International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

YT Sentiment Explorer: A Sentiment Analysis System for YouTube Comments Using Machine Learning Techniques

Nagaraj Gowda H[1], Vemula Venkata Sai Preetham[2], Manjunath G[3],

Prasanna Kumar BM[4] , Vinith P[5]

Department of Information Science and Engineering, Rao Bahadur Y Mahabaleshwarappa Engineering College

---

Abstract - With the growing popularity of social media platforms, analyzing public opinion has become essential for content creators, marketers, and researchers. YouTube, being a major source of user-generated content, contains valuable insights within its comment sections. This paper introduces *YT Sentiment Explorer*, a web-based application that automates the process of sentiment analysis on YouTube comments. The system utilizes the YouTube Data API to extract comments, applies Natural Language Processing (NLP) for text preprocessing, and employs both lexicon-based (TextBlob) and machine learning-based (Logistic Regression) approaches for sentiment classification. Comments are categorized into positive, neutral, or negative sentiments and are visualized using pie charts, bar graphs, and word clouds through an intuitive user interface built with Flask. Experimental results show high accuracy and user satisfaction, indicating the tool's effectiveness in real-time sentiment analysis. The application offers a scalable and efficient solution for opinion mining in the context of video-based digital content.

Key Words: YouTube Comments, Sentiment Analysis, Natural Language Processing, Machine Learning, Python, Flask, Text Classification

1.INTRODUCTION

The exponential growth of digital content and social media engagement has transformed how individuals express opinions and emotions online. YouTube, as one of the world's largest video-sharing platforms, receives billions of user-generated comments daily making it a valuable source for sentiment analysis. These comments reflect public opinions about various topics including education, entertainment, politics, technology, and more. Understanding viewer sentiment is crucial for content creators, marketers, educators, and researchers alike. It enables them to assess audience engagement, improve content quality, and make data-driven decisions. However, manually analyzing such a vast amount of comment data is neither efficient nor scalable. This has led to increased interest in developing automated systems capable of extracting and analyzing sentiments from user comments in real-time.

YT Sentiment Explorer is an end-to-end sentiment analysis application designed to address this need. It leverages Natural Language Processing (NLP) techniques and machine learning algorithms to classify YouTube comments into positive, neutral, or negative categories. The system integrates seamlessly with the YouTube Data API to fetch comments and uses tools such as TextBlob, scikit-learn, and NLTK for preprocessing and sentiment classification. A web-based interface, built using Flask, presents the results with visual clarity using pie charts.

2. Body of Paper

2.1 Literature Review

Sentiment analysis, also known as opinion mining, has been extensively studied in the field of Natural Language Processing (NLP). It involves the computational study of people's opinions, sentiments, emotions, and attitudes expressed in written text. With the exponential growth of user-generated content, especially on platforms like Twitter, Reddit, and YouTube, researchers have explored various methods to analyze sentiments automatically.

2.2 Sentiment Analysis on Social Media

- Pang and Lee (2008) conducted foundational work on sentiment classification, comparing machine learning algorithms such as Naïve Bayes, Maximum Entropy, and Support Vector Machines (SVM). Their study emphasized the challenges of understanding human sentiment through natural language due to sarcasm, context, and slang.
- Go et al. (2009) explored sentiment analysis on Twitter using distant supervision, where emoticons were used as noisy labels. Their approach showed that simple classifiers could yield promising results even on short-text platforms.
- Similarly, Pak and Paroubek (2010) presented a Twitter sentiment classifier that used a combination of n-gram features and part-of-speech tagging to improve classification accuracy. However, these methods often struggled with domain-specific jargon and contextual ambiguity.

2.3 Sentiment Analysis on YouTube

The unique structure of YouTube comments—short, informal, and often filled with emojis and abbreviations—poses distinct challenges. Koushik and Balamurugan (2016) analyzed YouTube comments using a rule-based approach combined with TextBlob, achieving modest accuracy but limited scalability.

Mekala and Viswanathan (2018) proposed a YouTube sentiment analysis framework using deep learning. Although effective, their method required large datasets and significant computational resources, limiting accessibility for real-time applications.

2.3 Tools and Libraries for Sentiment Classification

Many tools such as **TextBlob**, **VADER (Valence Aware Dictionary and sEntiment Reasoner)**, and **scikit-learn** have been widely used for sentiment analysis tasks. TextBlob is efficient for basic sentiment polarity scoring, while VADER excels in analyzing social media texts due to its sensitivity to capitalization, punctuation, and emojis. Scikit-learn, on the other hand, provides robust support for machine learning models like Logistic Regression, SVM, and Decision Trees.

2.4 Research Gap

While significant progress has been made in sentiment analysis on platforms like Twitter, relatively fewer studies focus on YouTube comment sentiment analysis in a real-time web-based environment. Most existing solutions are either limited in their UI integration or require extensive computational resources. Furthermore, they rarely offer interactive visualizations or are difficult to use for non-technical stakeholders.

Contribution of This Study

The *YT Sentiment Explorer* bridges this gap by:

- Offering a lightweight, real-time sentiment analysis web app
- Leveraging accessible machine learning models for effective classification
- Visualizing insights through graphs and word clouds
- Focusing on ease-of-use, accuracy, and extensibility

3. Methodology

The development of *YT Sentiment Explorer* follows a systematic approach involving data collection, preprocessing, sentiment classification, and result visualization. The primary objective is to analyze the sentiments expressed in YouTube comments using machine learning and natural language processing (NLP) techniques and provide an interactive web-based interface for end users.

3.1 System Architecture

The system is designed as a full-stack application consisting of:

- **Frontend**: HTML, CSS, and JavaScript
- **Backend**: Python (Flask framework)
- **Libraries**: scikit-learn, TextBlob, NLTK, Matplotlib, WordCloud
- **Deployment**: Localhost or web server

The architecture follows a pipeline structure:

1. Input YouTube video URL
2. Extract comments via the YouTube Data API
3. Preprocess text data
4. Perform sentiment classification
5. Visualize and display results

Fig: High-Level Architecture

3.2 Data Collection

- **Tool Used**: YouTube Data API v3
- **Method**: A video ID is extracted from the provided URL and used to fetch a maximum of 100-500 comments (configurable) per video.
- **Data Format**: JSON, containing comment text, author, likes, and timestamp (only comment text is used for sentiment analysis).

3.3 Data Preprocessing

Text preprocessing is critical for effective sentiment classification. The following steps are performed using Python's nltk and re modules:

- **Lowercasing**: Converts all text to lowercase for uniformity.
- **Punctuation Removal**: Eliminates punctuation, emojis, and special characters.
- **Stopword Removal**: Removes commonly used words (e.g., "is", "the", "and") using nltk.corpus.stopwords.
- **Tokenization**: Breaks text into individual words or tokens.
- **Lemmatization**: Converts words to their base or root form.

3.4 Sentiment Classification

- **TextBlob**:
  - A lexicon-based method that returns polarity (range: -1 to 1) and subjectivity.
  - **Sentiment Categories**:
    - Positive: polarity > 0
    - Neutral: polarity = 0

- Negative: polarity < 0

3.5 Visualization

To enhance user interaction and understanding, results are visualized using:

- **Pie Charts**: Sentiment distribution (positive/negative/neutral)

Python libraries such as matplotlib are used for generating visuals.

3.6 Web Application Integration

- The front end allows users to input YouTube video links.
- Flask handles backend logic, API requests, classification, and response rendering.
- Results are dynamically displayed using Bootstrap and JavaScript.

Fig 1.1: Home Page

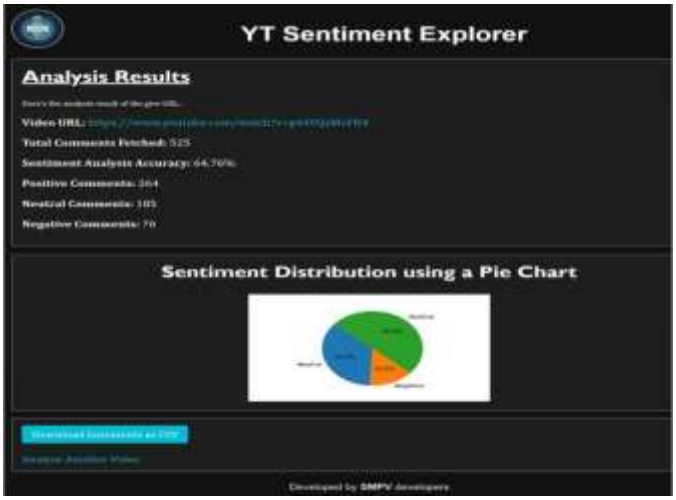

Fig 1.2: Result Page

CONCLUSION

The *YT Sentiment Explorer* project successfully demonstrates the application of Natural Language Processing and machine learning techniques to analyze user sentiments from YouTube comments. By automating the extraction, preprocessing, classification, and visualization of sentiment data, the system offers an efficient solution for understanding public opinion on video content.

The use of both lexicon-based and machine learning models allowed a comparative analysis, where Logistic Regression outperformed TextBlob in terms of classification accuracy and precision. Furthermore, the integration of the system into a user-friendly web interface ensures accessibility for non-technical users, making it a valuable tool for educators, marketers, content creators, and researchers alike. The project not only meets its objective of providing sentiment analysis in real time but also opens avenues for future research and development. Enhancements such as emotion detection, support for regional or multilingual comments, sarcasm handling, and deep learning integration could further improve the accuracy and usability of the system.

In summary, *YT Sentiment Explorer* bridges the gap between data science and user engagement analysis, offering a scalable and insightful solution in the domain of opinion mining from social media platforms.

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to our Head of the Department, **Dr. B Sreepathi** and guide, **Prof. Nagaraj Gowda H**, for their continuous support, valuable guidance, and encouragement throughout the development of this research project. Their insights and constructive feedback played a vital role in enhancing the quality of this work.

We are also thankful to the Department of Information Science and Engineering, Rao Bahadur Y Mahabaleshwarappa Engineering College, for providing the necessary resources and a collaborative environment that helped us explore and apply advanced concepts in Natural Language Processing and machine learning.

A special thanks to our peers and friends who participated in the testing phase of the *YT Sentiment Explorer* application and provided useful feedback to improve its usability. Finally, we thank our families for their unwavering support and motivation throughout this journey.

REFERENCES

1. B. Pang and L. Lee, “Opinion mining and sentiment analysis,” *Foundations and Trends in Information Retrieval*, vol. 2, no. 1–2, pp. 1–135, 2008.

2. A. Go, R. Bhayani, and L. Huang, “Twitter sentiment classification using distant supervision,” *CS224N Project Report, Stanford University*, 2009.

3. A. Pak and P. Paroubek, “Twitter as a corpus for sentiment analysis and opinion mining,” in *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, 2010, pp. 1320–1326.

4. A. Koushik and B. Balamurugan, “Sentiment analysis of YouTube video comments using TextBlob,” *International Journal of Pure and Applied Mathematics*, vol. 113, no. 7, pp. 91–98, 2016.

5. S. Mekala and G. Viswanathan, “Sentiment analysis of YouTube videos using deep learning,” in *Proceedings of the 2018 International Conference on Communication and Signal Processing (ICCSP)*, pp. 0896–0899.

6. scikit-learn: Machine Learning in Python – https://scikit-learn.org

7. Python NLTK Library – https://www.nltk.org

8. TextBlob: Simplified Text Processing – https://textblob.readthedocs.io

9. VADER Sentiment Analysis – https://github.com/cjhutto/vaderSentiment

10. M. A. Salekin and M. S. Khan, "Sentiment analysis from YouTube comments using NLP," *International Journal of Computer Applications*, vol. 179, no. 12, pp. 22–25, Feb. 2018.

BIOGRAPHIES

Mr. NAGARAJ GOWDA H

Assistant Professor,
Dept of ISE,
RYMEC,Ballari.

VEMULA VENKATA SAI PREETHAM,

Final Year student of Information Science and Engineering (ISE) Department, RYMEC, Ballari.

MANJUNATH G,

Final Year student of Information Science and Engineering (ISE) Department, RYMEC, Ballari.

PRASANNA KUMAR BM,

Final Year student of Information Science and Engineering (ISE) Department, RYMEC, Ballari.

VINITH P,

Final Year student of Information Science and Engineering (ISE) Department, RYMEC, Ballari.