Samu3INM /
Group-2-Phase-4-Project

Code Issues Pull requests 1 Actions Projects Wiki Security Insights Se

0 stars 4 forks 1 watching Branches Activity
Tags

Public repository

1 Branch 0 Tags Go to file t Go to file + Add file Code ...

| | | |
|---|---|---|
| Samu3INM Merge pull request #5 from Jayneluck/main | | bc1fb1d · 2 days ago |
| GROUP_2_PHASE_4_NON-TE... | Add files via upload | 4 days ago |
| Group_2_(Phase_4_Project).ip... | Final notebook | 4 days ago |
| README.md | Updated README with latest content | 4 days ago |

README

# Sentiment Analysis of Apple vs Google

# Business Understanding

The dataset presents an analysis of different emotions drawn out of tweets about either an Apple or Google product.The Businesses can better understand client impressions in real time by classifying tweet sentiment (positive, negative, and neutral) using natural language processing (NLP).

#**Problem Statement** With sizable and devoted user bases, Apple and Google are significant actors in the tech sector. Maintaining competitiveness, controlling brand reputation, and enhancing services all depend on knowing how customers feel about their products.

It is not scalable to manually track public sentiment given the hundreds of tweets regarding their products that are posted every day. An automated method is required in order to categorize tweets about Apple and Google into favorable, negative, and neutral sentiments.

# Objectives

1. Build a model that can rate the sentiment of a Tweet based on its content. Using tweet content to base predictions.

2. Building a binary classifier to distinguish between positive and negative tweets.
3. Building a multiclass classification.
4. Evaluating model performance using evaluation metrics like Accuracy and Precision for binary and Multiclass: Macro F1-score, Weighted Accuracy, and per-class performance

# Data Understanding

# Data Cleaning

# EDA

Overall,

Most tweets are neutral(no emotion) Far tweets express negative sentiment. This imbalance can make it harder for a model to learn how to correctly identify minority classes(especially negative).

The data is largely dominated by neutral sentiments, especially from the unknown brand group.

Among known brands, iPad and Apple have the strongest positive sentiment.

There's no strong negative trend for any specific brand, which could be good news for all involved.

This plot visualizes the frequency of words found in tweets with negative sentiments. Words such as SXSW, mention, ipad,quot and apple are larger which shows they are mentioned more frequently in negative contexts.

This plot shows that words like iphone, ipad, apple, Google, SXSW, RT and link appear bolder and larger, which indicates they are mentioned more often in positive contexts

## Text preprocessing

**Cleaning Text**

Removing mentions

Removing hashtags

Removing special characters and punctuation

Converting text to lowercase

Removing common words.

Reducing words to root form

## Feature Engineering

As the number of words in a tweet increases, the tweet length (likely measured in characters) also increases — and vice versa.

This is expected, but the correlation of 0.91 quantifies this relationship as very strong.

This suggests that either variable could be a good proxy for the other in modeling or exploratory data analysis.

# Modeling

## Label Encoding

## Binary Classification

## Logistic Regression Model

### Class Imbalance

Accuracy: 86% looks good at first glance. The model learns to be really good at classifying class 1

## Multinomial NaiveBayes

## Hyperparameter Tuning

### Confusion Matrix based on binary

The model performs quite well with high precision (few false positives) and high recall (few false negatives).

An accuracy of 85.7% is decent, but the F1 score is even more informative in this context at 91.4%, which shows a good balance between precision and recall.

The 36 false positives and 65 false negatives might be important depending on your use case — especially if missing positives (false negatives) is costly (e.g., in disease detection).

## NN-Models

### ANN Model (Sigmoid)

#### Multiclass Classification

**Class 0:** Precision: 1.00 When it predicts class 0, it's always correct.

Recall: 0.01 But it almost never predicts class 0 correctly.

F1-score: 0.02 Very poor overall performance for class 0.

**Class 1:** Precision: 0.65

Recall: 0.95 Model does a great job detecting class 1.

F1-score: 0.77 Strong performance.

**Class 2:** Precision: 0.69

Recall: 0.24 Model misses most class 2 instances.

F1-score: 0.36 Needs improvement.

**Class 1 dominates predictions.**

### Confusion Matrix based on Multiclass classification

The model performs very well on Class 1.

Class 0 is almost completely misclassified, with only 1 correct prediction.

Class 2 is mostly confused with Class 1, indicating a need to improve class separation, especially between Class 1 and Class 2.

#### Class Imbalance

**Class 0** is dominating in performance.

**Class 1** has the lowest recall the model struggles more to identify actual class 1s.

**Class 2** is performing moderately well.

### ANN Model (Softmax)

### LSTM Model

### Hyperparameter Tuning (Randomised Search)

# Evaluation

1. **Logistic Regression**: Achieved a reasonable accuracy, but performance might be limited by class imbalance. SMOTE improved the model's ability to handle the imbalanced classes.

2. **Multinomial Naive Bayes**: Performance before and after hyperparameter tuning needs comparison. This is the best model's accuracy and F1-score, on binary classification.

3. **ANN (Sigmoid)**: Performance metrics (accuracy, precision, recall, F1-score) provide a good overview of the model's classification capabilities.

4. **ANN (Softmax)**: The ANN with Softmax activation performs multiclass classification, and its performance is measured using accuracy, macro-averaged precision, recall, and F1-score. The classification report provides a detailed breakdown of its performance per class.
5. **LSTM:** The LSTM model utilizes sequential data processing. Performance metrics and the classification report highlight its ability to classify sentiments based on sequential features. This is the best model on multiclass classification.

# Conclusion

•Neutral sentiment is dominant in the dataset, indicating a lack of strong opinions, which might imply a need for more engaging or polarizing content from brands.

•Positive sentiment for well-established brands like Apple and iPad is a good sign, suggesting strong brand loyalty and customer satisfaction.

•The absence of significant negative sentiment across all brands could indicate a generally favorable perception, but the dataset's neutrality suggests that there may be a gap in passionate customer advocacy or brand differentiation.

•The word frequency plot offers valuable insights into both sentiment trends and the relevance of specific words, which can guide further analysis or marketing efforts.

# Recommendations

•**Focus on unknown brands:** Analyze why sentiment is mostly neutral to improve brand awareness and engagement.

•**Leverage positive sentiment for Apple and iPad:** Highlight strengths from positive tweets to boost loyalty and reach.

•**Monitor for emerging negative trends:** Track sentiment regularly to catch and respond to potential issues early.

### Releases

No releases published
Create a new release

### Packages

No packages published
Publish your first package

## Contributors 4

- Samu3lNM
- deeogeto
- Jayneluck
- Severino36

## Languages

● Jupyter Notebook 100.0%