International Journal of Innovative Research in Engineering
Volume 7, Issue 1 (January - February 2026), PP: 76-80.
https://www.doi.org/10.59256/ijire.20260701009
www.theijire.com
ISSN No: 2582-8746

Analyzing Customer Review Sentiments using Machine Learning

Omness Eliel Samuel Amari[1], Arun Udayasuriyan[2]
[1]*Department of Information Technology & Computer Science, Parul University, Gujarat, India.*
[2]*Assistant Professor, Department of MCA, Faculty of IT & CS, Parul University, Gujarat, India.*

How to cite this paper:
Omness Eliel Samuel Amari[1], Arun Udayasuriyan[2], "Analyzing Customer Review Sentiments using Machine Learning", IJIRE-V7I1-76-80.

Abstract: *Sentiment analysis is one of the fundamental techniques in natural language processing used to automatically detect customer reviews' opinions and emotions. With the rapid growth of online shopping platforms, the volume of customer feedback continues to rise each day, making manual analysis impractical. This paper presents a machine learning-based approach for analyzing the sentiment in customer reviews. Sentiment classification is done using Logistic Regression following text preprocessing and feature extraction methods. The proposed system demonstrates strong performance in accurately classifying customer sentiments and provides valuable insights for businesses to assess customer satisfaction and improve their services accordingly*

Keywords: *Sentiment Analysis, Text Mining, Machine Learning, TF-IDF, Logistic Regression, E-commerce Customer Reviews*.

I.INTRODUCTION

Sentiment analysis, also called opinion mining, is the computational process of identifying and classifying negative, positive, or neutral opinions in the text. This is true as sentiment analysis has gained considerable prominence over the last few years as online reviews, social media content, and customer feedback go through the roof. These textual data sources provide insight into what is popular with the public and in what consumer tastes. It is becoming increasingly important to organizations to analyze sentiment to understand attitudes of society, inform product and services enhancements, and help strategic decisions. In contrast, manual sentiment analysis is time-consuming, expensive, and inefficient, particularly on large-scale datasets. That has made sentiment analysis using machine learning techniques an active and valuable research area. In online commerce environments, online customer reviews are essential for influencing consumer behavior and buying decisions. By interpreting sentiment from these reviews, companies can glean useful insights into their customers' overall satisfaction and also where we can make changes. Automated sentiment analysis allows for efficient opinion mining, leading to large-scale analysis of textual data. In this study we aim to construct a sentiment classification model leveraging traditional machine learning methods applied to customer review data.

II. LITERATURE REVIEW

This includes multiple studies of e-commerce sentiment analysis techniques. Sudhir & al. [1] showed a comparative study of different strategies in sentiment analysis techniques and classifiers and their strengths and limitations. Basani & al. [2] used sentiment analysis on product review features on e-commerce-based platforms, which proves its effectiveness for getting insight on customer opinion evaluation. He & al. [3] introduced a fusion-based approach in sentiment analysis that improves the precision of the product experience evaluation. Similarly, Demircan & al. [4] proposed machine learning-based sentiment analysis models based on e-commerce data, highlighting the adaptability of the classic classifier approach for other languages and datasets. Pang & Lee (2008)[5] provided foundational methods for opinion mining using machine learning methodologies.

III. METHODOLOGY

3.1 Dataset Description.

The Women's Clothing E-commerce Customer Reviews dataset has been collected from Kaggle and consists of customer feedback for an online retail platform. Textual reviews and ratings of 1 to 5 were included in each record. So sentiment labels are assigned as:
Rating $\geq 4 \rightarrow$ Positive sentiment.
Rating $\leq 2 \rightarrow$ Negative sentiment.
We use neutral reviews to improve classification performance.

3.2 Algorithm Used

Logistic Regression is a supervised binary classification machine learning algorithm which is a logistic (sigmoid) function that estimates the probability of an input being in a class.

Why using Logistic Regression?

It's simple and interpretable, efficient on large text datasets and works perform well in sentiment classification.

3.3 Implementation Steps.

In the first phase of implementation the Users need to load Women’s Clothing E-commerce Customer Reviews dataset, a dataset that reflects customer's feedback for analysis. Afterward, the review text is preprocessed and cleansed for noise like punctuation, stop words and inconsistencies. This text is then converted into numerical vectors using TF-IDF vectorization, thereby allowing machine learning algorithms to interpret the data correctly. Logistic Regression model is trained on prepared data and sentiment predictions on test data. At last, our metrics of accuracy and classification report evaluate our model performance on what it shows overall.

IV. EXPIREMENTAL SETUP/ SYSTEM ARCHITECTURE

4.1 Experimental Setup.

All of our experiments were conducted on Google Colaboratory (Colab), that convenient cloud-based environment with available Python modules packed with libraries for data analysis and machine learning. All took place in Python 3.10, all in interactive Jupyter Notebooks, where we could easily tweak, execute and visualize results on the fly. I went with the classic suspects.Pandas and NumPy for data processing, Scikit-learn for model creation and training, NLTK for text processing. We uploaded the Women’s Clothing E-commerce Customer Reviews dataset directly to Colab to preprocess it there. After that, we split the data into 80% use for training and 20% for testing. I then applied TF-IDF vectorization to transform the text reviews into numbers the model could crunch and then trained a Logistic Regression classifier to predict sentiment. Finally, we tested whether it met performance criteria for accuracy, precision, recall, and F1-score .The implementation code is provided in the Appendix section in the form of code screenshots

4.2 System Architecture.

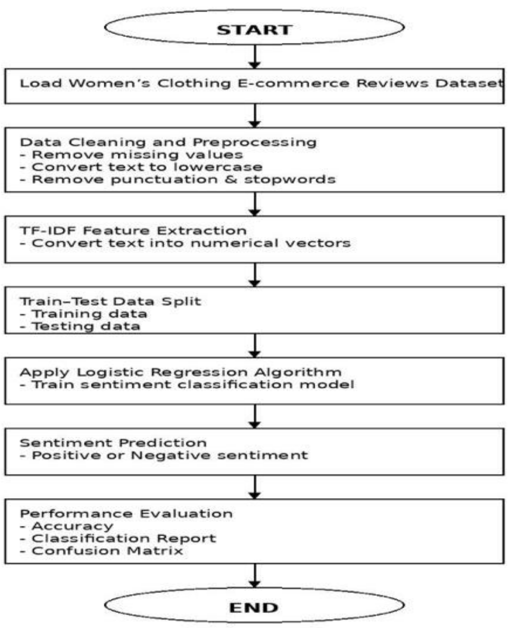

Figure 1: System Architecture of the Proposed Sentiment Analysis Approach

Our sentiment analysis system operates as if it were an assembly line: it gets messy customer reviews and sends out clear positive or negative labels at the end. Step is directed on specific single task to keep everything in order and a smooth flow. Raw reviews are next, we scrub them clean by removing punctuation, odd characters, and common stopwords, while all the reading is lowercase so “Happy” and “happy” are treated the same. Then, the cleaned text shifts to feature extraction, where TF-IDF converts words into numerical vectors. It strategically ranks words by importance — boosting the unique but downplaying frequent and unhelpful repeats — so the model can process them easily. The number vectors are then passed to our Logistic Regression classifier during training to observe the relationships between certain word combos and happy or grumpy customers. After training, it acts as a rapid predictive framework of new reviews sentiment. In the end, results come out as “positive” or “negative” labels for each review, validated with performance metrics to ensure valid accuracy. See Figure 1 for this full workflow for a visual map.

Figure 1 illustrates the workflow of the proposed sentiment analysis system. The process begins with loading the Women’s Clothing E-commerce Reviews dataset, followed by data cleaning and preprocessing steps such as removing missing values, converting text to lowercase, and eliminating punctuation and stop words. TF-IDF feature extraction is then applied to transform textual data into numerical vectors. The dataset is split into training and testing sets, and a Logistic Regression classifier is trained to perform sentiment classification. Finally, the system predicts customer sentiment and evaluates performance using accuracy, classification report, and confusion matrix.

V.RESULTS

The trained Logistic Regression model successfully applied the above-mentioned model correctly to classify customer sentiment based on the reviews. It distinguishes between positive and negative feedback, and can still perform reliable operations on unseen data with high accuracy as well.

Model Accuracy: 0.8207109737248841

Classification Report:

	precision	recall	f1-score	support
Negative	0.58	0.42	0.49	457
Neutral	0.50	0.21	0.30	588
Positive	0.86	0.98	0.92	3484
accuracy			0.82	4529
macro avg	0.65	0.54	0.57	4529
weighted avg	0.79	0.82	0.79	4529

Figure 2: Classification Report of the Logistic Regression Model

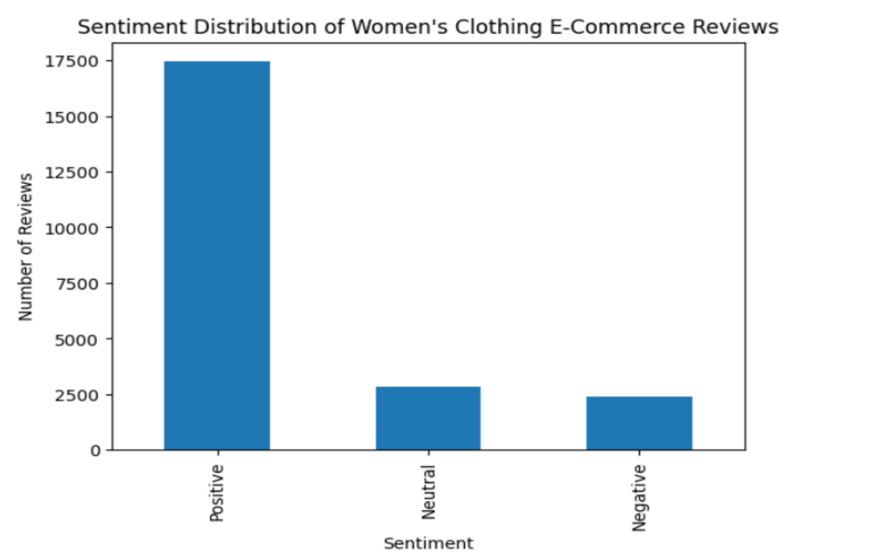

Figure 3: System Architecture of the Proposed Sentiment Analysis Approach

We examined our sentiment analysis model with the usual classification metrics of accuracy, precision, recall, and F1-score. The Logistic Regression approach was able to achieve an overall accuracy of 82.07%, showing a strong predictive performance for sentiment classifier with unseen and new test training data. As indicated in figure 2 , the total classification report has accuracy (precision), recall (recall) and F1-score (f1-score) for each sentiment class. The model succeeds in acquiring positive reviews and struggles with neutral & negative reviews probably since positive reviews are overrepresented in the dataset and create some class imbalance. The sentiment distribution given in Figure 3 for the Women's Clothing E-commerce Reviews dataset indicates the fact that positive reviews definitely lead. That asymmetry helps explain why the model performs better in cases of positives.

VI. DISCUSSION

Our research reveals that our sentiment analysis methods successfully sort customer reviews in the Women’s Clothing E-commerce Customer Reviews dataset. The Logistic Regression model achieved a performance around 82% on fresh test data, which is evidence that combining TF-IDF with this simple feature classifier is a reliable approach for text feedback. Upon viewing the classification report more closely, for instance, it performs exceptionally well at identifying positive reviews probably due to the fact that the latter are most of the data , while neutral and negative reviews seem to have lower precision and recall. This fits with other research with uneven class sizes and vague wording in neutral comments trip up models. Our findings are consistent with previous works, which lauded classic machine learning approaches such as Logistic Regression for sentiment learning on mid-sized data with TF-IDF features. More expensive deep learning could boost scores, but would require data and computational muscle. Note that the approach is simple and easy: Logistic Regression demonstrates which words matter most so it’s perfectly suited for real world use, and running it in Google Colab makes it easy to replicate. However, class imbalance had a negative impact on minority classes and only one model for testing restricted comparisons. Future work may utilize ensemble or neural network approaches to refine neutral/negative detection. In practice, e-commerce sites could use this to scour reviews and monitor satisfaction trends, and to drive smarter product, service and marketing decisions.

VII.CONCLUSION AND FUTURE WORK

In this work, we have found a sentiment analysis approach for a dataset of Women’s Clothing E-commerce Customer Reviews for classifying customers’ suggestions into positive, neutral, or negative categories. It performed recognizable text preprocessing, feature extraction (TF-IDF), and classification via Logistic Regression. The results demonstrate the model performs well and reaches an accuracy of approximately 82 percent in new test data. It does well on praise, but a little worse on the neutral and negative ones, primarily due to uneven distribution of classes in the dataset. Nevertheless, this validates that traditional machine learning, with adequate preprocessing, brings reliable results for sentiment tasks. Why Logistic Regression is so interesting in this regard is that it is clear and efficient, ideal for e-commerce applications that are important to know the why behind predictions. It was also easy to copy and expand running experiments in Google Colab. However, there are limitations: bag-of-words approach neglects word context and relationships, and class imbalance hurt minority classes. Going further, we might consider using things like word embeddings, like Word2Vec or GloVe, along with deep learning models like LSTMs, CNNs or Transformers to help us understand subtleties more. Adding the features of data balancing, tweaking the hyperparameters, multilingual support, or real-time analyses can take it to another level.

References

1. Sudhir & al. Comparative study of various approaches, applications and classifiers for sentiment analysis. Prajval Sudhir, Varun Deshakulkarni Suresh, et al., Global Transitions Proceedings, Vol. 2, Issue 2, pp. 205–211, 2021. Available on Comparative study of various approaches, applications and classifiers for sentiment analysis-SciEngine .
2. Basani . Application of Sentiment Analysis on Product Review (E-Commerce). Y. Basani (Yuniarta Basani), “Application of Sentiment Analysis on Product Review E-Commerce,” (conference/technical report), 2019. Available on (PDF) Application of Sentiment Analysis on Product Review E-Commerce .
3. He & al. Exploring E-Commerce Product Experience Based on Fusion Sentiment Analysis Method. He, Zhou & Zhao (authors listed in several copies), “Exploring E-Commerce Product Experience Based on Fusion Sentiment Analysis Method,” Res Militaris / conference article (online), 2022–2024 (published copies vary by repository). Available on RES MILITARIS .
4. Demircan & al. Developing Turkish sentiment analysis models using machine learning and e-commerce data . M. Demircan, A. Seller, F. Abut, M. F. Akay, International Journal of Cognitive Computing in Engineering, Vol. 2, pp. 202–207, 2021. Available on Developing Turkish sentiment analysis models using machine learning and e-commerce data - ScienceDirect .
5. Pang & Lee .Opinion Mining and Sentiment Analysis (survey / foundational methods). Bo Pang & Lillian Lee, “Opinion Mining and Sentiment Analysis,” Foundations and Trends in Information Retrieval (survey), 2008. Available on omsa.pdf . [6] M. Exwell, “Women’s E-Commerce Clothing Reviews Dataset,” Kaggle, 2018. [Online]. Available at: https://www.kaggle.com/datasets/mexwell/womens-e-commerce clothing-reviews

Appendix A: Implementation Code

This appendix presents screenshots of the Python code used to implement the proposed sentiment analysis system. The code was executed in the Google Colaboratory environment.

Step 2: Collect Reviews (Load Dataset)

```
# Load the dataset
data = pd.read_csv("Womens Clothing E-Commerce Reviews.csv")

# Display first 5 rows
print(data.head())
```

```
   Unnamed: 0  Clothing ID  Age                    Title  \
0           0          767   33                      NaN
1           1         1080   34                      NaN
2           2         1077   60  Some major design flaws
3           3         1049   50         My favorite buy!
4           4          847   47         Flattering shirt

                                         Review Text  Rating  Recommended IND  \
0  Absolutely wonderful - silky and sexy and comf...       4                1
1  Love this dress!  it's sooo pretty.  i happene...       5                1
2  I had such high hopes for this dress and reall...       3                0
3  I love, love, love this jumpsuit. it's fun, fl...       5                1
4  This shirt is very flattering to all due to th...       5                1

   Positive Feedback Count  Division Name Department Name Class Name
0                        0      Initmates        Intimate  Intimates
1                        4        General         Dresses    Dresses
2                        0        General         Dresses    Dresses
3                        0 General Petite         Bottoms      Pants
4                        6        General            Tops    Blouses
```

Figure A1: Python Code for Loading the Dataset

Step 5: Feature Extraction (TF-IDF)

```
vectorizer = TfidfVectorizer(stop_words='english', max_features=5000)

X = vectorizer.fit_transform(data['Clean Review'])
y = data['Sentiment']
```

Step 6: Train–Test Split

```
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)
```

Step 7: Sentiment Classification (Machine Learning Model)

```
model = LogisticRegression(max_iter=1000)
model.fit(X_train, y_train)
```

```
LogisticRegression
LogisticRegression(max_iter=1000)
```

Figure A2: TF-IDF Feature Extraction Code, Train–Test Split Implementation, Logistic Regression Model Training Code