ISSN: 2584-0495 Vol. 2, Issue 3, pp. 700-705

International Journal of Microsystems and IoT

ISSN :(Online) Journal homepage: https://www.ijmit.org

Sentiment Analysis on IMDb Movie Reviews Using Machine Learning – Logistic Regression

Yeshitha B

Cite as: Yeshitha, B. (2024). Sentiment Analysis on IMDb Movie Reviews Using Machine Learning – Logistic Regression. International Journal of Microsystems and IoT, 2(3), 700–705. https://doi.org/10.5281/zenodo.11163160

Published online: 11 March 2024.

DOI: https://doi.org/10.5281/zenodo.11163160

International Journal of Microsystems and IoT
Vol. 2, Issue 3, pp. 700-705; DOI: https://doi.org/10.5281/zenodo.11163160

Sentiment Analysis on IMDb Movie Reviews Using Machine Learning – Logistic Regression

Yeshitha B

Department of Electronics and Communication Engineering, R.V. College of Engineering (RVCE), Bengaluru, Karnataka, India

ABSTRACT

Sentiment Analysis deals with handling large number of honest reviews given by the consumers and categorizing them into specific class labels. It helps the company or the brand to know if their product is useful to the public. This paper aims to use sentiment analysis for Movie Reviews. IMDb is one such popular and trustworthy platform for information on movies. It analyzes the reviews provided by the viewers which helps people to decide if the movie is worth watching. In this paper Natural Language Processing (NLP), sklearn and Logistic regression tools are used to identify and examine the sentiments of the IMDb movie reviews to train the model. The model performance is studied in terms of accuracy score. It has achieved a good accuracy of 90.064%. Hence it is believed to have good potential in analyzing customer feedback, product reviews and survey responses. It is also expected to perform better if integrated with Deep Learning models.

KEYWORDS

IMDb Movie Review; Long short-term memory; Machine Learning; Natural Language Processing; Opinion mining; Sentiment Analysis; Term Frequency-Inverse document Frequency (TF-IDF)

1. INTRODUCTION

Sentiment analysis helps in analyzing online reviews and critiques [1][2]. It is very crucial to understand customers feedback which helps the company or brand to work towards full filling customers' needs to sustain in their business [3]. Sentiments consists of a statement being Positive, Negative or Neutral [1][4], whereas emotion detection focuses on emotions such as happy, sad, angry etc. and not just polarity of the statement [2]. Figure 1 shows Polarity- classes of sentiment analysis. Customer's sentiments can be found in social media, Customer services, Marketing sector, and movie reviews etc. [1].

Types of sentiment analysis:

a. Fine-grained sentiment analysis: This type of opinion mining or sentiment analysis depends only on polarity of the statement, like positive, negative, or neutral [2].

b. Emotion detection: It detects emotions like disappointment, sadness, happiness, angry etc. It is also called the lexicon method of sentiment analysis [2].

c. Aspect based sentiment analysis: It concentrates on a particular feature [2].

d. Multilingual sentiment analysis: Analysis of sentiments in different languages [2].

Approaches used in Sentiment Analysis:

a. Rule-based approach: Here if the number of times positive words occurring is greater than negative words then it is classified as positive sentiment [2][5].

b. Automatic Approach: This uses Machine learning tools. Features are extracted from the text using libraries like NLTK, sklearn etc. Classification is done using algorithms like SVM, Logistic regression, Naïve Bayes etc. [2][5].

c. Hybrid Approach: It is combination of Rule-based approach and Automatic approach. Example: CNN- LSTM [1][2][4][5].

Even though Machine learning and Deep learning techniques are advancing on a larger scale, there are challenges in this field. The limitations include:

a. Emotion detection from speech where the data is in the form of tone.

b. Detection of emotions like sarcastic, ironic etc. [1][3].

c. Comparing neutral statements are also challenging.[3]

d. Words that convey different meaning depending on the context [1].

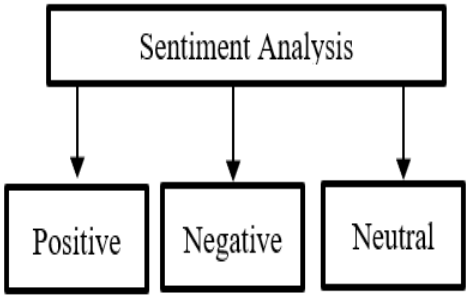

Fig. 1 Polarity

In this paper, Fine-grained sentiment analysis is carried out on IMDb movie reviews. Automatic approach with machine learning tools like logistic regression is opted to train the dataset. Testing data is used to validate model performance. The confusion matrix and accuracy score of the model is obtained. Output is classified as positive or negative.

2. LITERATURE SURVEY

In this paper "Sentiment analysis using multinomial logistic regression," [10], uses multinomial logistic regression for classification of reviews based on its polarity on Twitter dataset. The dataset was divided in 90:10 ratio that is 90% training and 10% testing, it also uses bag-of-words method of feature extraction for training the model. This method is taken as the base paper.

In the paper "Sentiment Analysis of IMDb Movie Reviews Using Long Short-Term Memory" [6], uses LSTM algorithm to classify the reviews into positive and negative statements. The words are converted into vectors by using Doc2Vec model which is then given as inputs to the algorithm. The algorithm contains an input gate, forget gate and output gate. It has the capability of remembering the previous work to take the decision about the reviews.

In the paper "Sentiment Analysis of Azerbaijani twits Using Logistic Regression, Naive Bayes and SVM," [7], compares different machine learning models using vectors that is, count vector, TF- IDF and bag of words. It was concluded that count vector and TF- IDF method of vectorization gave better results than bag of words.

In this paper "Twitter Sentiment Analysis Based on Ordinal Regression," [8], machine learning models like support vector regression (SVR), Random Forest, and Decision Tree (DT) are used to compare the performance. TF-IDF vectorization method is used. Decision Tree outperforms all the other algorithms.

3. PROPOSED SYSTEM

Figure 2 shows the proposed system flowchart for sentiment analysis of IMDb movie reviews.

3.1 Dataset

In this paper the IMDB dataset consists of 50K movie reviews. The sentiments are classified as Positive or Negative with respect to the IMDb ratings [6]. The .csv file required to train the model is available in Kaggle repository. Figure 3 shows the dataset used in training.

3.2 Splitting of Data

The dataset is split into training and testing. The training set is used to train the model and the test set is a subset of training.

set but is not used in training rather it is used to test the model performance after the training is done. [6]. 75-25 split is used in this paper, that is 75% for training and 25% for testing. Out of 50K entries 37.5K entries are used for training and 12.5K entries are used for testing.

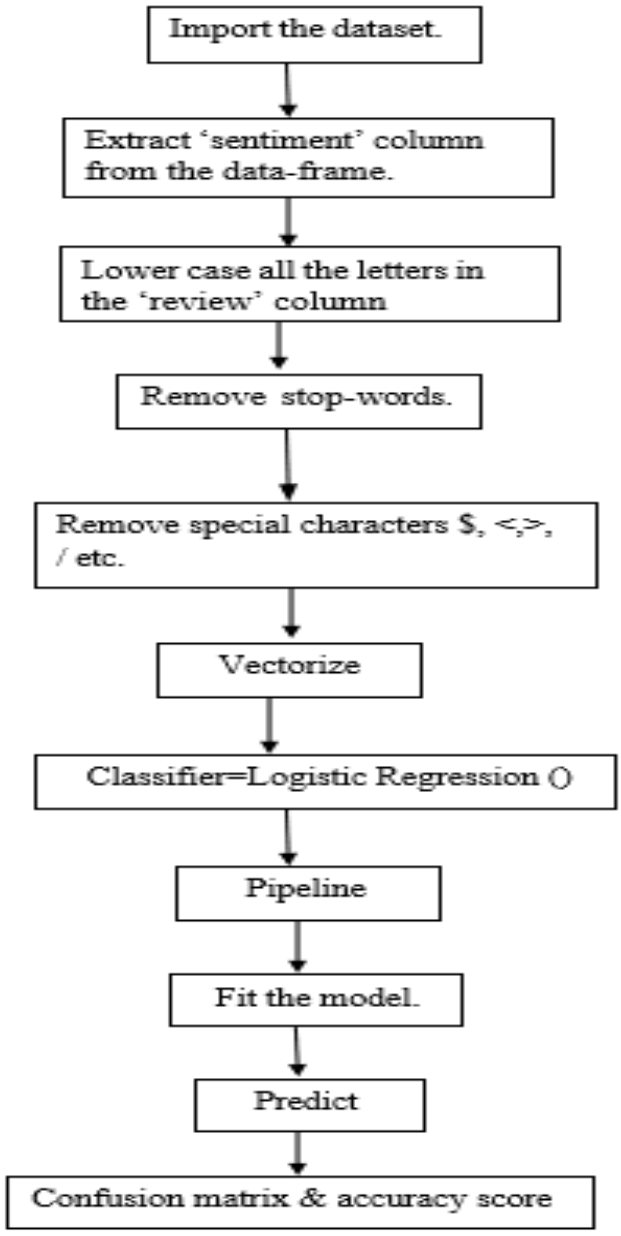

Fig. 2 System Flowchart

	A	B
1	review	sentiment
2	One of the other reviewers has mentioned that after watching ju	positive
3	A wonderful little production. The filming technique	positive
4	I thought this was a wonderful way to spend time on a too hot s	positive
5	Basically there's a family where a little boy (Jake) thinks there's a	negative
6	Petter Mattei's "Love in the Time of Money" is a visually stunnir	positive
7	Probably my all-time favorite movie, a story of selflessness, sacr	positive
8	I sure would like to see a resurrection of a up dated Seahunt ser	positive
9	This show was an amazing, fresh & innovative idea in the 70's w	negative

Fig. 3 Glimpse of the dataset used.

3.3 Data Preprocessing

Data Preprocessing is a crucial step as it involves cleaning the data which are not required for training hence enhancing the performance of the model after training. First step in data preprocessing is to convert the sentences into lowercase letters, second step is to remove stop words, third step removes anything other than letters (a-z or A-Z) and digits it can be symbols ($, %, <,>) or even emojis [6][7][8]. These steps can be implemented by using the Natural Language Toolkit (NLTK) library.

3.4 TF-ID Vectorization

Term Frequency-Inverse Document Frequency (TF-IDF) Vectorization is used. It explains the relevance of a word in a text. Each column contains a word which has a corresponding value which represents the importance of that word [7][8]. Let X be the of number of times a word appears in a sentence and Y be the number of words in a sentence then [7][8],

$$\text{Term Frequency} = X/Y \quad (1)$$

Let Q be the number of sentences and Z be the number of sentences that contain the word X [7][8]

$$\text{Inverse Document frequency} = \log (Q/Z) \quad (2)$$

As the denominator term in Equation (2) increases the importance of the word decreases [7]. The value in each cell is called as weight of the cell ($W_{x,y}$).

$$W_{x,y} = TF \times IDF = (X/Y) \times \log (Q/Z) \quad (3)$$

3.5 Logistic Regression

Logistic Regression is a type of classification algorithm. The output of the algorithm is discrete or is a categorical value such as Yes or No, 1 or 0, or Positive or Negative [7]. Equation (4) is the Sigmoid Function.

$$\sigma(z) = 1/(1+e^{-z}) \quad (4)$$

$$z = w_0x_0 + w_1x_1 + w_2x_2 + \ldots\ldots\ldots w_nx_n \quad (5)$$

($w_0, w_1 \ldots w_n$) are the regression coefficients ($x_1, x_2, \ldots x_n$) are the independent variable [9].

Binomial Logistic regression is used as the output contains only 2 possible types, Positive (1) or Negative (0) [7].

3.6 Confusion matrix and Accuracy score

Confusion matrix determines the performance of the trained model [6]. Since the output is the classification of 2 labels Positive and Negative, a 2×2 matrix is obtained. The accuracy score formula is given by Equation (6) [10]. TP, FP, FN, TN are True Positive, False Positive, False Negative and True Negative respectively [6]. Table I shows the standard confusion matrix of 2 × 2.

$$\text{Accuracy Score} = [(TP+TN) \div (TP+TN+FP+FN)] \quad (6)$$

Table I. Standard Confusion Matrix

	POSITIVE	**NEGATIVE**
POSITIVE	TP	FP
NEGATIVE	FN	TN

4. RESULT AND DISCUSSIONS

To test the working of the model, sample inputs as mentioned in Table III. was fed to the model. Table III shows the Actual and Predicted output of the model. Hence, it can be concluded that the sample inputs given to the model predicted the output which is same as the actual output. All 5 out of 5 sentiments are predicted correctly.

Figure 4, Figure 5 and Figure 6 show the predicted output of the model for a few of the sample inputs mentioned in Table III.

Table II. Confusion Matrix of IMDb Reviews

	POSITIVE	**NEGATIVE**
POSITIVE	5566	684
NEGATIVE	558	5692

From Table II. The accuracy score is 0.90064, obtained by using Equation (6).

Accuracy Score = (TP+TN) ÷ (TP+TN+FP+FN)
= [(5566+5692) ÷ (5566+5692+684+558)]

Accuracy Score in percentage = 90.064%
Figure 7 shows the confusion matrix obtained in simulation.

```
y_pred=model.predict(x_test)
ypred=model.predict(['its pretty bad movie '])
print(ypred)
```

```
['negative']
```

```
[16] from sklearn.metrics import accuracy_score, confusion_matrix
     accuracy_score(y_pred,y_test)
```

```
0.90064
```

Fig.4 Output of sample input (i)

```
y_pred=model.predict(x_test)
ypred=model.predict(['A must watch movie'])
print(ypred)
```

```
['positive']
```

```
[16] from sklearn.metrics import accuracy_score, confusion_matrix
     accuracy_score(y_pred,y_test)
```

```
0.90064
```

Fig.5 Output of sample input (iii)

```
y_pred=model.predict(x_test)
ypred=model.predict(['its pretty ugly movie. I recommend not to watch it'])
print(ypred)
```

```
['negative']
```

```
[16] from sklearn.metrics import accuracy_score, confusion_matrix
     accuracy_score(y_pred,y_test)
```

```
0.90064
```

Fig.6 Output of sample input (v)

```
from sklearn.metrics import confusion_matrix
matrix=confusion_matrix(y_test,y_pred)
print(matrix)
```

```
[[5566  684]
 [ 558 5692]]
```

Fig.7 Confusion Matrix obtained in simulation.

Table III. Actual Sentiment vs Predicted Sentiment of Sample Inputs (External Data)

Sentiment	Actual	Predicted
A Must Watch Movie.	Positive	Positive
It's A Horrible Movie.	Negative	Negative
It's Pretty Ugly Movie. I Recommend Not to Watch It.	Negative	Negative
It's Pretty Bad Movie	Negative	Negative
It's A Beautiful Movie!!	Positive	Positive

Accuracy Score in percentage = 90.064% Figure 7 shows the confusion matrix obtained in simulation. From Table III. Example (5) "It's pretty ugly movie. I recommend not to watch it." The novelty of the model lies in predicting correctly as negative regardless of the positive word "pretty" present before the negative word "ugly" without the use of LSTM algorithm.

Better accuracy can be obtained by integrating it with DL models, or by further cleaning the data.

5. COMPARATIVE STUDY

Logistic Regression showcased competitive performance, achieving an accuracy of 90% on IMDb movie reviews. Despite its simplicity, it demonstrated robustness in sentiment classification as against the pre-trained models like ResNet, VGG16, AlexNet which have accuracies of 85%, 86%, and 87% respectively [11].

6. CONCLUSION

Sentiment Analysis or opinion mining is used to extract information from the critiques, reviews, text, or comments and classify as Positive or Negative. This is a powerful marketing tool that enables us to understand customers' emotions, loyalty, customer satisfaction and acceptance. This project uses Logistic regression along with NLP, sklearn tools to classify the IMDb dataset. From the results it can be concluded that the model has achieved a highest accuracy score of 90.064%. The model could predict correctly during other positive words in a negative review without the use of LSTM algorithm.

TF-IDF method of feature extraction results in better performance when compared to paper [10] which uses bag-of-words for classification of polarity of statements.

Text based reviews encourages genuine feedback rather than rating from 0 to 5 but it would consume a lot of readers time in interpreting the data, hence using this algorithm ensures genuine feedback as well as reduced amount of time in analyzing the data by the viewers. This could be potentially used in customer feedback, product reviews or in survey responses.

It is expected to perform even better when integrated with DL models, or by further cleaning the data.

REFERENCES

1. R. R. Subramanian, N. Akshith, G. N. Murthy, M. Vikas, S. Amara and K. Balaji, "A Survey on Sentiment Analysis," 2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence), 2021, pp. 70-75, doi: 10.1109/Confluence51648.2021.9377136.
2. B. Saju, S. Jose and A. Antony, "Comprehensive Study on Sentiment Analysis: Types, Approaches, Recent Applications, Tools and APIs," 2020 Advanced Computing and Communication Technologies for High Performance Applications (ACCTHPA), 2020, pp. 186-193, doi: 10.1109/ACCTHPA49271.2020.9213209.
3. S. Srivastava, A. Nagpal and A. Bagwari, "Various Approaches in Sentiment Analysis," 2020 12th International Conference on Computational Intelligence and Communication Networks (CICN), 2020, pp. 92-96, doi: 10.1109/CICN49253.2020.9242618.
4. N. Raghuvanshi and J. M. Patil, "A brief review on sentiment analysis," 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), 2016, pp. 2827-2831, doi: 10.1109/ICEEOT.2016.7755213.
5. A. P. Jain and P. Dandannavar, "Application of machine learning techniques to sentiment analysis," 2016 2nd International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), 2016, pp. 628-632, doi: 10.1109/ICATCCT.2016.7912076.
6. S. M. Qaisar, "Sentiment Analysis of IMDb Movie Reviews Using Long Short-Term Memory," 2020 2nd International Conference on Computer and Information Sciences (ICCIS), 2020, pp. 1-4, doi: 10.1109/ICCIS49240.2020.9257657.
7. H. Hasanli and S. Rustamov, "Sentiment Analysis of Azerbaijani twits Using Logistic Regression, Naive Bayes and SVM," 2019 IEEE 13th International Conference on Application of Information and Communication Technologies (AICT), 2019, pp. 1-7, doi: 10.1109/AICT47866.2019.8981793.
8. S. E. Saad and J. Yang, "Twitter Sentiment Analysis Based on Ordinal Regression," in IEEE Access, vol. 7, pp. 163677-163685, 2019, doi: 10.1109/ACCESS.2019.2952127.
9. X. Zou, Y. Hu, Z. Tian and K. Shen, "Logistic Regression Model Optimization and Case Analysis," 2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT), 2019, pp. 135-139, doi: 10.1109/ICCSNT47585.2019.8962457.
10. W. P. Ramadhan, S. T. M. T. Astri Novianty and S. T. M. T. Casi Setianingsih, "Sentiment analysis using multinomial logistic regression," 2017 International Conference on Control, Electronics.
11. https://paperswithcode.com/sota/sentiment-analysis-on-imdb

Yeshitha B received her B.Tech degree in electronics and communication engineering from R V College of Engineering, India in 2023. Her areas of interest are embedded systems, Computer Networks, artificial intelligence, machine learning and human computer interaction.

E-mail: yeshithab.ec19@rvce.edu.in