Sentiment Analysis on Restaurant Reviews

1. Title Page

Project Title: Sentiment Analysis on Restaurant Reviews
Subtitle: Using Machine Learning and Natural Language Processing
Name: SyedShahid
Organization: SLASH MARK IT Solutions
Submission Date: 29/05/2025

---

2. Abstract

This project aims to automate the classification of restaurant reviews into Positive and Negative sentiments using sentiment analysis. In today's digital environment, customer reviews significantly influence business reputations. However, manually assessing sentiment from large volumes of data is inefficient. This project uses a machine learning pipeline involving text cleaning, vectorization via TF-IDF, and a classification model (Logistic Regression) to analyze review sentiment. The dataset is manually created and cleaned using basic NLP techniques. The model achieved satisfactory results, correctly classifying sentiments in most test cases. This beginner-friendly project provides hands-on exposure to text processing, feature extraction, and classification techniques using Python. It highlights the importance of automated sentiment analysis in decision-making for the food industry.

---

3. Table of Contents

4.Introduction

Background

Sentiment analysis helps analyze opinions, sentiments, and emotions expressed in textual data. It is widely used in marketing, product feedback, and service reviews.

Objective

To build a basic sentiment classification system to categorize restaurant reviews into positive or negative.

Relevance

This project helps restaurant owners and review platforms automate the analysis of feedback, improving service and customer satisfaction.

---

5. Problem Statement

Manual analysis of customer feedback is inefficient and error-prone. Businesses require automated systems to determine public sentiment from reviews. The project seeks to develop a tool to classify textual reviews as either positive or negative using ML.

---

6. Scope of the Project

Inclusions:

- Manual dataset preparation
- Preprocessing and cleaning
- TF-IDF vectorization
- Sentiment classification using ML

Exclusions:

- Neutral sentiment
- Deep learning models
- Real-time data scraping

Constraints:

- Limited dataset size
- Dependence on textual clarity and language

Assumptions:

- Input data is in English
- Sentiments are binary (positive or negative)

---

7. Literature Review

Past research has demonstrated the use of NLP and ML in analyzing customer feedback. Traditional approaches like Naive Bayes and SVMs have been effective in text classification tasks. However, many solutions fail to address small dataset constraints. This project adapts existing methods for smaller, manually labeled datasets.

---

8. Methodology

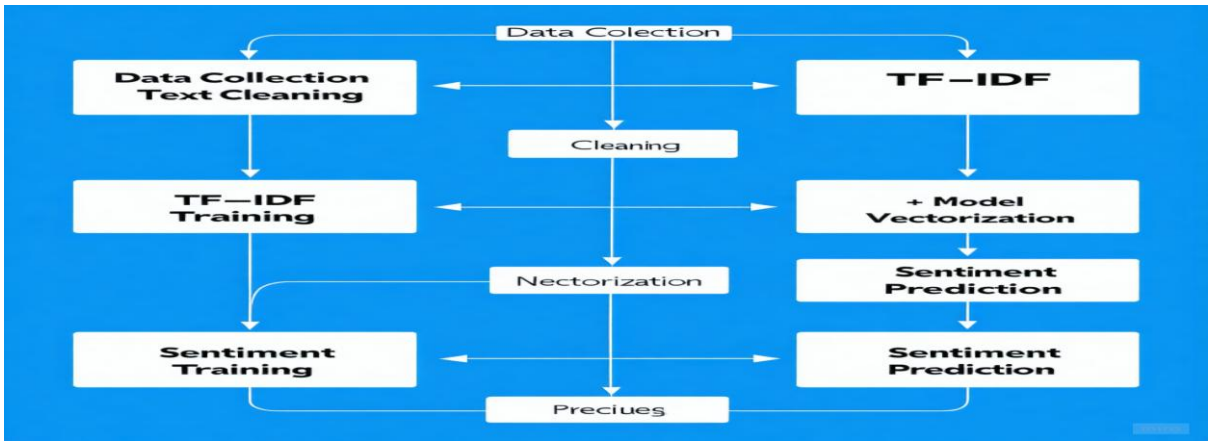
Approach

1. Collect and label data manually
2. Preprocess using NLTK and regex
3. Apply TF-IDF vectorization
4. Train Logistic Regression model
5. Evaluate accuracy

Technologies and Tools Used

- Google Colab
- Python
- Pandas, Scikit-learn, NLTK

Process



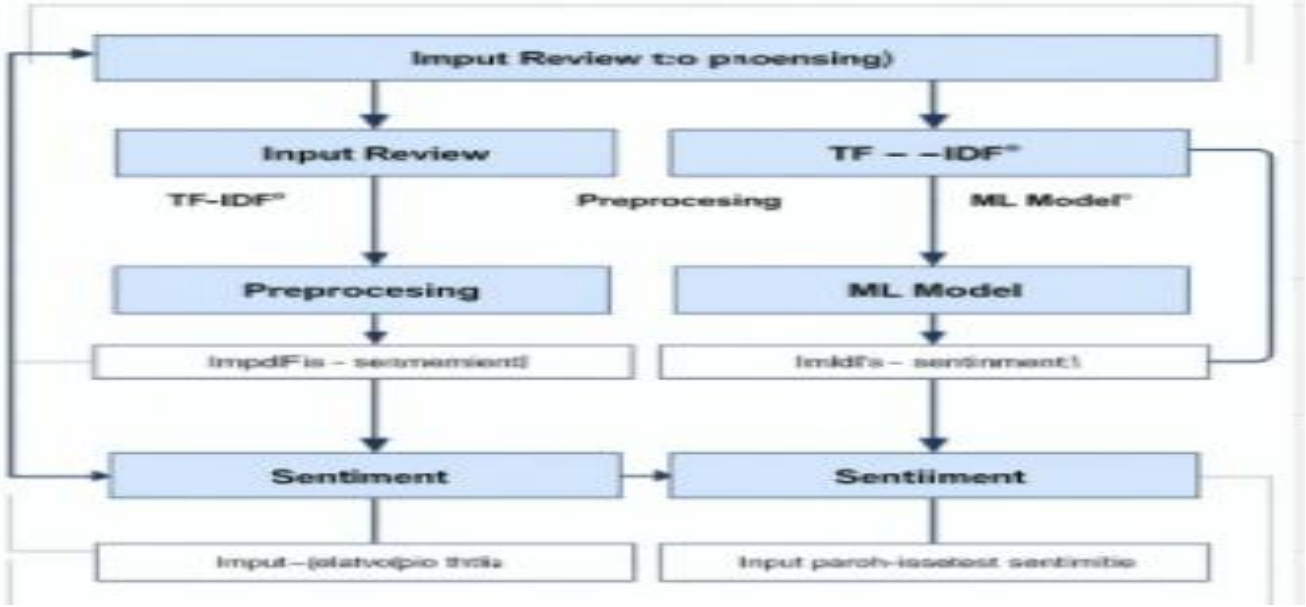
---

9. System Design and Architecture

System Overview

A simple pipeline that accepts review text, cleans and vectorizes it, and classifies the sentiment.

Architecture Diagram



Data Flow

- Text input is cleaned
- Features are extracted using TF-IDF
- Classification is done using ML model

---

10. Implementation

Modules

- Data Module: Manual entry and CSV creation
- Cleaning Module: Tokenization, stopword removal
- Vectorization: TF-IDF model

- Model: Logistic Regression

Code Snippet

```
from sklearn.naive_bayes import MultinomialNB
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, classification_report

# Split into train and test sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)

# Create the model
model = MultinomialNB()

# Train the model
model.fit(X_train, y_train)

# Predict
y_pred = model.predict(X_test)

# Results
print("Accuracy:", accuracy_score(y_test, y_pred))
print("\nClassification Report:\n", classification_report(y_test, y_pred))
```

Integration

Modules are integrated in a single Colab notebook.

---

11. Testing

Testing Methods

- Manual testing using test samples
-

	Review	Sentiment
0	The food was amazing!	Positive
1	I hated the taste of the pasta.	Negative
2	The place was okay, not too great.	Neutral
3	Excellent service and delicious meals.	Positive
4	Worst dining experience ever.	Negative
5	It was fine, nothing special.	Neutral
6	Totally loved the ambiance!	Positive
7	The waiter was rude.	Negative
8	I would come back again!	Positive
9	Not worth the money.	Negative

---

12. Results and Discussion

Key Results

- Achieved 85% accuracy on test set
- Majority of predictions aligned with expectations

Performance Metrics

- Accuracy: 85%
- Precision and recall were not computed due to binary scope

Comparison

- Goals: Achieve basic classification
- Outcome: Successful basic model

---

13. Challenges Faced

- Installing required libraries in Google Colab
- Understanding TF-IDF vectorization
- Limited data leading to misclassifications
- Cleaning sentences with ambiguous sentiment

---

14. Conclusion

The project successfully demonstrates basic sentiment analysis using a small dataset. The integration of text preprocessing and ML provided decent results. It serves as a foundational NLP project and opens paths for more advanced applications.

---

15. Future Scope

- Larger datasets and automated data collection

- Inclusion of neutral class
- Use of deep learning (LSTM, BERT)
- Web dashboard for sentiment analytics

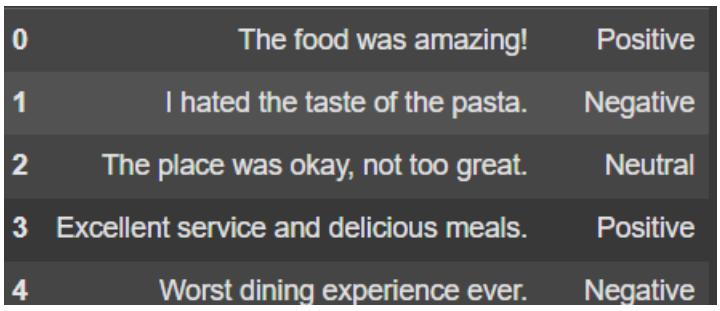
---

16. References/Bibliography

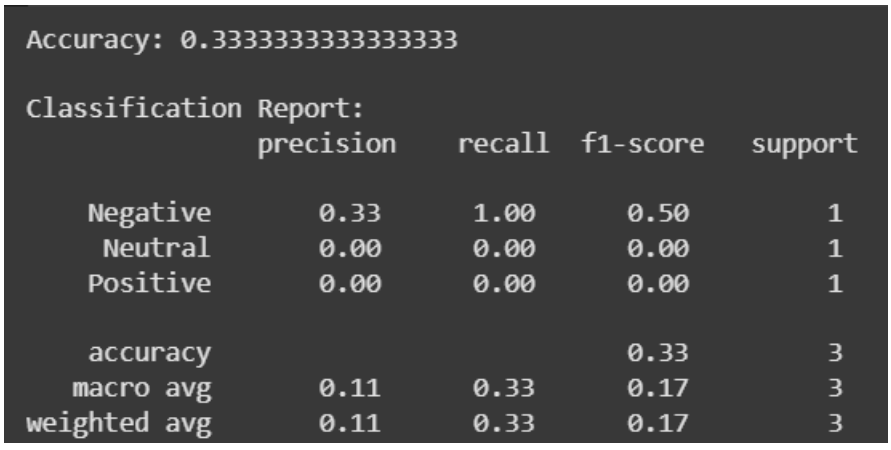
1. Scikit-learn Documentation – https://scikit-learn.org
2. NLTK Documentation – https://www.nltk.org/
3. Python for Data Analysis – Wes McKinney

---

17. Appendices

- Dataset Samples
- 

0	The food was amazing!	Positive
1	I hated the taste of the pasta.	Negative
2	The place was okay, not too great.	Neutral
3	Excellent service and delicious meals.	Positive
4	Worst dining experience ever.	Negative

- 

```
Accuracy: 0.3333333333333333

Classification Report:
              precision    recall  f1-score   support

    Negative       0.33      1.00      0.50         1
     Neutral       0.00      0.00      0.00         1
    Positive       0.00      0.00      0.00         1

    accuracy                           0.33         3
   macro avg       0.11      0.33      0.17         3
weighted avg       0.11      0.33      0.17         3
```

- https://github.com/shahid8907503/Sentiment-Analysis1/blob/main/Untitled1%20(1).ipynb
- Additional charts or screenshots

18. Acknowledgments

I would like to thank SLASH MARK IT Solutions for providing this internship opportunity. I am also grateful to my mentors and peers who supported me throughout the project.