be completely unreadable. Another drawback of POS tagging is ambiguity. Some words can be nouns and verbs giving confusion to the tagging system as well as any words that POS-Tagging does not understand will be skipped. Both can lead to inaccuracies in the results.

# **Results**

A screen shot of a computer

AI-generated content may be incorrect.**Sentiment Analysis Results**

Figure Sentiment Analysis Results

As can be seen in the screenshot to the left here are the results for the Sentiment Analysis of the dataset. The method has picked out many positive and negative reviews and a small number of neutral reviews. For the Sentiment Analysis labels any positive review has a Sentiment Value greater than 0 and negative values are values less than zero, while neutral labels have a Sentiment Value of 0. There is some margin for these values with the Analysis method having values ranging from -0.005 to 0.005 as neutral.

## **Visualisation of Results**

To better understand the results, they were interpreted graphically. By interpreting the results graphically, a further understanding of them can be gathered and the data can be interpreted easily.

### **Bar Chart**

The first way of visualising these results was a bar chart. Having a Bar chart allowed for a visual representation of the Sentiment Analysis results as well as a comparison of the results. The below code creates the bar chart.

A computer code on a black background

AI-generated content may be incorrect.

Figure Bar Chart Code

A graph with different colored squares

AI-generated content may be incorrect.

Figure Bar Chart Results

This bar chart is the output from the above code. This shows the vast number of positive reviews compared to the smaller amounts of negative reviews and the little number of neutral reviews.

### **Scatter Graph**

Then a scatter graph was created using the below code.

A computer code on a black background

AI-generated content may be incorrect.

Figure Scatter Graph Code

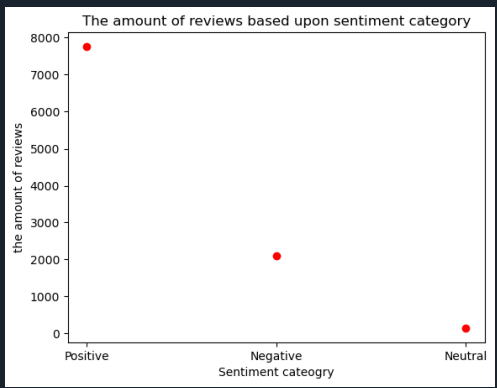


Figure Scatter Graph Results

This is the output from the scatter graph further showing the high amount of positive and negative reviews as well as the tiny number of neutral reviews.

### **Pie Chart**

And a pie chart was created using the below code.

**A blurry image of a computer screen

AI-generated content may be incorrect.**

Figure Pie Chart Code

A pie chart with different colored triangles

AI-generated content may be incorrect.

Figure Pie Chart Results

The pie chart shows the number of reviews that there are per category in a percentage order further developing understanding of the results. This pie chart shows that most reviews were positive, with a small minority negative, and that a tiny number of reviews were neutral showing that reviewers either loved or hated reviews and a very small amount did not love or hate the reviews.

By having these three separate graphs/charts a visual representation of the data can be built showing people who may not understand the results an easy way to interpret them.

## **Errors and Mistakes in Sentiment Analysis**

As with any computer the inability for it to detect Sarcastic comments can lead to errors in labelling. Abiola, O. et al. (2023) suggests that Text Blob will ignore terms it is unfamiliar with and only considers words and expressions to which it can apply extremes and midpoints to help it arrive at the final score. In some reviews there would be cases where the words are not understood by Text Blob so they will be skipped over by the analysis method and could lead to some false positive/negative results. There would also be cases where negative reviews contain positive words and Text Blob would be unable to interpret this.

Some examples of this with explanations as to why this occurred are below: A close up of a text

AI-generated content may be incorrect.

Figure Error in Sentiment Analysis 1

This review was identified to be positive by the Sentiment Analysis Method. It was given a score of 0.1807716049382716 despite it clearly being a negative review when read through. This shows that Text Blob cannot understand the tone of a review. This review started off with some positive words such as “good”, “interesting”, “enjoy”, “intriguing” and “superior”. The sentence “The DVD cover is intriguing, and honestly the cover is better than the film itself”. To a human reading this sentence, this is a negative comment about the movie suggesting that it has not lived up to the potential it had. But a machine only sees the positive words of “intriguing” and “better”. Therefore, while this should lower the sentiment score in fact it raises it giving a false positive review.

A close up of black text

AI-generated content may be incorrect.This movie was given a positive score from the system. The score was 0.13333333333333333. Due to the size of the review there is not many words for Text Blob to go on to give it a score. The only descriptive words mentioned on the review are “good” and “like” which are both positive words. Despite these words being positive they are being used in a negative way. This tricks the interpreter into thinking the review is positive when in fact it is negative. Repustate (2021) argues that tone is a real problem for Sentiment Analysis methods, tone can be a problem to understand verbally, and it is even more difficult in the written word.

Figure Error in Sentiment Analysis 2

This review was labelled negative by the system and given a value of -0.0833333333333333 despite this review being positive. Repustate (2021) suggests that due to the Polarity of Sentiment Analysis methods words like love and hate are high on positive and negative scores in polarity but terms such as not so bad can be left out which affects the sentiment score. In this case the term How can you not buy this set might not be fully understood by the interpreter with only the negative term “not” being taken from the term.

Figure Error in Sentiment Analysis 3

# **Collocation Extraction Results**

# **Without POS-Tagging**

## **Positive Reviews Filtered**

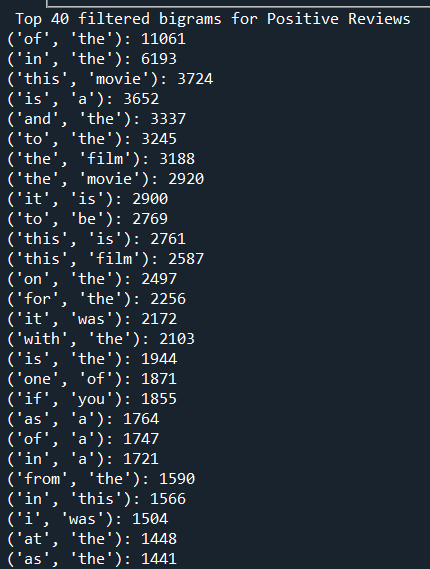
These are the results for the 40 most frequent filtered bigrams in positive reviews. As mentioned above filtered in this result set is that all the results are words. Some of the most frequent results are “of” and “the”, “this” and “movie” and “is” and “a”. These results suggest that people who positively reviewed the movie were very complementary of the movie in their reviews.

Figure Positive Filtered Collocation Extraction Results 1

A screen shot of a computer code

AI-generated content may be incorrect.

Figure Positive Filtered Collocation Extraction Results 2

## **Positive Reviews Unfiltered**

A screen shot of a computer program

AI-generated content may be incorrect.These are the unfiltered collocation extraction results for the positive reviews. These results have punctuation and special characters in them. As can be seen some of the most frequent bigrams are “,” and “,” “’” and “’” and “.” and “the”. These results show that the dataset needs to be cleaned more before analysis since there is an extreme amount of unnecessary punctuation in the dataset which will affect the collocation extraction results if certain measures are not put in place.

Figure Positive Unfiltered Collocation Extraction Results 1

A screen shot of a computer code

AI-generated content may be incorrect.

Figure Positive Unfiltered Collocation Extraction Results 2

## **Negative Reviews Filtered**

**A screen shot of a computer code

AI-generated content may be incorrect.**These are the filtered results for the 40 most frequent collocations in negative reviews in the dataset. As can be seen the results are very similar to the positive filtered results showing that without proper removal of the stop words in the dataset there is a lot of common words used in every review in the dataset even if the reviews are in a different category.

When looking at the results some of the most common bigrams are “in” and “the”, “from” and “the” and “one” and “of”. These results suggest that there could be potential exaggeration of the results such as the bigram one and of could be used in an exaggerated statement such as “one of the words movies ever made”.

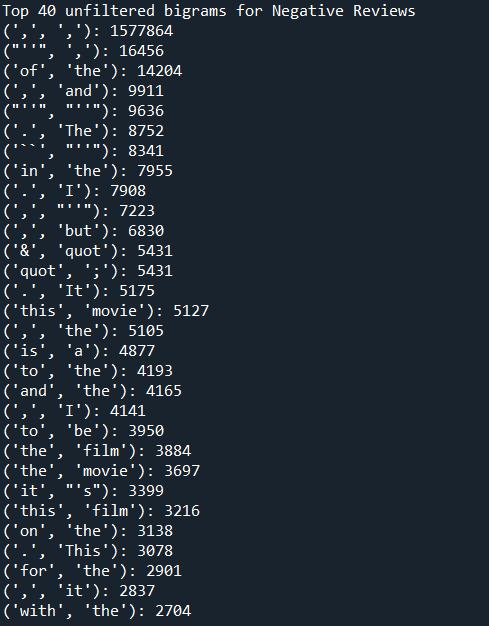
Figure Negative Filtered Collocation Extraction Results 1

**A screen shot of a computer

AI-generated content may be incorrect.**

Figure Negative Filtered Collocation Extraction Results 2

## **Negative Reviews Unfiltered**

****These are the results for the 40 most frequent collocations in the negative review set. As with the positive review results there is a large amount of punctuation showing in the results showing that the dataset needs to be cleaned to have more effective and meaningful results.

Some of the most frequent bigrams are “,” and “but”, “to” and “be” and “of” and “the”. The bigram “,” and “but” could suggest that the negative reviews could have some positive statement about the film but the overall message of the review being negative about the reviewed film.

Figure Negative Unfiltered Collocation Extraction Results 1

**A screen shot of a computer code

AI-generated content may be incorrect.**

Figure Negative Unfiltered Collocation Extraction Results 2

# **With POS Tagging**

## **A screenshot of a computer program AI-generated content may be incorrect.Positive Reviews**

Figure Positive POS Tagged Collocation Extraction Results 1

These are the results of the 40 most frequent bigrams in positive reviews. For this pos-tagging was applied to the bigrams and stop words were removed. Lenovo (2025) argues that stop words are commonly used words such as “the” and “and” that are excluded from text processing tasks, because they do not carry much meaning and occur often in the English Language.

Using POS tagging has many benefits to the analysis of the reviews when compared to the base Collocation Extraction. Swamy and Srinath (2021) suggests that POS tagging allows the overall structure of the sentence can be understood by the computer. This allows for higher level tasks such as parsing and machine translation.

Some of the most frequent bigrams are “Special” and “Effects” and “Mel” and “Gibson”. This would suggest that reviewers are very happy with the Special Effects in the reviewed movies and that reviews that tend to be positive are for movies that feature Mel Gibson. Another commonly featured bigram is “would” and “recommend” suggesting that reviewers would tend to recommend the movie they have viewed to other people potentially influencing people’s decision to buy the movie themselves.

**A computer screen shot of white text

AI-generated content may be incorrect.**

Figure Positive POS Tagged Collocation Extraction Results 2

## **Positive Reviews with NN Pos Tagging**

**A screen shot of a computer program

AI-generated content may be incorrect.**This form of POS tagging focuses on POS-tagging with only Tags that are Nouns (NN). Doing this allows for an increased focus into Nouns and offer a deeper understanding of the review set.

Figure Positive NN POS Tagged Collocation Extraction Results 1

In accordance with Jurafsky and Martin (2019) knowing whether a word is a noun or not tells you about neighbouring words with nouns likely to follow determiners and adjectives, as well as the structure of sentences with nouns being part of noun phrases. Also, having POS tagging focus on words allows for information extraction with named entities such as people or film names being shown clearly (Jurafsky and Martin, 2019).

Some of the most frequent bigrams that show up are “director” and “cut”, “blu” and “ray” and “picture” and quality. This would suggest that reviewers are happy with the director’s cut version of movies, Blu-ray versions of movies and the overall picture quality of the movies.

**A screen shot of text

AI-generated content may be incorrect.**

Figure Positive NN POS Tagged Collocation Extraction Results 2

## **Negative Reviews with Pos Tagging**

**A screen shot of a computer program

AI-generated content may be incorrect.**As with the Positive Reviews with Pos Tag stop words are removed from the list to have more meaningful results.

Figure Negative POS Tagged Collocation Extraction Results 1

Some of the most frequently used bigrams are “special” and “effects”, “waste” and “time” and “one” and “worst”. This would suggest that reviewers do not like the special effects in the movies that they reviewed. The waste and time bigram would suggest that reviewers believe that they wasted their time watching the movie and the one and worst bigram would suggest that reviewers believe that the movie they reviewed was one of the worst movies they have watched. This would significantly impact people from buying the movie themselves since people have had negative experiences watching the movie.

**A screen shot of a computer screen

AI-generated content may be incorrect.**

Figure Negative POS Tagged Collocation Extraction Results 2

## **A screen shot of a computer program AI-generated content may be incorrect.Negative Reviews with NN Pos Tagging**

Figure Negative NN POS Tagged Collocation Extraction Results 1

These are the negative reviews with the Noun (NN) Pos-Tagging applied.

Some of the most frequent bigrams are “time” and “money”, “horror” and “movie” and “video” and “game”. This suggests that reviewers firstly found the movies they watched to be a waste of time and money. These movies could have been horror movies by the frequency of that bigram. They could also be video game adaptations of movies such as the movie Mortal Kombat, that was influenced by the video game series, that frequently shows up as a bigram overall in the negative movie reviews section.

**A screen shot of a computer screen

AI-generated content may be incorrect.**

Figure Negative NN POS Tagged Collocation Extraction Results 2

# **Conclusion**

In this report the results of a sentiment analysis and a collocation extraction with and without pos-tagging of a movie reviews data set have been analysed, showing potential errors and the importance of the multiple collocation extraction methods used. As well as this the methods used to perform this task have been described in detail showing the process of this code being produced as well as the benefits and weaknesses of the methods used. The Sentiment Analysis method used managed to successfully analyse the reviews, with some unfortunate errors and the collocation extraction methods used successfully gathered insight into the dataset.

# **References**

ABIOLA, O, ABAYOMI-ALLI, A, ARGOGUNDADEA, T, O, MISRA, S and ABOYOMI-ALI, O. 2023. Sentiment analysis of COVID-19 tweets from selected hashtags in Nigeria using VADER and Text Blob analyser, *Journal of Electrical Systems and Information Technology* [online]. 10(5). pp-1-20. Available from: <https://jesit.springeropen.com/counter/pdf/10.1186/s43067-023-00070-9.pdf> [Accessed 1 April 2025].

ARUMUGAM, R, SHANMUGAMANI, R, BYIRINGIRO, A, JOSHI, C and MUTHUSWAMY, K,   
2018. *Hands-On Natural Language Processing with Python: A Practical Guide to Applying Deep Learning Architectures to Your NLP Applications* [eBook]. 1st Ed. Birmingham: Packt Publishing Ltd. Available from: <https://ebookcentral.proquest.com/lib/edgehill/reader.action?docID=5456142&ppg=26> [Accessed 25 March 2025].

AWAN, A, A, 2024. *What is Tokenization?* [online]. Available from: <https://www.datacamp.com/blog/what-is-tokenization> [Accessed 27 April 2025].

CAMBRIDGE DICTIONARY, 2025. *Collocation* [online]. Available from: <https://dictionary.cambridge.org/dictionary/english/collocation> [Accessed 25 March 2025].

GEEKS FOR GEEKS, 2021. *How to read csv file with Pandas without header?* [online]. Available from: <https://www.geeksforgeeks.org/how-to-read-csv-file-with-pandas-without-header/> [Accessed 10 March 2025].

GEEKS FOR GEEKS, 2024. *How to convert tab-separated file into a dataframe using Python* [online]. Available from: <https://www.geeksforgeeks.org/how-to-convert-tab-separated-file-into-a-dataframe-using-python/> [Accessed 10 March 2025].

GEEKS FOR GEEKS, 2024. *POS(Parts-Of-Speech) Tagging in NLP* [online]. Available from: <https://www.geeksforgeeks.org/nlp-part-of-speech-default-tagging/> [Accessed 25 March 2025].

JURAFSKY, D and MARTIN, H, J, 2019. *Part-of-Speech Tagging* [online]. Available from: <https://web.stanford.edu/~jurafsky/slp3/old_oct19/8.pdf> [Accessed 10 April 2025].

LENOVO, 2025. *What are stop words?* [online]. Available from: <https://www.lenovo.com/gb/en/glossary/stop-words/?orgRef=https%253A%252F%252Fwww.google.com%252F> [Accessed 7 April 2025].

LULU, G, YUE, P and PENGYUAN, L, 2021. A Comparative Study of Collocation Extraction Methods from the Perspectives of Vocabulary and Grammar A Case Study in the Field of Journalism. In *Proceedings of the 35th Pacific Asia Conference on Language, Information and Computation* [online]. 1(1), pp.201-210. Available from: <https://aclanthology.org/2021.paclic-1.21.pdf> [Accessed 25 March 2025].

MEDIUM, 2024. *Understanding and Fixing Memory Errors in Python* [online]. Available from: <https://medium.com/@ryan_forrester_/understanding-and-fixing-memory-errors-in-python-a86e0abdca58> [Accessed 31 March 2025].

REAL PYTHON, 2025. *Using the Python defaultdict Type for Handling Missing Keys* [online]. Available from: <https://realpython.com/python-defaultdict/> [Accessed 10 April 2025].

REPUSTATE, 2021. *Sentiment Analysis Challenges and How To Overcome Them* [online]. Available from: <https://www.repustate.com/blog/sentiment-analysis-challenges-with-solutions/> [Accessed 1 April 2025].

SATURNCLOUD, 2023. *How to Fix Python Pandas Error Tokenizing Data* [online]. Available from: <https://saturncloud.io/blog/how-to-fix-python-pandas-error-tokenizing-data/#:~:text=For%20example%3A,the%20rest%20of%20the%20file>. [Accessed 10 March 2025].

SATURNCLOUD, 2023. *Python Pandas Convert valuecounts Output to Dataframe [*online]. Available from: <https://saturncloud.io/blog/python-pandas-convert-valuecounts-output-to-dataframe/#:~:text=value_counts()%20method%20is%20a,count%20is%20the%20corresponding%20value.&text=As%20you%20can%20see%2C%20the%20output%20of%20>. [Accessed 10 March 2025].

SWAMY, A and SRINATH, S (2021) POS Tagging and NER System for Kannada Using Conditional Random Fields, *International Journal of Information Retrieval Research (IJIRR)* [online]., 11(4), pp.1-13. Available from: <https://go.gale.com/ps/retrieve.do?tabID=BIGAcademic&resultListType=RESULT_LIST&searchResultsType=MultiTab&retrievalId=06121fae-3370-4f93-95af-8df1d0a07001&hitCount=1&searchType=AdvancedSearchForm&currentPosition=1&docId=GALE%7CA760473569&docType=Article&sort=Pub+Date+Reverse+Chron&contentSegment=GBIB&prodId=GBIB&pageNum=1&contentSet=GALE%7CA760473569&searchId=R1&userGroupName=edge&inPS=true> [Accessed 10 April 2025].

W3 SCHOOLS, 2025. *Python Dictionaries* [online]. Available from: <https://www.w3schools.com/python/python_dictionaries.asp> [Accessed 10 April 2025].

# **Appendix**

**Source Code for Sentiment Analysis**

A screen shot of a computer

AI-generated content may be incorrect.

A screen shot of a computer screen

AI-generated content may be incorrect.

**Bar Chart code**

A computer code on a black background

AI-generated content may be incorrect.

**Scatter Graph code**

A computer code on a black background

AI-generated content may be incorrect.

**Pie Chart Code**

**A blurry image of a computer screen

AI-generated content may be incorrect.**

**Collocation Extraction Source Code**

**Positive without POS Tagging**

**A screen shot of a computer program

AI-generated content may be incorrect.**

**Negative without Pos Tagging**

**A screen shot of a computer program

AI-generated content may be incorrect.**

**Collocation Extraction with Pos Tagging**

**Positive**

**A screen shot of a computer program

AI-generated content may be incorrect.**

**Negative**

**A screen shot of a computer code

AI-generated content may be incorrect.**

**Collocation Extraction with NN Pos Tagging**

**Positive**

**A screen shot of a computer code

AI-generated content may be incorrect.**

**Negative**

A screen shot of a computer code

AI-generated content may be incorrect.

# **Student Contribution Table**

|  |  |
| --- | --- |
| Task | Student Contribution |
| Sentiment Analysis | * Student 25646648 applied Text Blob to review texts to gather positive and negative reviews and classify which is which. * Student 25709054 applied pandas to visually interpret the data in graphs and charts making data more understandable. |
| Collocation Extraction | * Student 25709054 implemented collocation extraction without POS-filtering. * Student 25646648 implemented collocation extraction with POS-filtering. |