**Project Report**

**Generic Sentiment Analysis and Classification on Movie Reviews**

**Team Members:**

Varsha Reddy Gummy Reddy

Venkata Jayanth Mandava

Harshini Reddy Neerugutti

Sai Sreya Sri Pitta

Github Link:

<https://github.com/Harshinireddyy19/NLP-Project-group9>

Goals and Objectives:

The sentiment analysis of a movie will rate how positive or negative a movie is, which gives us the overall rating of a movie. Hence, the process of recognizing if a movie review is positive or negative can be made automated as the machine learns while training the data.

Also, the knowing if the movie is watchable or not is a basic recommendation system that is found everywhere. We additionally added a classification to this system which tells the type of genre this system belongs to.

Motivation:

Whenever we try to buy a product online (from amazon or any other shopping website), we usually see the reviews of that product and buy it if it has good reviews. Some reviews may be good or bad and other reviews could be neutral and reading all the reviews to buy a single product is neither efficient nor time saving process. Therefore, using the Sentiment Analysis, it is possible for us to extract positive, negative and neutral words.

Similar to product reviews, Movie Reviews can help us in determining whether a particular movie is watchable or not.

Objectives:

The objectives of this idea are:

* To extract the opinions of the reviewers and classify the texts as positive and negative from various perspectives.
* To analyze the given reviews and sort out the movies to understand what their overall reaction to the movie was i.e., if they liked or hated the movie.

Significances:

Sentiment analysis is a method of determining the emotional tone behind a group of words that is used for understanding the opinions, attitudes and emotions expressed in an online mention.

* Here, we will be classifying the movie reviews based on whether the person liked the movie or not. This is useful when the director/creator of the movie wants to measure the overall performance of the movie using the reviews.
* The model that we are generating here can also be used for creating a recommender system that provides recommendations for the viewers to watch the movie or not based on their previous reviews.
* Another application can be to be find out a group of similar viewers with similar taste of the movies.

**Introduction**

Natural language processing, text analysis, and computational linguistics are used in sentiment analysis to extract and locate subjective information in source materials. It attempts to ascertain the stance a speaker or writer has toward a particular subject or the general contextual polarity of a work. His or her assessment or evaluation can be the attitude. Emotional condition The emotional message that is being told. The classification of the polarity of a text at the level of the document, sentence, feature, or aspect is a crucial problem in sentiment analysis. It focuses on the degree to which a document, a sentence, or a feature expresses an opinion and whether that opinion is good, negative, or neutral. It occasionally examines emotions other than polarity, such as "angry," "sad," and "glad." Subjectivity/objectivity identification is a task in sentiment analysis that concentrates on categorizing a given text (often a sentence) into one of the two classifications (objective or subjective). This issue might occasionally be more complex than polarity categorization since the subjectivity of words and phrases can rely on their context and an objective document may contain subjective sentences (such as a news piece that quotes people's thoughts).

There are four basic categories that can be used to categorize existing sentiment analysis methods. They are concept-level approaches, statistical methodologies, lexical affinity, and keyword detection. Based on the presence of words that clearly express affect, such as joyful, sad, fearful, and bored, keyword spotting categorizes text into different affect categories. By taking into account terms other than those with obvious influence, lexical affinity improves keyword-based strategy. Additionally, it presumes a "affinity" between various emotions and arbitrary phrases. Latent Semantic Analysis, Support Vector Machines, Bag of Words, and Semantic Orientation are just a few examples of machine learning components that statistical methods have an impact on. Concept-level approaches use components from knowledge representation, such as ontologies and semantic networks, in contrast to the solely syntactical techniques outlined above. This allows them to discover nuanced semantic expressions. Through the examination of concepts that are implicitly related to other concepts that do so, but which do not overtly convey pertinent information. A brand-new sentiment analysis method has been put out by Stanford University. Most traditional emotion prediction systems simply analyze words alone, assigning positive points for good phrases and negative points for negative words, before aggregating these values. In that method, the word order is disregarded, and crucial information is lost. In contrast, the new deep learning model of this strategy really creates a representation based entirely on the sentence structure. By considering how words combine to form longer phrases meanings, it computes the sentiment. With that strategy, the model is less susceptible to deception than earlier models.

**Background**

Minhoe Hur et al., 2016 [8] :

proposed a system for predicting box office collection based on movie review sentiment They have made use of In addition to predictors, viewer opinions are used as input variables, and three machine learning-based algorithms (artificial neural network, regression tree, and support vector regression) were used to obtain a non-linear relationship between the box office and its collection predictors.

Aurangzeb Khan, 2011 [19]:

proposed a rule-based technique for sentiment analysis for customer reviews and software reviews that uses SentiWordNet to achieve higher accuracy than a pure lexicon-based technique The proposed system has a document level accuracy of 91% and a sentence level accuracy of 86%.

Lei Zhang et al., 2010 [18]:

proposed an algorithm for ranking and extracting product features from opinion documents Initially, they had user reviews and it was difficult for the machine to distinguish between positive and negative reviews. They extracted product features using the associated rule mining technique.

So, these are some of the related works in our project. We are trying to implement sentiment analysis with the classification in which it is used to classify different classes where we are classifying as positive review ,neutral review and also negative review.

In all the other projects that we have gone through, we found only sentiment analysis with the movie reviews. But in our project, we extended this idea to add the classification and predict the respective genre of that review. This can be done using the naïve bayes classifier.

**Model**

Architecture of the model:

Word Features

negative tweets

positive tweets

Feature Extraction

Training Set

classifier

Tweet

Feature extraction

Genre

Positive

Negative

As we can see in the above chart, it is clear that a classification is being done. Initially, the tweets/reviews will be sent for training the model. The model will consider the features extracted and learn from these training sets. Later, it builds a classifier which has the ability to predict the required output given any related input. For example, if a tweet is given, it analyses the review and classify whether it is a positive review or negative one along with the genre of the movie based on some specific words present in it.

Wrokflow of the model:

Diagram

Description automatically generated

Initially after loading the data, the first step we are doing here is label encoding. That is, we are replacing the categorical outputs with the numerical ones so that the model building is easier. For example, the positive tweet is replaced with 1 and negative tweet with 0.

Later, The data is sent for cleaning or pre-processing where the removal of special characters, converting the entire data into lower case letters and removing the stop words. After this is done, the bag of words is created which is simply a representation of the data.

Its time for us to now start splitting the dataset into training and testing. The ratio considered here for the split is 80:20 where 80% of the data is used for training the model and 20% is used for the testing. After splitting the data, training and testing are done respectively. The algorithms used for implementing this model will be described later in the report.

**Dataset**

In this project we are using Sentiment analysis based on customer feedback about movies. It captures mindset of customer about movie. Our Data set consists of review given by movie lovers. Dataset consists of positive, negative, and neutral feedbacks, based we can provide review of a movie. Below is the sample dataset consisting of movie reviews. Movie Genre is another column that predicts genre of the movie such as Fantasy, history, sports, science fiction, crime, horror etc.., based on the review.

Text

Description automatically generated

Detail Design Features:

As the Dataset consists of thousands of reviews or sentiments. So, the text needs to be pre-processed dataset is messy, people write reviews on their own word its human understandable machines are lack of understanding of the unstructured text. For machine readable text need to be cleaned. In preprocessing text lowering the text, removing URLs, punctuations, stop words and misspelled words. Tokenization and lemmatization need to be applied to clean the text. The term movie genre is a character that doesn’t require any explanation about movie. Generally, movie genre is given based on synopsis and screen play. We have a dataset containing reviews feature extraction of reviews are done using any of the TF-IDF methods reviews are divided into bigrams and using multinomialNB genre is predicted from genre array. Example: Jack finished his job and fires one last bullet into hatchet, who is lying on the chair Consider above example this statement is passed through multinomialNB and it checks with training set values and returns thriller as genre because its consists of words such as fires bullet. In such a way genre is classified for a review.

* Text Exploratory Analysis is used to find most frequent words other than stop words but repeated words from data set.
* Word embedding is used to represent words as a numeric vector, words are encoded in real valued vectors it’s a crucial step for solving NLP problems. Simple embedding methods such as Word2Vec, GloVe, Bag-of-words, TF-IDF and ELMO are used for vector representation.
* Model Building is to select supervised classification model that should satisfies requirements. There are various algorithms for classification choosing the best algorithm is critical task.

**Analysis:**

* The whole movie review dataset mainly consists of 3 fields. One being the review itself, classification of the movie review being the second, and the genre of the movie reviewed is the third one. The third feature is the one that we added explicitly in order to implement our own extended version of movie review analysis.
* During the analysis, we had done the following:

1. **Cleaning the dataset:** First we had pre-processed the dataset- we removed the special characters, dropped the HTML tags if any, and converted all the text into lower case.
2. **Removing of Stop words:** And then also removed the stop words, in order diminish the unnecessary noise in the text.
3. **Stemming:** We applied the technology of stemming which helped in converting the original review into a minimised format that contains only important words that would contribute further in making the final sentiment analysis.

* For faster computation of the sentiment behind a review, we felt that Text summarization would also be helpful. Hence we performed text summarization on the datapoints. This helped to give a final summary for each review in each of the datapoints. And the final analysis on what the review is whether a positive one or a negative would be easier with this.
* **Bag Of Words:** We also used the bag of Words technique. With bag Of words, we got the frequency of each word in the finalised document. This will help in identifying what the document or each datapoint (a review) is talking about. If there are more positive words for example, then we can compute on finalising on the review being the positive one.
* The 2 regular type of reviews that we observed are the reviews being the summary of the movie, and other being how much did they like/unlike the movie. The summary part can be used to classify what genre of the movie is being reviewed. And the other type can be used for the sentiment analysis behind each word they expressed.
* But there can also be another kind of reviews where there can be mixed feelings expressed. Even in such condition, the model should be able to decide on the final sentiment behind it. This is was one of the things that we found out that could be an issue, during the evaluation of the model. This can tackled to some extend if the training dataset also contains such datapoints as result the model will learn accordingly, And an other way could be using the frequency method. Depending on the frequency of positive/negative words, we could take a decision the final sentiment.
* As the review’s final classification is solely dependent on the review written, which means it is mainly directly dependent on the words that are being used. Hence removal of stop words, using stemming techniques would help in focusing only on those that contribute the most to the final decision which was very important to remove the noise and improve the efficiency of the model.
* As a part of analysis, we have visualised the data as follows:
  + **Data Pre-processing:** This is how our dataset looks like

**Graphical user interface, text, application, email

Description automatically generated**

**Description of data:**

**Graphical user interface, application, table

Description automatically generated with medium confidence**

As the sentiment and the Genre features have categorical values, we wanted to replace them with simple numeric values for simpler computation. As the sentiment is of binary classification, we simply converted positive label to 1 and negative label to 0. And the Genre feature had many categories we went on and gave values like 0 for “Other genre”, 1.0 for “action” and so on.

Graphical user interface, application

Description automatically generated

And then we have checked if there were any null values, and replaced them with their corresponding mean.

As you can see below, the Genre field had 2 entries with null values. So we have replaced it with its mean. And checked for null values again as you can see, and it seems to be the data is ready for processing.

Graphical user interface, text, application

Description automatically generated

* + **Data Visualization:** As a part of analysis only, we wanted to identify the frequency of each class label that is present. For this, we used the bar graphs as below.

**Chart, bar chart

Description automatically generated**

And we have also visualised the count of datapoints (i.e the reviews) for each of the genre using the same bar graphs as follows:

Chart, histogram

Description automatically generated

* + **Data Cleaning:**

We wanted clean the data. i.e remove all the unnecessary characters from the review field. For this we have used regex, pattern matching techniques. Any character except for an alphabet will be replaced by a space. So with this, all the html tags, unnecessary special characters will be dropped.

Along with that simultaneously we had removed the stop words in each of the review.

**Graphical user interface, text, application, email

Description automatically generated**

**Implementation and Preliminary Results:**

* Coming to the actual implementation part, the main ML algorithm that we had simultaneously built 3 models- Gaussian, Multinomial, and Bernoullis.
* **Multinomial NB :** Multinomial naive bayes is a specialized version of naive bayes designed to handle text documents using frequency of the words occurred in it as it's underlying method of calculating probability.
* So, based on the words that occurred in a given corpus, its frequency and also based on the pattern of its occurrence, it will learn those patterns and then decide on the target label.
* For suppose, a given document that has vocabulary related to how thrilling, exciting a movie was, and the target label is also thriller. It will learn the words that finally converge to the target label- “Thriller”. And each time it encounters similar words that it has learnt, it will finalise that the genre was “Thriller”. And similarly, this will be the same for each and every target label.
* The training of the model includes:
  + Create a set containing the classes and a set for vocabulary encountered.
  + Create a data structure that maintains which classification that document belongs to, the word count for each word in that document, and the total number of words.
* This will help in calculating the posterior probabilities mainly and hence classify the genre.
* **Gaussian NB:** A Gaussian Naive Bayes algorithm is a special type of NB algorithm. It’s specifically used when the features have continuous values. It is best suitable when all the features are following a gaussian distribution i.e, normal distribution.
* An approach to create a simple model is to assume that the data is described by a Gaussian distribution with no co-variance (independent dimensions) between dimensions.
* This model can be generated by simply finding the mean and standard deviation of the data points for each label, which is all what is needed to define such a distribution.
* **Bernoullis NB:** Bernoulli Naive Bayes is a part of the Naive Bayes family. It is based on the Bernoulli Distribution and accepts only binary values, i.e., 0 or 1.
* If the features of the dataset are binary, then we can assume that Bernoulli Naive Bayes is the algorithm to be used. So, we can use this model to classify our sentiment for our movie review.

First we did split the dataset into train and test with 80:20 ratio. We have simultaneously given the training data to 3 of the classifiers from the python libraries as shown below:

Graphical user interface, text, application, email

Description automatically generated

After fitting the training data, we have tested the model by finding its accuracy by comparing the actual label and the model-predicted label as follows:

Graphical user interface, text, application

Description automatically generated

Chart, bar chart

Description automatically generated

After comparing the 3 models, we found that Multinomial NB was most suitable for our model. Gaussian also gave same accuracy, but as our project needs to predict both the sentiment and the genre, Gaussian would not be suitable for categorical values like Genre. Hence, we had chosen Multinomial.

Graphical user interface, text, application, email

Description automatically generated

**Project Management and Implementation Status:**

This project is managed by all members of the team involving in each step of the process such as data collection, analysis, and implementation.

Work Completed:

In this increment, we collected the data, analyzed it and preprocessed this data for training the model. We had to add a new column to our dataset as we decided to add a new feature of genre classification to the project. Then, the dataset is cleaned by using different techniques and preprocessed the data by removing stopwords which will otherwise impact the results and accuracy of the model. The model is then trained using 3 algorithms using bernoullis, guassian and multinomialNB. After this, the test data has been given to each of the models to find the highest accuracy model so that this model could be used for the next sentiment analysis and genre classification. Now, since the multinomial naïve bayes model has higher accuracy compared to the others, we used this for testing the outputs. These tweets have been used for two things in our project: i) They have been used for classifying whether the review is a positive one or a negative one ii) The genre of the movie is also classified based on the ngrams model.

Responsibility (Task, Person):

We altogether have surfed over the internet and decided to do this project. each one of us has researched on the project on what to do, how to implement it and analyzed the expected results for this project. We went through various websites and went through few existing projects for reference and found few loopholes and decided in overcoming these limitations.

**Varsha Reddy Gummy Reddy** Worked on the installations of the modules and libraries. She also analyzed the features of the project and analyzed the methods that should be used in this project.

**Venkata Jayanth Mandava** worked on collecting and loading of the data. He added a new column to the dataset and gave the values to each of the row by analyzing the data. Apart from this, he also designed the features that needs to be implemented in this project.

**Harshini Reddy Neerugutti** has done the data analysis like how the data should work on the project and worked on training and testing of the models implemented.

**Sai Sreya Sri Pitta** did the preprocessing of the data using python and other related tools as discussed. She also did the cleaning of the data and applied different techniques like stemming, stopwords removal, etc for the data to be used for training and testing.

Contributions (Members/Percentage):

All of the team members equally contributed to this project. Everybody had a crucial role in this project and worked as a team in building this.

Issues or concerns:

The major issues concerned with this idea is integrating the sentiment analysis with classification. Doing both of them separately is an easy task but analyzing the movie reviews along with predicting the genre based on the words present in the tweet is a difficult task. We need to find the keywords relating to the genre and the make a prediction of that review. Since naïve bayes is used in the classification of the textual data rather than numerical data, these algorithms can be used for the textual reviews we got.

**References:**

1. Datasets for movie reviews: <https://www.cs.cornell.edu/people/pabo/movie-review-data/>
2. About sentiment analysis: [IRJET-V6I9188.pdf](https://www.irjet.net/archives/V6/i9/IRJET-V6I9188.pdf)
3. [(PDF) Sentiment Analysis-An Objective View | J4R - Journal for Research - Academia.edu](https://www.academia.edu/25824227/Sentiment_Analysis_An_Objective_View)
4. <https://towardsdatascience.com/a-guide-to-text-classification-and-sentiment-analysis-2ab021796317>
5. <https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.BernoulliNB.html>
6. <https://towardsdatascience.com/multinomial-naive-bayes-classifier-for-text-analysis-python-8dd6825ece67>
7. <https://iq.opengenus.org/gaussian-naive-bayes/>