☰   Sitina-Celine / **Sentiment_Analysis_project4**

<> **Code**   ⊙ Issues   ⇄ Pull requests   ▷ Actions   ⊞ Projects   📖 Wiki   ⊘ Security   📈 Insights   ⚙ Settin

☆ 0 stars   ⑂ 0 forks   ⊙ 0 watching   ⑂ Branches   ⋏ Activity
⊗ Tags

🌐 Public repository

⑂   ⑂ **5 Branches**   ⊘ **0 Tags**   ⑂   ⊗ | 🔍 Go to file   t | Go to file   Add file +   Code   ···

🐷 **wachingawathiri** Updated presentation slides		ffbad48 · 23 minutes ago	🕓
📄 Dockerfile	docker file for app containerization	6 hours ago	
📄 README.md	Update README.md	2 hours ago	
📄 SENTIMENT_ANALYSIS_REP...	Refactor code structure for improve...	4 hours ago	
📄 Sentiment Analysis.pdf	Updated presentation slides	1 hour ago	
📄 index.ipynb	Resolved README.md conflict by re...	4 hours ago	
📄 judge-1377884607_tweet_pr...	Sentiment analysis modelling	yesterday	
📄 sentimentanalyzer.py	Resolved README.md conflict by re...	4 hours ago	

Aspect-Based Sentiment Analysis of Tweets Directed at Brands and Products using Natural Language Processing.

Team Members

- Celine Sitina
- Gabriel Tenesi
- Wesley Kipsang

📖 README   ✏ ☰

Business Understanding

Business Overview

Social media platforms like Twitter are where people openly share their thoughts, complaints, and praise about products and brands. These conversations show how customers truly feel and what matters to them. For big companies like Apple and Google, understanding this feedback is key to improving products and maintaining a strong brand image.

According to Aga Khan University (2022), https://ecommons.aku.edu/cgi/viewcontent.cgi?article=1069&context=etd_ke_gsmc_ma-digjour, analyzing social media discussions gives organizations valuable insights into consumer attitudes and market trends that support smarter business decisions. Listening to what people say online helps companies respond faster, build trust, and stay connected to their customers.

Problem Statement

Apple and Google continuously monitor customer satisfaction to stay ahead in the technology market. However, given the massive volume and speed of data generated on Twitter, manual tracking of sentiment is impractical. Without automated systems, valuable insights into customer satisfaction, emerging issues, and product perception may be overlooked.

This project aims to address this challenge by developing a machine learning model capable of classifying tweets related to Apple and Google as positive, negative, or neutral. The outcome will support organizations in understanding real-time consumer opinions, measuring brand perception, and identifying areas for improvement based on public feedback.

Business Objective

Main objective:

To develop an NLP-based sentiment analysis model that automatically classifies tweets about Apple and Google into positive, negative, or neutral categories.

Specific objectives:

1. To explore and clean the tweet dataset, handling missing values, duplicates, and irrelevant characters.

2. To preprocess textual data through tokenization, stopword removal, and lemmatization.

3. To convert cleaned text into numerical features using appropriate vectorization techniques such as TF-IDF or Word2Vec.

4. To train and evaluate multiple classification algorithms (e.g., Logistic Regression, Naive Bayes, SVM) to identify the best-performing model.

5. To interpret and visualize model predictions, identifying which features most influence positive and negative sentiment.

6. To provide actionable insights that can guide Apple and Google in improving customer experience and brand perception.

Research Questions

1. How can the dataset be explored and cleaned to ensure data quality and reliability for sentiment analysis?

2. What preprocessing techniques are most effective for preparing Twitter text data for modeling?

3. Which text vectorization method (e.g., TF-IDF) produces better numerical representations for tweet classification?

4. Which classification algorithms yield the highest accuracy and robustness in predicting tweet sentiment?

5. Which textual features (words, phrases, or hashtags) most strongly influence model predictions of sentiment?

6. How can the resulting sentiment insights be applied by Apple and Google to improve customer satisfaction and brand reputation?

Success Criteria

Model Performance: Achieve at least 85% classification accuracy and a macro F1-score ≥ 0.80 across all sentiment classes (positive, negative, neutral).

Model Interpretability: Clearly explain which features (words, hashtags, expressions) most affect sentiment predictions using tools such as LIME or SHAP.

Business Value: Provide insights that help Apple and Google understand customer sentiment, identify common issues, and track brand reputation effectively

Data Understanding

Data overview

The dataset contains 9,093 tweets collected from crowdFlower, with the goal of identifying whether the emotion in a tweet is directed at a brand or product, and if so, what sentiment it carries. It includes 3 columns,

- tweet_text: The raw text of the tweet, expressing user opinions or emotions.
- emotion_in_tweet_is_directed_at: The specific brand or product the emotion is directed at (if identified)
- is_there_an_emotion_directed_at_a_brand_or_product: Indicates whether the tweet expresses emotion toward a brand/product

data characteristics

- Number of rows: 9,093
- Number of columns: 3

- Data types: All columns are of type object (textual).
- Target variable: is_there_an_emotion_directed_at_a_brand_or_product.
- Feature variable: tweet_text.
- Filtering scope: Tweets directed at Apple or Google will be selected for analysis.

Data Analysis

Our target variable includes various sentiment labels such as Positive emotion, Negative emotion, and No emotion toward brand or product.

From the distribution of sentiments, both brands receive a mix of sentiments, with 'I can't tell' being the least. The positive sentiments dominate in both brands, but the Apple brand receives more positive sentiments compared to the Google brand.

Model Development

The logistic Regression is taken as the baseline model. It has an overall accuracy of 98%. looking at other metrices the positive seems to be perfoming better while the other classes have a poor perfomance.

Logistic Regression gave strong baseline results, but SVM which achieved a 98% accuracy is often better at handling complex, high-dimensional text data like TF-IDF. It can pick up on subtle differences between sentiments—especially Neutral and Negative, so it's a good choice to compare against the baseline.

The Naive Bayes model achieved an accuracy of 93.97% with a macro F1-score of 0.94, with a strong and balanced performance across all classes. It achieved high precision for Negative 0.96 and Positive 0.98 sentiments, and perfectly recalled the Neutral class 1.00. However, it performed slightly below SVM and Logistic Regression.

Deployment

A sentiment analyzer was deployed to predict the sentiment on tweets about apple and google products as either positive, Negative, or Neutral. This is the link to the Sentiment Abnalyzer streamlit app.

https://sentiment-analyzer-3zs1.onrender.com/

Conclusion

- This project explored different models for sentiment analysis, including VADER, Naive Bayes, Logistic Regression, and SVM. The SVM model performed best, achieving an accuracy of 0.9812, showing strong ability to understand patterns in the text. Overall, machine learning models, especially those using TF-IDF features, performed much better than rule-based approaches like VADER.

Recommendations

- Use SVM as the main model for sentiment classification as it performed better than other models.

- Logistic Regression performed strongly with an accuracy of 0.98 and can be used alongside SVM when transparency and stakeholder understanding are priorities

Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

Contributors 2

wachingawathiri

Sitina-Celine celine sitina

Languages

● **Jupyter Notebook** 99.6%   ● **Other** 0.4%

Suggested workflows
Based on your tech stack

| **Pylint** | Configure |
| Lint a Python application with pylint. | |

| **SLSA Generic generator** | Configure |
| Generate SLSA3 provenance for your existing release workflows | |

| **Docker image** | Configure |
| Build a Docker image to deploy, run, or push to a registry. | |

[More workflows](#)   Dismiss suggestions