**Capstone - Review Sentiment Predictor App**

Julie Walin

College of Information Technology, Western Governors University

Computer Science Capstone

February 2021

**Capstone - Review Sentiment Predictor App**

Table of Contents

[A1. Letter of Transmittal 5](#_Toc64059659)

[A2 – Project Proposal 6](#_Toc64059660)

[Problem Summary 6](#_Toc64059661)

[Benefits 6](#_Toc64059662)

[The Data Product 7](#_Toc64059663)

[The Data 7](#_Toc64059664)

[Objectives and Hypothesis 7](#_Toc64059665)

[Methodology 8](#_Toc64059666)

[Funding 9](#_Toc64059667)

[Impact on Stakeholders 9](#_Toc64059668)

[Sensitive Data 9](#_Toc64059669)

[JAW Software Expertise 10](#_Toc64059670)

[Executive Summary 10](#_Toc64059671)

[Problem Statement 10](#_Toc64059672)

[Customer Description 10](#_Toc64059673)

[Gaps in Existing Data Product, if Applicable 11](#_Toc64059674)

[Data 11](#_Toc64059675)

[Methodology of Development 11](#_Toc64059676)

[Deliverables 12](#_Toc64059677)

[Implementation Plan 12](#_Toc64059678)

[Validating Results 13](#_Toc64059679)

[Programming Resources and Human Resources 13](#_Toc64059680)

[Project Timeline 14](#_Toc64059681)

[Data Product 15](#_Toc64059682)

[Methods 15](#_Toc64059683)

[Descriptive Method 15](#_Toc64059684)

[Predictive Method 16](#_Toc64059685)

[Datasets 16](#_Toc64059686)

[Decision-Support functionality 16](#_Toc64059687)

[Data Cleaning and Preparation 17](#_Toc64059688)

[Methods and Algorithms Supporting Data Exploration 17](#_Toc64059689)

[Data Visualization 20](#_Toc64059690)

[Implementation of Interactive Queries 22](#_Toc64059691)

[Implementation of Machine Learning 22](#_Toc64059692)

[Evaluating Accuracy of the Model 23](#_Toc64059693)

[Security 24](#_Toc64059694)

[Tools to Monitor and Maintain the Application 24](#_Toc64059695)

[Dashboard 25](#_Toc64059696)

[Documentation 28](#_Toc64059697)

[Business Vision 28](#_Toc64059698)

[Datasets 28](#_Toc64059699)

[The Code 29](#_Toc64059700)

[Hypothesis Acceptance 36](#_Toc64059701)

[Visualizations and Storytelling 36](#_Toc64059702)

[Model Accuracy 39](#_Toc64059703)

[Testing Results 40](#_Toc64059704)

[Source Code and Executables 42](#_Toc64059705)

[Guide to Installation and Use 42](#_Toc64059706)

[References 43](#_Toc64059707)

# A1. Letter of Transmittal

February 20, 2021

Samuel M. Ovie

Fine Movie Theaters

7946 W Some St.

Springfield, MO 47394

Dear Mr. Ovie,

In light of the huge volume of user-entered movie reviews that your staff is faced with every day, I am pleased to offer to you a new technology tool which I believe will ease the burden on your employees and streamline some of their work. As we have previously discussed, your firm is concerned about the large amount of time that is needed to examine and categorize every movie review that is entered into your online system. As your system currently works, a person reads every review and categorizes it as either a positive or negative review. This has grown from a small task to one that is now nearly unmanageable and requires far too much staff time.

I have a solution to offer. I have developed a prototype of an app which can categorize the reviews for you, requiring no human interaction at all except for your current information technology support staff. This is the type of work that is well-suited for today’s artificial intelligence to handle, freeing up staff to work on other tasks and saving your firm a large amount of money.

The funding required for this application is approximately $29,680. This amount includes all of the planning, development, testing, and installation costs. My company, JAW Software, is well-equipped to develop, test, and install this app for you.

We at JAW Software are looking forward to working with you on this project. Please contact me with any questions.

Sincerely,

Julie Walin

JAW Software

# A2 – Project Proposal

## Problem Summary

As Fine Movie Theaters (FMT) continues to expand and grow, it finds itself faced with a huge amount of user-entered information. This information needs to be examined and classified. Initially, it was fine for all of this information to be evaluated manually by employees, but recently, the job has become too large, and additional staff has been required. This is considered un-skilled labor, as the only skill necessary is simple reading of text. It is becoming very expensive for FMT to employ such a large staff simply to read text and classify it as positive or negative, and has taken staff away from other necessary tasks.

## Benefits

Fine Movie Theaters (FMT) simply has too large of a volume of movie reviews to continue to process them manually. Using the Review Sentiment Predictor App will allow the movie reviews to be categorized as positive or negative automatically, behind the scenes, without needing to pay and staff to do it manually. The staff hours saved will be a significant cost savings to FMT. With this large volume of accurate information available concerning how moviegoers are responding to each movie, FMT will be able to respond to but acclaim and criticism in a timely manner, and make well-informed decisions about movies and the preferences of the audience in the future.

## The Data Product

The app will load the data, automatically clean the data and prepare it for analysis. Next, some data visualization tools will be used to show information about the data, such as text length and the most commonly used words. That information will be available to view in a dashboard for FMT employees who may have an interest in the analysis of the raw data.

After cleaning, the data will be processed by a machine learning algorithm which will predict whether each individual review is a positive review, meaning the reviewer liked the movie, or a negative review, meaning they did not like it.

## The Data

The data collected will simply be the name of the movie being reviewed and the raw text of the review which was left by the viewer. The maximum length of the review will be limited by the review-entry software which already exists on the website of FMT.

The dataset used for the prototype can be seen at: https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews

This dataset contains only the text of the review and a prediction of the sentiment behind the review: positive or negative. It doesn’t contain the title of the movie. That will be supplied by FMT’s system at the time of the user entering the review.

## Objectives and Hypothesis

The main objective of this application is to relieve the tremendous burden on the staff of Fine Movie Theaters (FMT) in the huge task of classifying movie reviews as positive or negative, and to provide timely feedback to FMT about how audiences are responding to their movie offerings. The information provided by this product will allow FMT to respond far more quickly to positive viewer response, perhaps extending a popular movie, and also far more quickly to negative viewer response, perhaps ending the showing of an unpopular movie. Having this information so quickly, as early as mere milliseconds after a review is entered, will enable FMT to be more responsive to the desires of their customers.

The hypothesis is that JAW Software can expand upon the prototype that has been created to offer a robust, quick, and beneficial app that will predict the sentiment of a review with at least 85% accuracy. After the machine learning model has been trained with more data, the accuracy is predicted to increase. This application will be able to help FMT improve their operations, and, from that, increase customer satisfaction.

## Methodology

This project is well-suited to the waterfall method of project development. This methodology completes one category of tasks at a time, and flows to the next after one is complete. This is not a particularly complex project, and will require only a small staff. The prototype is already developed and working, and the data-entry portion of the project is already in production in FMR’s online system. Due to the project’s simplicity, requirements can be successfully gathered up-front.

## Funding

|  |  |  |  |
| --- | --- | --- | --- |
| Item | Cost, USD | Time, Hours | Total, USD |
| Planning and Design | 140 / hour | 40 | 5,600 |
| Development and Testing | 190 / hour | 100 | 19,000 |
| Installation | 190 / hour | 12 | 2,280 |
| Site Testing and Validation | 140 / hour | 20 | 2,800 |
|  |  |  |  |
| Maintenance agreement | 200 / hour | As needed |  |
|  |  |  |  |
|  |  |  |  |
| Totals |  | 172 | 29,680 |

## Impact on Stakeholders

This app will streamline the process of understanding and learning from the reviews submitted by moviegoers. It will allow Fine Movie Theaters (FMT) to use their resources far more efficiently, and will allow them to be more responsive to the tastes and wishes of their customers. It has the potential to improve the financial health and profitability of FMT, thus improving the value of the stock and potentially making FMT a more stable employer to work for. It gives FMT the information it needs to carry content that their audiences want to see. In light of all of these benefits, it adds value for all stakeholders.

## Sensitive Data

No sensitive data is collected during the collection of the user-entered movie review. No personal or identifying information is collected, and there is no way to trace a movie review back to the person who entered it into the system. FMT’s data-entry system to collect movie reviews already exists, and already guarantees anonymity of the user.

## JAW Software Expertise

The developer assigned to this project has 8 years of relevant experience developing software solutions, web apps, and data analytics products. In addition, the developer holds a Bachelor’s Degree in Computer Science and is well-trained in application development, security, and support. Your trust is well-placed when placed in JAW Software.

# Executive Summary

## Problem Statement

As Fine Movie Theaters (FMT) continues to expand and grow, it finds itself faced with a huge amount of user-entered information. This information needs to be examined and classified. Initially, it was fine for all of this information to be evaluated manually by employees, but recently, the job has become too large, and additional staff has been required. This is considered un-skilled labor, as the only skill necessary is simple reading of text. It is becoming very expensive for FMT to employ such a large staff simply to read text and classify it as positive or negative, and has taken staff away from other necessary tasks.

## Customer Description

The customer is Fine Movie Theaters (FMT), a medium-sized movie theater chain. They experience an unmanageably-large volume of new movie reviews each day, and need a way to analyze those reviews to detect the sentiments being expressed in each one.

The data visualization is expected to assist the data analysts employed by FMT to understand the content and sentiments expressed by their customers in their reviews without individuals needing to read and interpret each one.

## Gaps in Existing Data Product, if Applicable

The proposed Review Sentiment Predictor is not intended to replace any existing product.

## Data

The data that is needed to feed information into the Review Sentiment Predictor is the file already currently created by Fine Movie Theater’s Systems. It is a raw file currently consisting of only the review text. FMT’s user interface needs to be expanded to include collecting the name of the movie. The data must be cleaned and normalized, removing any null (missing) values.

The original dataset that has been used to train the machine learning model can be found at: <https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews> This is a .csv file which contains both the text of the reviews and a column to identify whether a review is positive or negative. The sentiment column of positive or negative is necessary in order to train the model to predict whether it is processing a negative or a positive review.

## Methodology of Development

This project is well-suited to the waterfall method of project development. This methodology completes one category of tasks at a time, and flows to the next after one is complete. This is not a particularly complex project, and will require only a small staff. The prototype is already developed and working, and the data-entry portion of the project is already in production in FMR’s online system. Due to the project’s simplicity, requirements can be successfully gathered up-front.

## Deliverables

* Project requirements document, signed by representatives from both Fine Movie Theaters and JAW Software
* Timeline
* Test Plan
* Installation Plan
* Maintenance Plan
* Modifications to existing FMT’s review system to add additional field of the name of the movie being reviewed
* The new, Review Sentiment Predictor web application
* Backup procedures

## Implementation Plan

1. Collecting Requirements and Planning – Meetings between representatives of Fine Movie Theatres and JAW Software will collect all of the requirements and complete the planning process. All involved parties will sign off on the plan.
2. Design – JAW Software will prepare detailed design plans, present them to Fine Movie Theatres (FMT) for approval, and obtain sign-off from FMT. Design will be flexible if FMT requests changes later, within reason. Larger changes after sign-off will require additional funding at the discretion of JAW Software.
3. Development and Testing – Software engineers at JAW Software will conduct the development and testing.
4. Milestones – Milestones will be tracked but will be flexible and adjustable as needed to accommodate any changes required.
5. User Testing – testing will be done both before and after implementation to ensure user acceptance. This is a fairly simple application and testing should be straightforward. The product will be available for user testing for 1 week before implementation and for 2 weeks after, during initial production use. After that time, the application will be considered to be accepted, barring any objections or requests for modifications.

## Validating Results

JAW Software’s Quality Assurance staff will test the completed application prior to installation. The minimum accepted accuracy rate is 96%. Extensive training of the machine learning model with large volumes of data should bring the model up to at least a 96% accuracy rate.

After installation, JAW Software will solicit acceptance information from users of the system to determine whether they feel their reviews were categorized correctly.

## Programming Resources and Human Resources

The application will require Python version 3.6 or higher. Development will be done in PyCharm with the use of libraries including but not limited to numpy, pandas, and scikit-learn. The app will be hosted by Amazon Web Services, along with the rest of Fine Movie Theater’s existing online systems.

Training datasets are freely available at no charge on the web, and will not be needed after the model is successfully trained to an accuracy of at least 96%.

The data collection resources already exist, as the existing review-collection systems used by FMT’s customers is going to continue to be used with only one minor modification.

Human resource costs: Two Software engineers for a total of 40 hours for planning and design and two Software Engineers for a total of 100 hours for development, for a total of 140 hours at a total estimated cost of $24,600 for Software Engineer staff expenses.

## Project Timeline

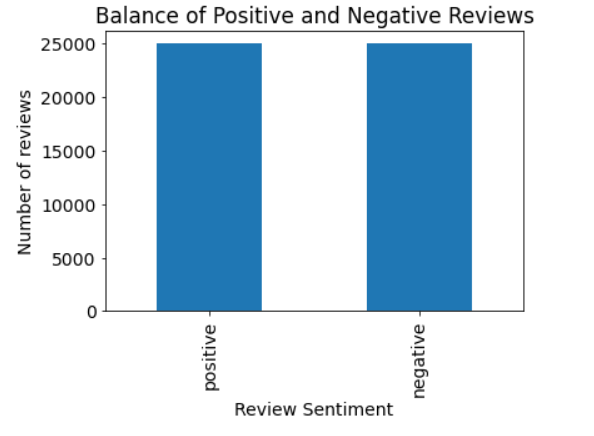
|  |  |  |
| --- | --- | --- |
| Task | Resources | Schedule, 2021 |
| Requirements | All Stakeholders | April 1 – 8 |
| Design | Designer and Software Engineers | April 9 - 23 |
| Design Review and Approval | Designer and Software Engineers, Customer Representatives | April 26 – 28 |
| Create and Train Machine Learning Model | Software Engineers | April 29 – May 20 |
| Create User Interfaces and Dashboards | Software Engineers, Designer | April 29 – May 20 |
| Demo and Approval | Software Engineers, Quality Assurance Analyst | May 21 – May 28 |
| Installation | Software Engineers | June 1 – 4 |
| Site Test and Sign-off | Quality Assurance Analyst, Customer Representatives | June 7 - 11 |

# Data Product

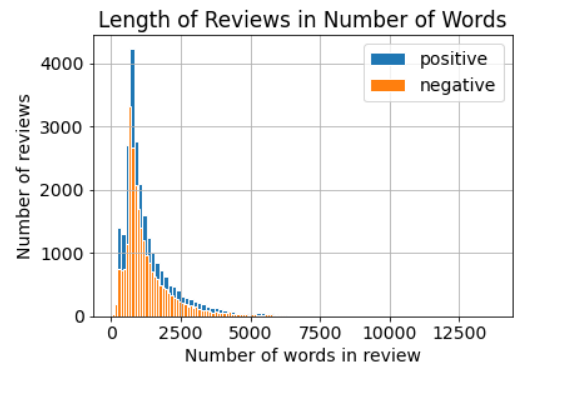
## Methods

### Descriptive Method

This bar chart shows the balance of positive and negative reviews in the combined dataset of both the training data and the testing data.

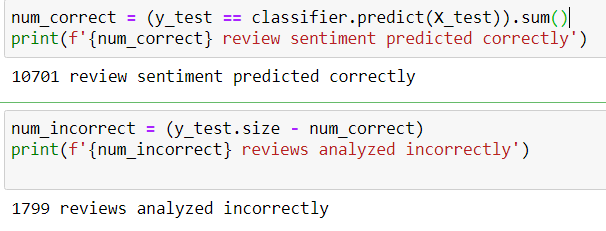


The following graph shows the length of negative reviews, in number of words, superimposed over a graph of the length of positive reviews, also measured in number of words. This graph shows that positive reviews tend to contain more words.



### Predictive Method

This application uses a Multinomial Naïve Bayes machine learning tool from scikit-learn to predict whether a review is positive or negative. The following is a copy of results from a test run of the trained model, using test data.



## Datasets

The .csv file of 50,000 reviews came from: <https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews>.

This dataset contains 25,000 positive reviews and 25,000 negative reviews. I used this data to train and to test the Multinomial Naïve Bayes Algorithm machine learning model that I created.

## Decision-Support functionality

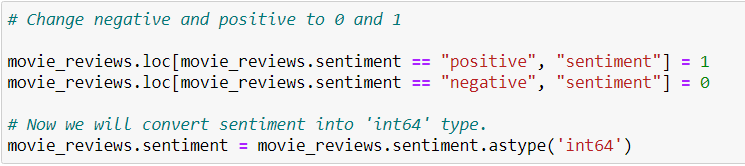
The Review Sentiment Predictor Application will support the decision-making process at Fine Movie Theaters (FMT) by providing timely, accurate information about what their customers are saying about the movies they have watched and about their whole experience at an FMT theater. This will allow FMT to be much more responsive to problems, because the information will be in FMT management’s hands quickly and efficiently. They will be able to extend movies that their customers are reviewing positively, and they will know when their customers are not responding favorable to a movie or are having other trouble with their movie theater experience.

## Data Cleaning and Preparation

This dataset did not require a great deal of cleaning and preparation before being visualized and fed into the machine learning algorithm. I checked for missing values, or nulls, and there were none. For some of the processing, I split the data into two separate datasets, one of only positive reviews and the other of only negative reviews.

Prior to feeding the data into the machine learning algorithm, I eliminated the common words that don’t carry any relevant meaning, such as “a”, “an”, and “the”. I also changed the text data “positive” and “negative” to a numeric 1 and 0, as that is what the algorithm needs in order to process the data.

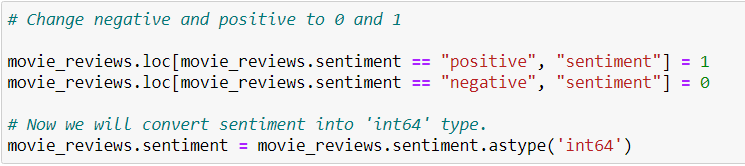
The following is the data cleaning code that was used to prepare the data.



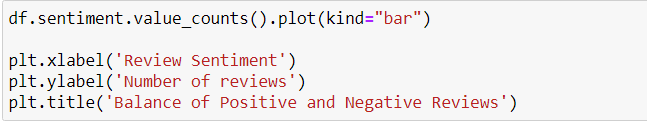
## Methods and Algorithms Supporting Data Exploration

Some of the data exploration and preparation tools used are as follows below:

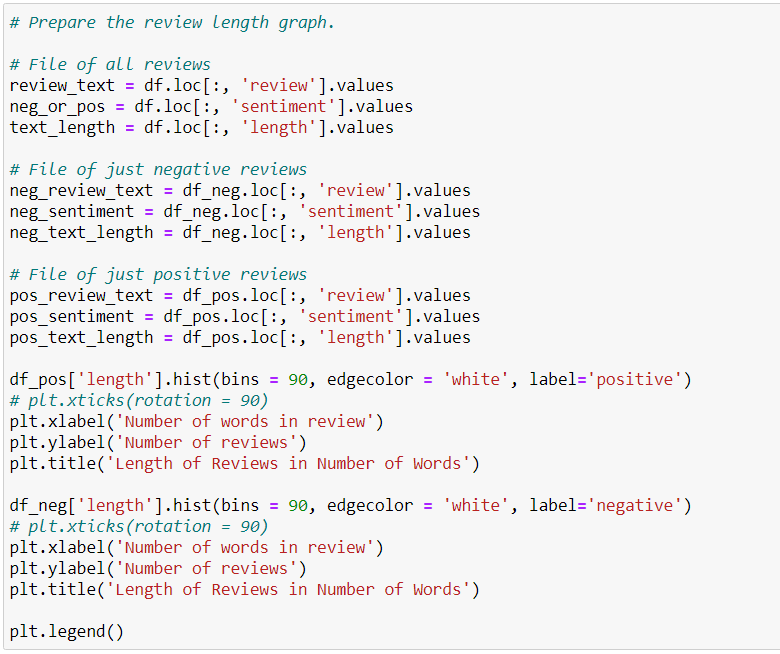
Cleaning and preparing the data:



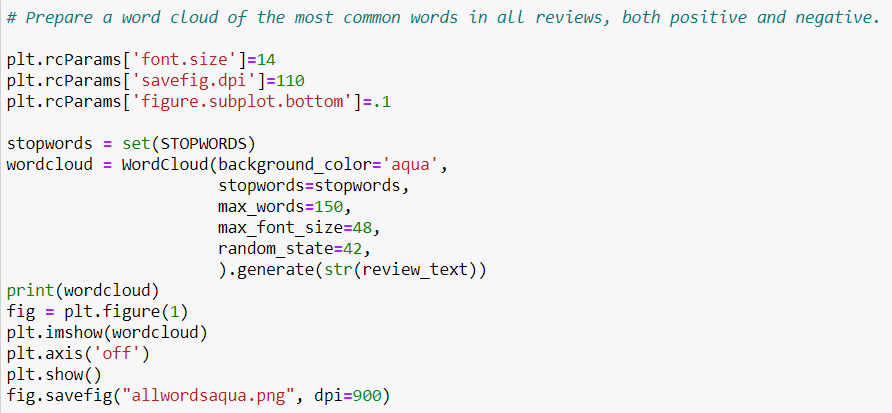
Making a bar chart for visualization of the balance between positive and negative reviews:



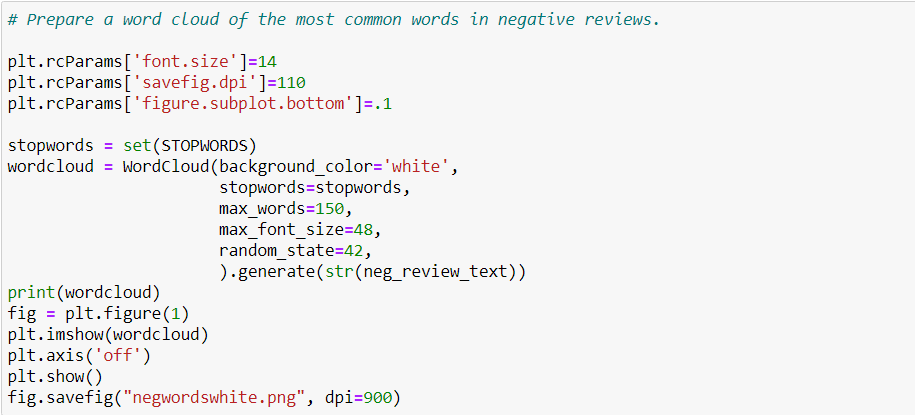
Making a chart of two graphs, one superimposed over the other, to visualize how the length of a review corresponds to whether the review is positive or negative:



Making word clouds to show 1) the most-used words in the whole dataset of 50,000 reviews, 2) the most-used words in all of the positive reviews, and 3) the most-used words in all of the negative reviews. The option “stopwords=stopwords” is used to remove the common but irrelevant words such as “a”, “an”, “and”, and “the”.

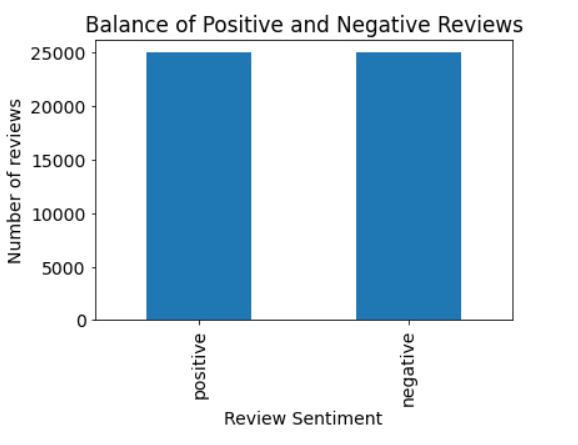




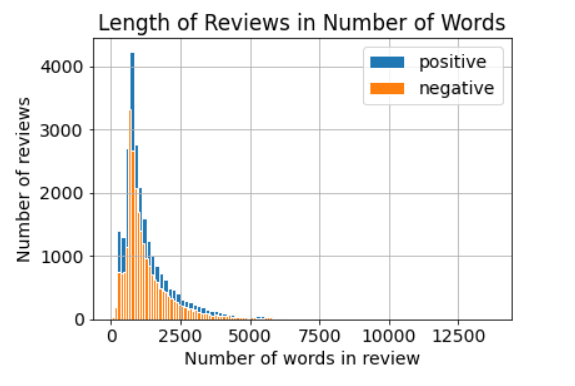


## Data Visualization

In order to understand the data better, some visualizations have been prepared. First, there is a bar chart to compare the balance of positive versus negative reviews. They are exactly equal at 25,000 reviews each.



Next, there is a bar graph of the negative reviews, in number of reviews and their length, superimposed over a bar graph of the positive reviews, in number of reviews and their length. This graph illustrates the idea that reviews containing more words are more likely to express a positive sentiment than negative.



In addition, three word clouds offer a visual representation of the most frequently used words, first in all of the 50,000 reviews, then in only the positive reviews, and then in the negative reviews.

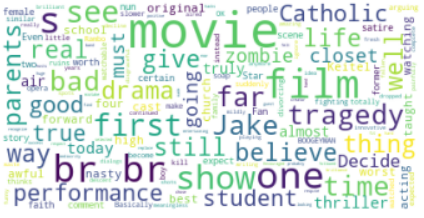
Here is the word cloud for the most commonly used words in all reviews. We can see that the top most commonly used words are “one,” “well,” and “movie.” A few other words seen in this word cloud are “comedy,” “violence,” and “characters.”



This is the word cloud for only the positive reviews. The most commonly used words in positive reviews are “movie,” “first,” “great,” “well,” and “watching.”

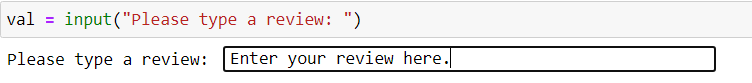


This is the word cloud for only negative reviews. The most commonly appearing words are “movie,” “film,” “tragedy,” “first,” and “show.” Some of the often-occurring words are “nasty,” “kill,” and “zombie.”



## Implementation of Interactive Queries

The prototype includes the option for a user to enter their own review and receive a prediction of whether the review is a positive or negative review.



## Implementation of Machine Learning

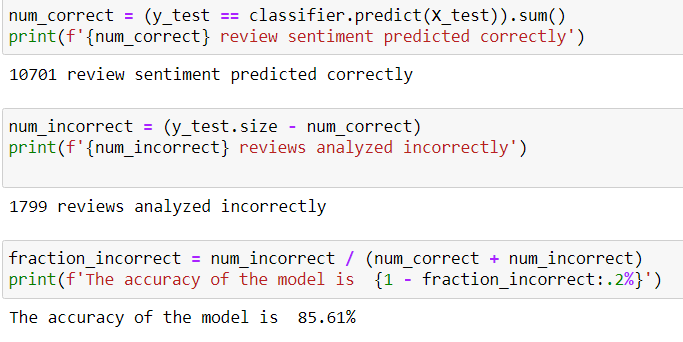
The Review Sentiment Predictor uses a machine learning algorithm called Multinomial Naïve Bayes. The data is cleaned and prepared, then I used 75% of the 50,000 records (37,500) in the original reviews dataset to train the model, then used the remaining 25% of the records (12,500) to test the model.

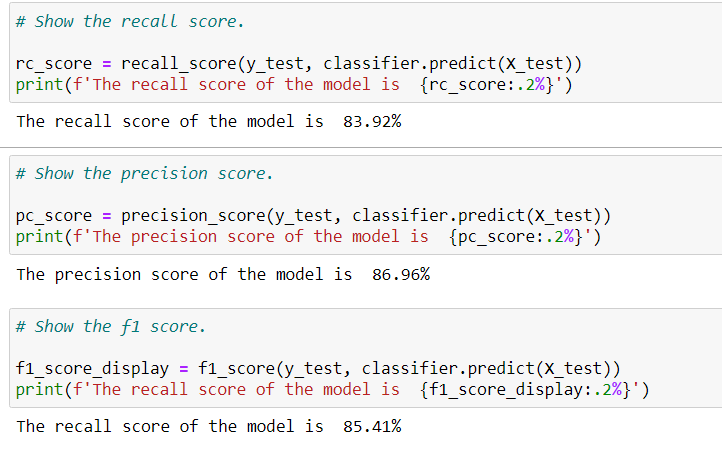
## Evaluating Accuracy of the Model

It is important to evaluate the accuracy of the model. With only 37,500 records to train the model in the prototype, a high percentage of accuracy is not expected, but for the production app, it can be trained with larger volumes of data. The additional, high volume training will eventually bring the model to a much higher percentage of accuracy.

After training, testing the model resulted in an 86% accuracy rate. The testing dataset contains 12,500 reviews. Of those, the model correctly classified the sentiment as either positive or negative for 10,701 of them, and incorrectly for 1799 of them, resulting in a total overall accuracy rate of 86%.

For the prototype, the accuracy results are:





## Security

* Only .csv files are permitted, and their file size is limited, thereby limiting denial-of-service attacks.
* No personal or identifying information of any kind is ever collected from the customers who enter reviews on the online system.
* No database queries are used at all in this application. All database access is done in FMT’s existing systems at the time of the creation of the review.
* The application will only be available within FMT’s intranet, and will not be available to the general public on the internet.

## Tools to Monitor and Maintain the Application

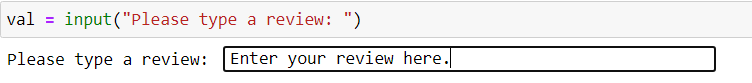
Every user session creates or appends to a log file, which contains details about the session. The data will include the session start and end time, the volume of reviews that were evaluated during the session, and the number of positives and negatives predicted. This will help analysts evaluate any anomalies, such as if the machine learning algorithm becomes unreliable, they can see if the model is predicting only positives or only negatives.

Retraining can and should be done any time a new volume of data becomes available, especially since new slang words frequently appear in common language, and the model will not know how to interpret these words until a new training dataset containing these new words in many reviews has been run through the model. This retraining should be done once per year at a minimum.

## Dashboard

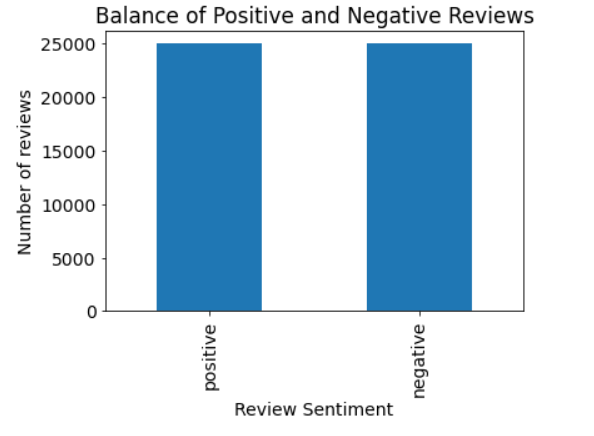
The dashboard is a very easy to use and well-documented Jupyter Notebook. Each cell contains commands. Each cell should be executed in order from the first cell, then the second cell, on through the whole notebook, until the last cell is executed last.

The Jupyter Notebook contains all of the data visualizations and the user interface, in which a person can enter their own review to receive a prediction from the model as to whether their review is positive or negative.

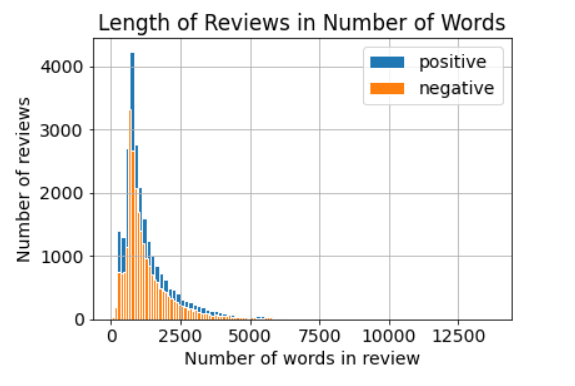


The visualizations provided on the dashboard are:

First, a bar chart showing the balance of positive and negative reviews in the training and testing combined dataset. It is a perfect balance of 25,000 positive and 25,000 negative reviews.



Next, there is a bar graph of the negative reviews, in number of reviews and their length, superimposed over a bar graph of the positive reviews, in number of reviews and their length. This graph illustrates the idea that reviews containing more words are more likely to express a positive sentiment than negative.



In addition, three word clouds offer a visual representation of the most frequently used words, first in all of the 50,000 reviews, then in only the positive reviews, and then in the negative reviews.

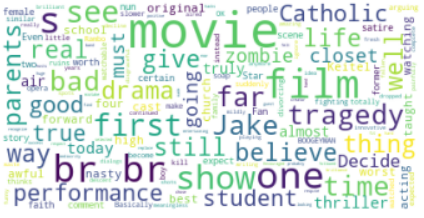
Here is the word cloud for the most commonly used words in all reviews. We can see that the top most commonly used words are “one,” “well,” and “movie.” A few other words seen in this word cloud are “comedy,” “violence,” and “characters.”



This is the word cloud for only the positive reviews. The most commonly used words in positive reviews are “movie,” “first,” “great,” “well,” and “watching.”



This is the word cloud for only negative reviews. The most commonly appearing words are “movie,” “film,” “tragedy,” “first,” and “show.” Some of the often-occurring words are “nasty,” “kill,” and “zombie.”



# Documentation

## Business Vision

The purpose of this project is to:

1. Create a machine learning model that can predict whether a review is positive or negative
2. Create an application that can be used to access the predictive model
3. Integrate the new machine learning predictive model with Fine Movie Theater’s Systems
4. Improve FMT’s responsiveness to its customers by providing this timely data to them

These objectives will be reached after coming to an agreement with FMT and obtaining a signed contract to plan, build, and deploy the application.

## Datasets

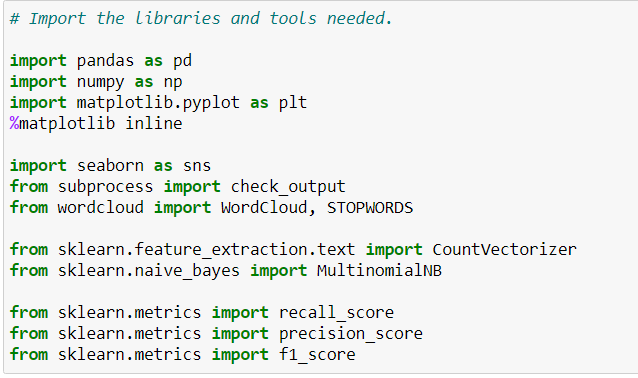
The original .csv file of 50,000 movie reviews was downloaded from (and can be viewed at) <https://www.kaggle.com/lakshmi25npathi/imdb-dataset-of-50k-movie-reviews>

Reviews\_negative.csv and reviews\_positive.csv are the two datasets of only negative and only positive reviews, respectively. They can be found in the reviews\_pre folder that will be submitted with this document. Imdb\_dataset.csv is the copy of the downloaded file that I used in the data analysis and training the model. It is also found in the reviews\_pre folder.

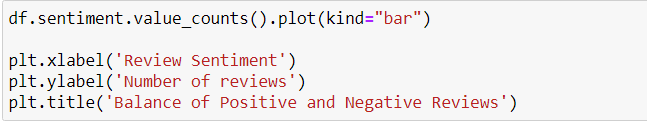
All remaining changes to the data in preparation for using it in the machine learning model were done within a dataframe, so the actual files were not changed and no additional files were made.

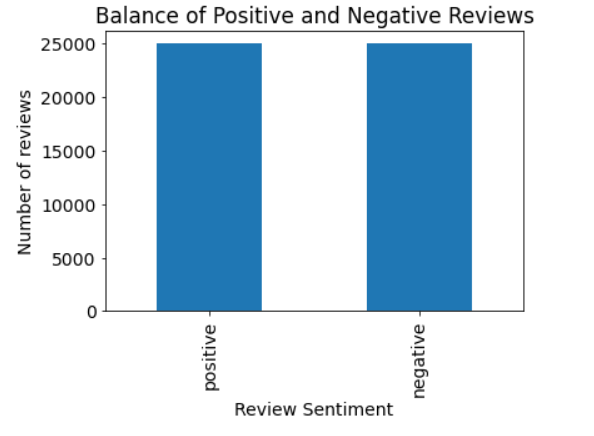
## The Code

The following imports were used to allow this application to use pandas, numpy, matplotlib, seaborn, scikit-learn, and others.

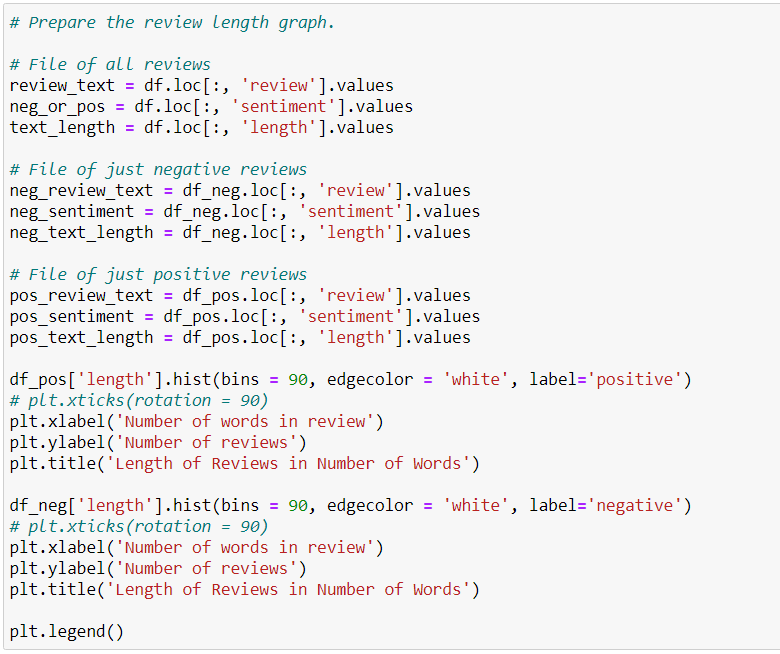


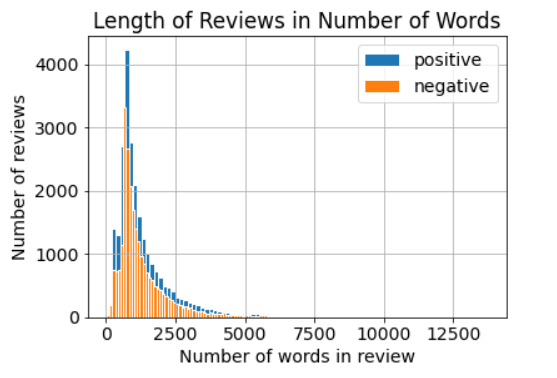
The following code is used to create a bar graph showing the balance between positive and negative reviews.





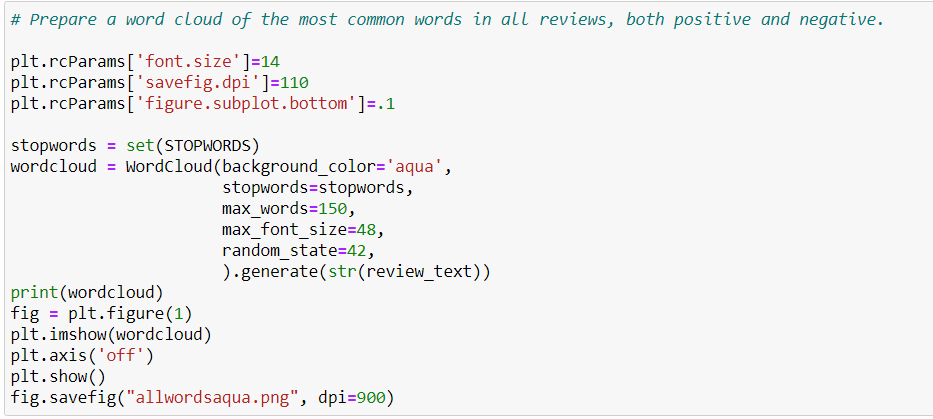
The code below is used to create a graph of the negative reviews, in word count, superimposed over a similar graph of positive reviews. The resulting visualization shows that longer reviews are more likely to be positive reviews.





In addition, three word clouds offer a visual representation of the most frequently used words, first in all of the 50,000 reviews, then in only the positive reviews, and then in the negative reviews.

Here is the word cloud for the most commonly used words in all reviews. We can see that the top most commonly used words are “one,” “well,” and “movie.” A few other words seen in this word cloud are “comedy,” “violence,” and “characters.”



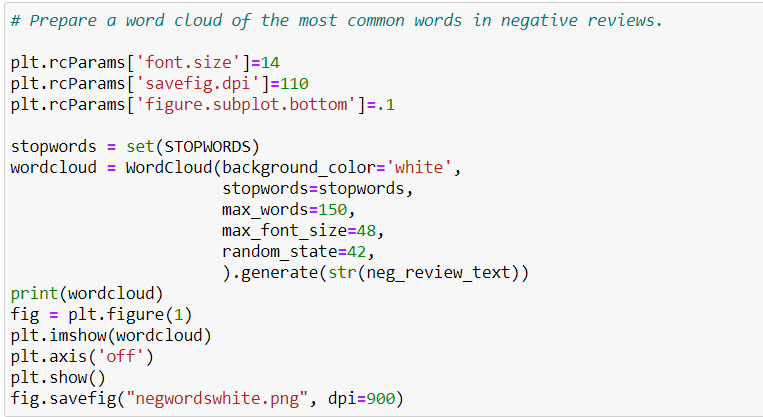


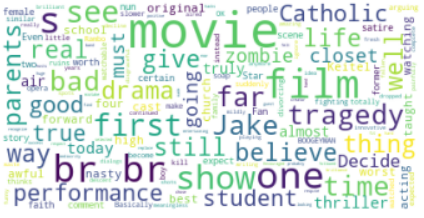
The following code creates the second word cloud, which shows the most-commonly used words in only the positive reviews.



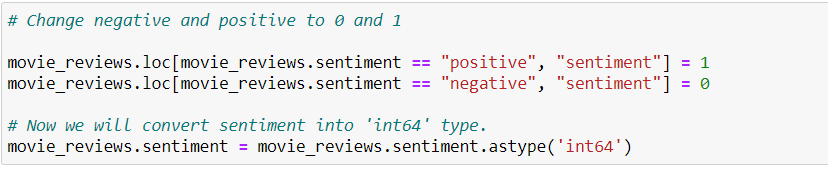


The following code creates the third word cloud, which shows the most commonly used words in the negative reviews.

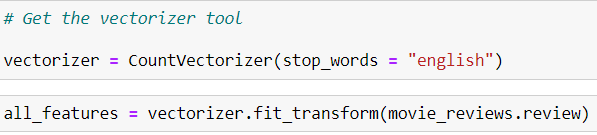




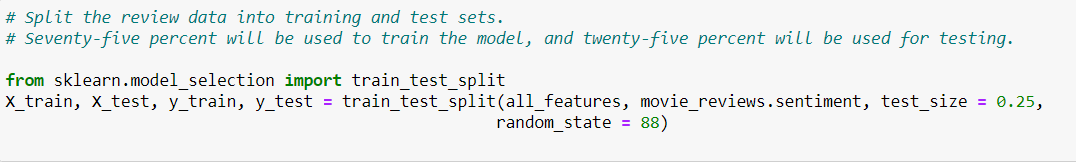
The following code modifies the data in the dataframe – not in the original .csv file – in order to change a column to numeric so the machine learning model will be able to process it.



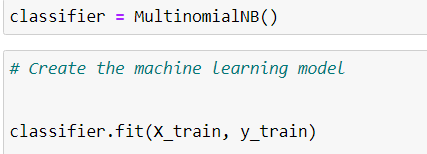
The code below gets the vectorizer tool in preparation for beginning to create the machine learning model.



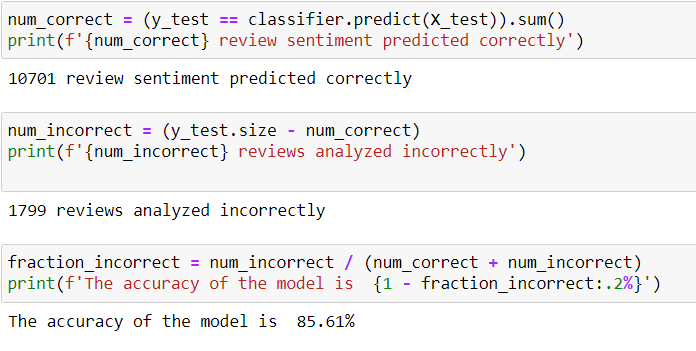
This code prepares the data by splitting it into 75% for training and 25% for testing.

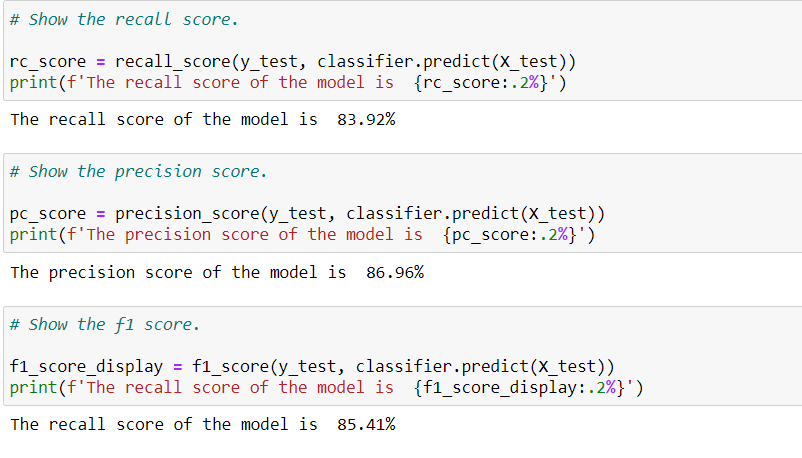


The following code creates the machine learning model as a Multinomial Naïve Bayes Algorithm.

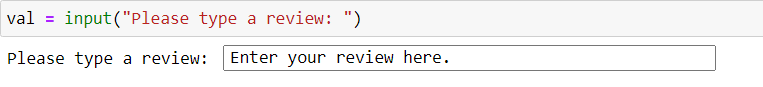


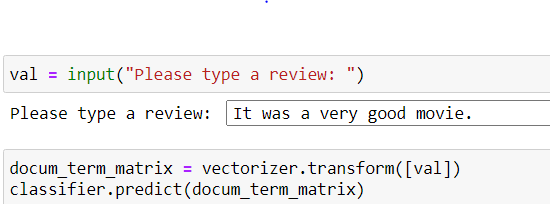
The following code evaluates the accuracy of the model after it was trained.



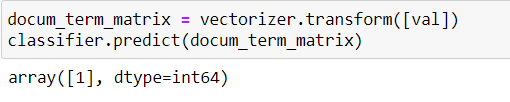


The following code allows a user to enter their own review and receive a prediction from the model as to whether their review is positive or negative.





The returned result is “1” which indicates a prediction that it was a positive review.



## Hypothesis Acceptance

The prototype of the model has achieved 86% accuracy with only one relatively small dataset to train it. Therefore, I believe that the model will become much more accurate as increasing volumes of data are passed through it for further training.

The first hypothesis, that a model can be created that will predict the sentiment of a review with at least 85% accuracy, has proven true and valid.

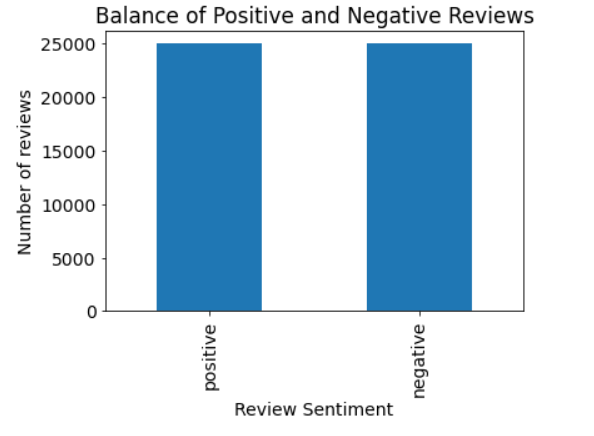
The second hypothesis, that a Multinomial Naïve Bayes Algorithm machine learning model can improve the operations and, from that, eventually the customer satisfaction, of a movie theater company, is likely sound and I believe the hypothesis is valid, but it will be up to the customer (FMT) to actually attempt to put it into practice, and the outcome will depend upon many variables at FMT.

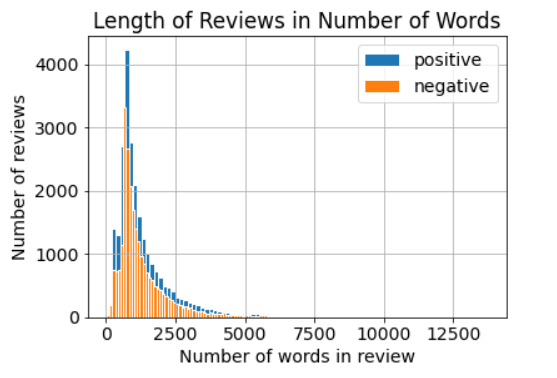
## Visualizations and Storytelling

Five visualizations are presented to aid in understanding the data and looking for patterns.

1. A bar graph showing the balance between positive and negative reviews in the input dataset of 50,000 reviews. This graph shows that there an equal number of each. This is significant because the model will have an equal chance to train for both recognizing positive reviews and recognizing negative reviews.
2. A graph showing negative reviews, by length of review in words, superimposed over a graph of positive reviews, also by length of review in words. This graph shows that longer reviews are more likely to be positive than negative.
3. A word cloud of the most commonly used words in all of the reviews, combined.
4. A word cloud of the most commonly used words in positive reviews.
5. A word cloud of the most commonly used words in negative reviews.

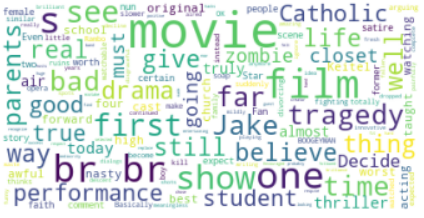
These five visualizations help the user become acquainted with the data and some general patterns.





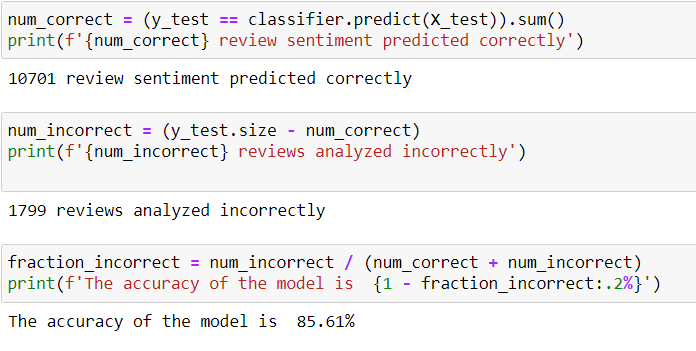


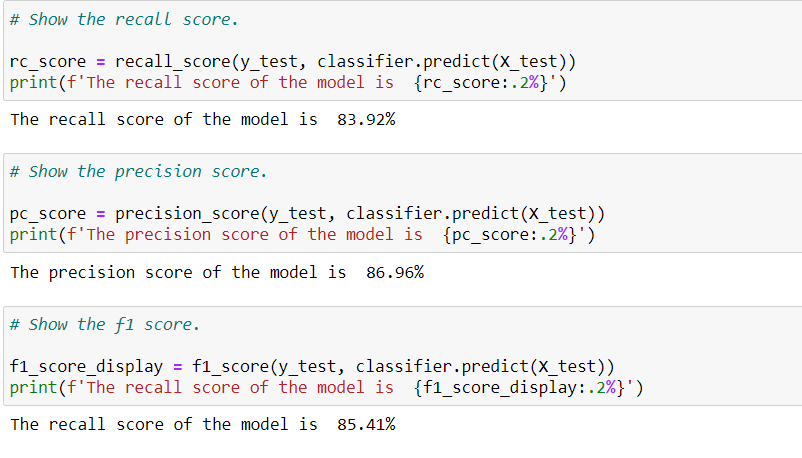




## Model Accuracy

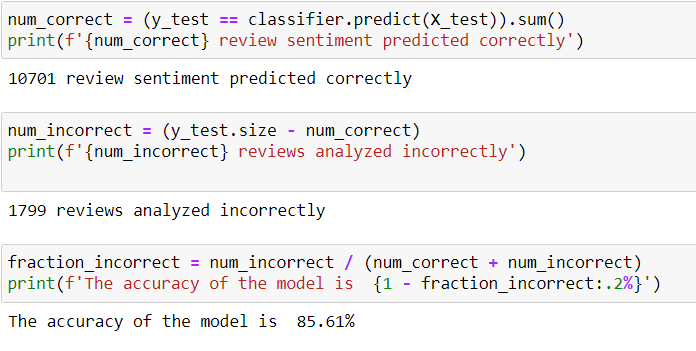
The following code evaluates the accuracy of the model after it was trained. The overall accuracy of predictions is 85.61%. That result can be improved in the future by running additional training data through the model so it can learn more words and more patterns of language.

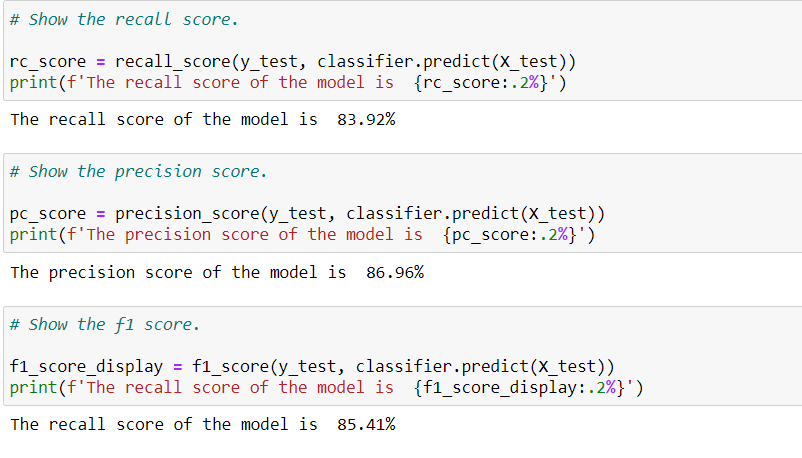




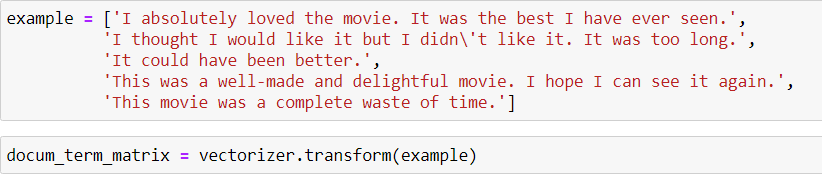
## Testing Results

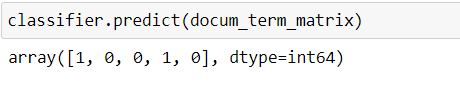
The original dataset contained 50,000 records. That dataset was split into 37,500 records to train the model and 12,500 to test the model’s accuracy. The following shows the results. The model is 85.61% accurate.



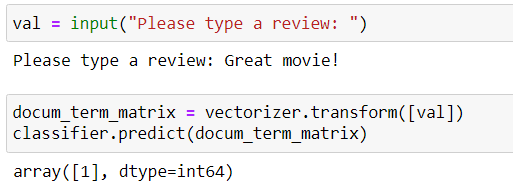


I also tested with a few additional reviews, and the model scored 100% on them, as shown below:





In addition, a user or tester can type their own review into the app, and receive a prediction:



## Source Code and Executables

The source code and executables are found in the reviews\_pre folder which will be uploaded with this document. The Jupyter Notebook is called prediction-notebook.

## Guide to Installation and Use

To execute this application, open the Jupyter Notebook called prediction-notebook, and execute the cells in order, waiting for each one to complete before executing the next one.

# References

None