YouTube Comments Sentiment Analysis

Using Machine Learning Approaches

IFTIKHAR ALI

August 16, 2025

Abstract

This project focuses on building a sentiment analysis model for YouTube comments. With the rapid increase in user-generated content, identifying sentiment polarity (positive, negative, or potentially neutral) is essential for applications such as content moderation, recommendation systems, and user engagement analysis. Using a dataset of over 18,000 YouTube comments, we apply preprocessing techniques such as tokenization, stopword removal, stemming, and lemmatization. Logistic Regression and Naïve Bayes models are trained on Bag-of-Words features, and their performances are compared. Logistic Regression consistently outperformed Naïve Bayes, achieving an accuracy of approximately 75%. This report documents the methodology, implementation, and evaluation of the models, concluding with insights for future improvements.

Contents

Chapter 1

Introduction

With the increasing popularity of online platforms like YouTube, analyzing user comments has become a critical task in understanding public opinion. Sentiment analysis is the process of determining whether a piece of text expresses a positive, negative, or neutral sentiment. This project aims to classify YouTube comments into sentiment categories using machine learning models.

Chapter 2

Problem Statement

The main objective of this project is to build a machine learning pipeline that automatically classifies YouTube comments into sentiment categories. This involves:

- Cleaning and preprocessing noisy text data.
- Applying natural language processing (NLP) techniques to prepare the data.
- Training machine learning models to classify sentiment.
- Evaluating the models with standard metrics.

Chapter 3

Dataset Description

The dataset used for this project is the **YouTube Comments Dataset** obtained from Kaggle. It consists of:

- **Total comments:** 18,408
- **Columns:**
  - *Comment*: The text of the YouTube comment.
  - *Sentiment*: The label assigned to the comment (Positive or Negative).
- **Missing values:** 44 missing comments, removed during preprocessing.

Note: Neutral comments were not included in this dataset. Future extensions could involve multi-class classification with positive, negative, and neutral sentiments.

Chapter 4

Methodology

The methodology of this project can be divided into the following phases:

4.1 Data Cleaning

- Removal of missing comments.
- Lowercasing text.
- Removal of punctuation, special characters, and extra whitespace.

4.2 Tokenization and Stopword Removal

- Tokenization using NLTK.
- Stopword removal (e.g., “the”, “is”, “and”).

4.3 Stemming and Lemmatization

- Stemming using Porter Stemmer.
- Lemmatization using spaCy.

4.4 Feature Extraction

Bag-of-Words representation using scikit-learn’s CountVectorizer with a vocabulary size of 5000.

4.5 Model Training

Two models were trained:

1. **Logistic Regression**
2. **Naïve Bayes (Multinomial)**

Chapter 5

Results and Evaluation

The models were evaluated on accuracy, precision, recall, and F1-score.

5.1 Model Accuracy

Model	Stemmed Accuracy	Lemmatized Accuracy
Logistic Regression	0.752	0.753
Naïve Bayes	0.696	0.695

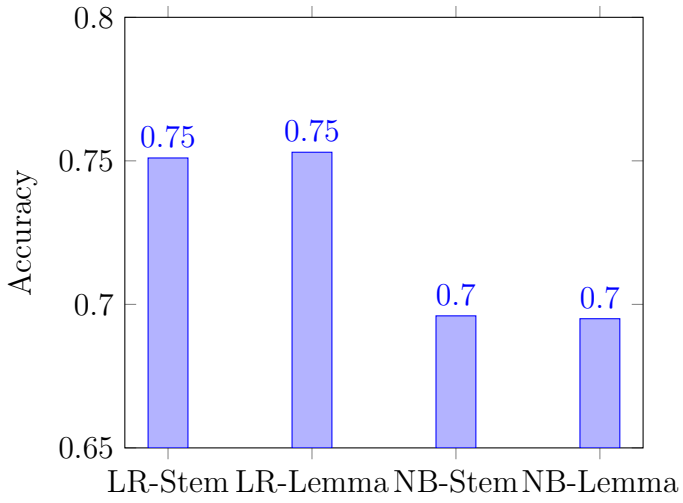
Table 5.1: Accuracy comparison across models and preprocessing methods.

5.2 Classification Metrics

Model	Precision	Recall	F1-score	Accuracy
Logistic Regression (Lemma)	0.75	0.74	0.74	0.753
Naïve Bayes (Lemma)	0.70	0.68	0.69	0.695

Table 5.2: Evaluation metrics for best-performing models.

5.3 Accuracy Comparison Chart



Chapter 6

Discussion

From the evaluation results, it is evident that Logistic Regression performs better than Naïve Bayes in this task. Lemmatization slightly improves the performance compared to stemming. The overall accuracy is around 75%, which indicates that the model is able to correctly classify most comments. However, handling neutral comments and using advanced models like transformers (e.g., BERT) could further improve performance.

Chapter 7

Conclusion

This project successfully demonstrated the application of NLP and machine learning for sentiment analysis of YouTube comments. Through preprocessing, feature extraction, and model training, we observed that Logistic Regression outperforms Naïve Bayes with an accuracy of 75%. While this is a promising result, future work could involve:

- Including neutral comments for multi-class classification.
- Using word embeddings (Word2Vec, GloVe).
- Applying deep learning models such as LSTMs or transformers.