YouTube Sentiment Analysis Project

A complete data analytics project that collects YouTube video comments, cleans the data, and performs sentiment analysis to understand viewer opinions and emotions.

📋 Table of Contents

🎯 Project Overview

This project demonstrates a complete data analytics pipeline:

1. **Data Collection**: Scrapes comments from YouTube videos using the YouTube Data API
2. **Data Cleaning**: Preprocesses and cleans the raw comment data
3. **Sentiment Analysis**: Analyzes the emotional tone of each comment (positive, negative, or neutral)
4. **Visualization**: Creates charts and graphs to visualize sentiment distribution

Project Goals

- Understand public opinion about specific YouTube videos
- Practice real-world data collection and cleaning
- Apply natural language processing (NLP) techniques
- Create meaningful visualizations from data

✨ Features

- ✅ Automated YouTube comment scraping
- ✅ Data cleaning and preprocessing
- ✅ Sentiment classification (Positive/Neutral/Negative)
- ✅ Statistical analysis of sentiment scores
- ✅ Interactive visualizations (pie charts, histograms)
- ✅ Export results to CSV files
- ✅ Identifies most positive and negative comments
- ✅ Handles replies and nested comments

🛠️ Technologies Used

Programming Language

- **Python 3.8+**

Libraries & Frameworks

Library	Purpose
`google-api-python-client`	YouTube Data API integration
`pandas`	Data manipulation and analysis
`vaderSentiment` or `textblob`	Sentiment analysis
`matplotlib`	Data visualization
`python-dotenv`	Environment variable management
`re` (regex)	Text cleaning and preprocessing

APIs

- **YouTube Data API v3** - For collecting video comments

📁 Project Structure

```
youtube-sentiment-analysis/
│
├── .env                          # API key storage (not committed to git)
├── README.md                     # Project documentation
│
├── youtube_scraper.py            # Step 1: Data collection script
├── data_cleaning.py              # Step 2: Data cleaning script
├── sentiment_analysis.py         # Step 3: Sentiment analysis script
│
├── youtube_comments_YYYYMMDD.csv     # Raw collected data
├── cleaned_comments_YYYYMMDD.csv     # Cleaned data
├── sentiment_results_YYYYMMDD.csv    # Final results with sentiment scores
└── sentiment_visualization_YYYYMMDD.png  # Generated charts
```

💾 Installation

Prerequisites

1. **Python 3.8 or higher** installed on your system
2. **Google Account** for YouTube API access
3. **VS Code** or any text editor
4. **pip** package manager

Install Required Libraries

```bash
pip install google-api-python-client
pip install pandas
pip install python-dotenv
pip install vaderSentiment
pip install matplotlib
```

Or install all at once:

```bash
pip install google-api-python-client pandas python-dotenv vaderSentiment matplotlib
```

🔧 Setup Instructions

Step 1: Get YouTube API Key

1. Go to Google Cloud Console
2. Create a new project (e.g., "YouTube-Sentiment-Analysis")
3. Enable **YouTube Data API v3**
4. Create credentials → API Key
5. Copy your API key

Step 2: Configure Environment Variables

1. Create a file named `.env` in your project folder
2. Add your API key:

```
YOUTUBE_API_KEY=your_api_key_here
```

⚠️ **Important**: Never share your API key publicly!

Step 3: Project Files

Create three Python files:

- `youtube_scraper.py`
- `data_cleaning.py`
- `sentiment_analysis.py`

Copy the provided code into each respective file.

📖 Usage Guide

Step 1: Collect YouTube Comments

```
bash

python youtube_scraper.py
```

What it does:

- Prompts you for a YouTube video URL
- Asks how many comments to collect
- Fetches comments and video metadata
- Saves raw data to CSV

Input Example:

```
👉 Video URL/ID: https://www.youtube.com/watch?v=dQw4w9WgXcQ
👉 Number of comments: 500
```

Output:

- `youtube_comments_YYYYMMDD_HHMMSS.csv`
- `video_info_YYYYMMDD_HHMMSS.csv`

Step 2: Clean the Data

```
bash

python data_cleaning.py
```

What it does:

- Automatically finds the latest comments file
- Removes duplicates and empty comments
- Cleans text (removes URLs, special characters, etc.)
- Saves cleaned data to CSV

Cleaning Operations:

- Convert to lowercase
- Remove URLs and emails
- Remove mentions (@username)
- Remove special characters
- Remove extra whitespace
- Filter out empty comments

Output:

- `cleaned_comments_YYYYMMDD_HHMMSS.csv`

Step 3: Analyze Sentiment

```
bash

python sentiment_analysis.py
```

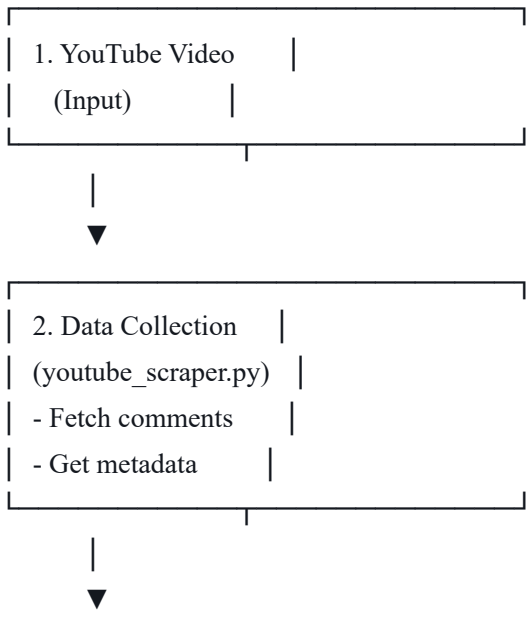
What it does:

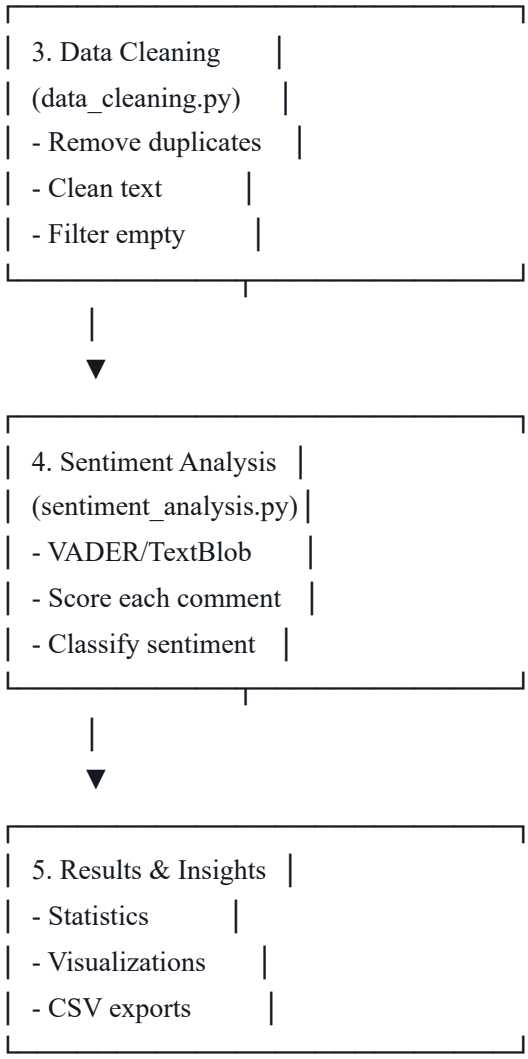
- Loads cleaned comments
- Analyzes sentiment for each comment
- Calculates statistics
- Creates visualizations
- Saves results

Output:

- `sentiment_results_YYYYMMDD_HHMMSS.csv`
- `sentiment_visualization_YYYYMMDD_HHMMSS.png`
- Console output with statistics

🔄 Project Workflow





📊 Results & Insights

Sample Output

```
📊 SENTIMENT ANALYSIS RESULTS:
============================================================

📈 Overall Sentiment Distribution:
  Positive: 245 (49.0%)
  Neutral:  156 (31.2%)
  Negative:  99 (19.8%)

📊 Sentiment Score Statistics:
  Average score: 0.234
  Most positive: 0.927
  Most negative: -0.873
```

😊 MOST POSITIVE COMMENT:
Score: 0.927
Text: This is absolutely amazing! Best video I've ever watched...

😞 MOST NEGATIVE COMMENT:
Score: -0.873
Text: This is terrible and completely misleading...

Visualizations

The project generates two types of charts:

1. **Pie Chart**: Shows the distribution of positive, neutral, and negative comments
2. **Histogram**: Displays the distribution of sentiment scores across all comments

---

🐛 Troubleshooting

Common Issues & Solutions

Issue 1: "API key not valid"

Solution:

- Verify your API key in `.env` file
- Ensure YouTube Data API v3 is enabled in Google Cloud Console
- Check for typos in the key

Issue 2: "quotaExceeded"

Solution:

- YouTube API has daily limits (10,000 quota units/day)
- Wait until the next day (resets at midnight Pacific Time)
- Request quota increase in Google Cloud Console

Issue 3: "commentsDisabled"

Solution:

- The video you selected has comments disabled
- Try a different video

Issue 4: "No module named 'vaderSentiment'"

Solution:

```
bash

pip install vaderSentiment
```

Issue 5: Charts not displaying

Solution:

```
bash

pip install matplotlib
```

If using WSL or headless environment, charts will be saved as PNG files.

🔮 Future Enhancements

Potential improvements for this project:

- [ ] Add support for multiple videos in one run
- [ ] Implement aspect-based sentiment analysis
- [ ] Add word clouds for positive/negative comments
- [ ] Track sentiment over time (temporal analysis)
- [ ] Add emotion detection (joy, anger, sadness, etc.)
- [ ] Create an interactive dashboard using Streamlit or Dash
- [ ] Add language detection and multi-language support
- [ ] Implement topic modeling (LDA) to find common themes
- [ ] Add comparison between multiple videos
- [ ] Export results to PDF report
- [ ] Add machine learning model training option
- [ ] Create REST API for the analysis pipeline

🎓 Learning Outcomes

By completing this project, you will learn:

- ✅ How to use REST APIs (YouTube Data API)

- ✅ Data collection and web scraping techniques
- ✅ Data cleaning and preprocessing
- ✅ Natural Language Processing (NLP) basics
- ✅ Sentiment analysis with VADER/TextBlob
- ✅ Data visualization with matplotlib
- ✅ Working with pandas DataFrames
- ✅ Environment variable management
- ✅ Project organization and documentation

📝 API Quota Information

YouTube Data API v3 Quota:

- Daily limit: 10,000 units
- Search query: 100 units
- Comment list: 1 unit per request
- Video details: 1 unit

Estimated capacity:

- ~1,000-10,000 comments per day depending on operations

🔐 Security Best Practices

1. **Never commit `.env` file** to version control
2. Add `.env` to `.gitignore`
3. Regenerate API keys if accidentally exposed
4. Use API key restrictions in Google Cloud Console
5. Don't hardcode API keys in source code

.gitignore Example

```
# Environment variables
.env

# Python cache
```

```
__pycache__/
*.pyc

# Data files (optional)
*.csv
*.png

# IDE
.vscode/
.idea/
```

📚 References & Resources

Documentation

- YouTube Data API Documentation
- VADER Sentiment Analysis
- Pandas Documentation
- Matplotlib Documentation

Tutorials

- Python for Data Analysis
- NLP with Python
- Sentiment Analysis Guide

👨‍💻 Author

Your Name

- Project: YouTube Sentiment Analysis
- Date: February 2025
- Purpose: Data Analytics Portfolio Project

📄 License

This project is open source and available for educational purposes.

🙏 Acknowledgments

- Google for YouTube Data API
- VADER Sentiment Analysis developers
- Python community for excellent libraries
- All contributors to open-source tools used

📞 Contact & Support

If you have questions or need help:

1. Check the Troubleshooting section
2. Review YouTube API documentation
3. Search for similar issues on Stack Overflow

⭐ Project Status

Status: Complete ✅

Last Updated: February 2025

Version: 1.0.0

🎯 Quick Start Checklist

- [ ] Python 3.8+ installed
- [ ] Created Google Cloud project
- [ ] Enabled YouTube Data API v3
- [ ] Got API key
- [ ] Created `.env` file with API key
- [ ] Installed all required libraries
- [ ] Created all three Python scripts

- [ ] Ran data collection script
- [ ] Ran data cleaning script
- [ ] Ran sentiment analysis script
- [ ] Reviewed results and visualizations

Congratulations! Your sentiment analysis project is complete! 🎉

This documentation was created as part of a data analytics portfolio project to demonstrate skills in data collection, cleaning, analysis, and visualization.