Julez89 /
dsc-phase-4-project

Code Issues Pull requests Actions Projects Wiki Security Insights

0 stars 0 forks 1 watching Activity

Public repository

main

Branches Tags

Julez89 ... 1 hour ago

View code

README.md

Phase 4 Project Description

Project Overview

For this project, I tried different classification methods to predict the sentiment of Tweets for Google and Apple products. My final model uses a random forest classifier after addressing the class imbalance by oversampling the minority class.

Business Problem & Stakeholder

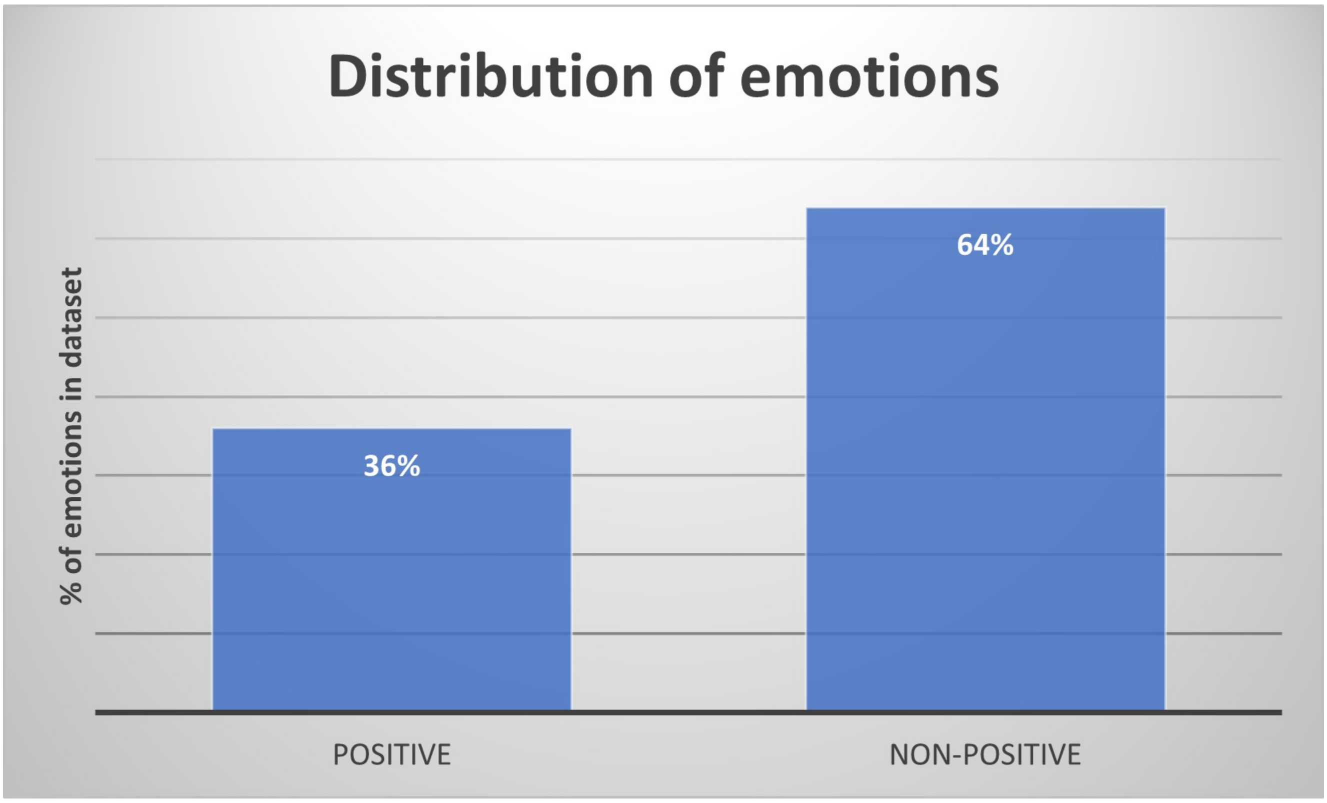
A marketing firm who has Apple and Goolge as their clients wants to be able to give better advice on which products or features to spend more marketing budget on and which marketing activities seem to work better than others. At the moment, they have employees who manually read through Tweets which is an activity that is time consuming and introduces bias. Questions that we want to answers are:

What are the most frequently used positive words about Apple and Google products? Will I be able to correctly identify a positive Tweet?

The Data

This project uses a data set with around 9000 Tweets about Apple and Google products during a SXSW conference. The dataset includes the Tweet, the Brand or Product and the Emotion. Only a third of the Tweets had a brand/product assigned at first which I fixed in my data cleaning steps. Even though we have three possibilities in the emotions (positive, negative and neutral), I will make this a binary classification and cluster it in positive vs. non-positive emotions.

Regarding my target variable (emotion), we are dealing with class imbalance. 64% of the data set is non-positive while 36% is positive - I will address this by oversampling the minority class.



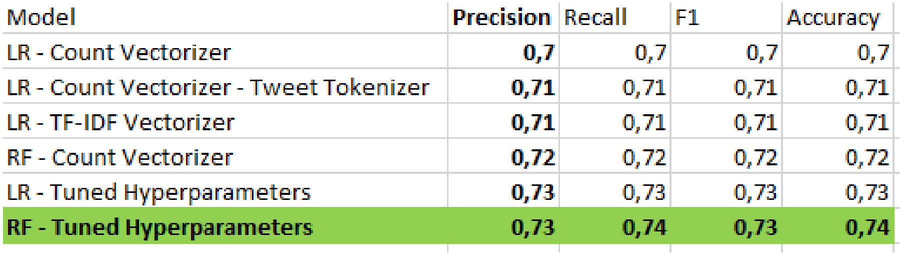
Approach

This task is a classification problem because my target variable is a categorical variable. It is either positive or non-positive for a tweet. I used a machine learning approach to solve the classification task. Machine learning algorithms are designed to automatically identify patterns and relationships within data, allowing them to detect complex relationships that may not be immediately apparent through manual analysis. Also, machine learning algorithms can improve over time as they are trained on more data, providing an ongoing opportunity to refine and improve the accuracy of the classification model. Finally, they are able to handle large amounts of data efficiently and quickly, making them an ideal choice for tasks involving large datasets. I am using an iterative approach to try to finding the best model for my predictions. In summary, I took the following steps:

- Loading and inspecting the data
- Cleaning of data (brand/product, emotions), dropping duplicate and missing values
- Exploratory data analysis with basic preprocessing using tokenization, removing stopwords and checking most common words
- Prepare data for modelling (removing stop words, train - test split)
- Start with a vanilla, simple logistic regression model
- Try different vectorizer and sampling approaches
- Try a more complex model with random forest classification and tune it
- Evaluate the final model (random forest, count vectorizer and oversampling).

Modeling

For my customer, the main priority is to be able to identify positive tweets, so that they can react to those positive experiences. Therefore, I'm prioritizing precision and trying to minimize false negatives. For my business objective of understanding and leveraging positive sentiment, we want to ensure that the insights derived from the model are reliable and genuinely reflect positive sentiments. High precision helps you achieve this by reducing the inclusion of unrelated or negative tweets in the positive sentiment category. My preparation steps included the preprocessing of data, removing stopwords, tokenization and random oversampling. To follow an iterative approach, I am starting with a very simple, vanilla logistic regression model, oversampling and count vectorizer which already performed ok (weighted avg precision= 0.70). As next model, I'm adding TweetTokenizer and saw a slight improvement (weighted avg precision =0.71). The change to use TfIDF vectorizer instead of count vectorizer did not change anything. Next, I am trying a Random Forest model with the standard hyperparameters. This model already improved (precision=0.72). Next, I'm using GridSearchCV for both the logistic regression and the random forest model to decide which model works best with tuned hyperparameters. Even though there is only a small difference, I'm choosing the random forest model as it has a slightly higher accuracy while having the same precision.



Model	**Precision**	Recall	F1	Accuracy
LR - Count Vectorizer	**0,7**	0,7	0,7	0,7
LR - Count Vectorizer - Tweet Tokenizer	**0,71**	0,71	0,71	0,71
LR - TF-IDF Vectorizer	**0,71**	0,71	0,71	0,71
RF - Count Vectorizer	**0,72**	0,72	0,72	0,72
LR - Tuned Hyperparameters	**0,73**	0,73	0,73	0,73
RF - Tuned Hyperparameters	**0,73**	**0,74**	**0,73**	**0,74**

Evaluation

My final model is a tuned random forest model with CountVectorizer, random oversampling and specified parameters such as minimum sample leafs, minimum sample split and the number of estimators. My final model has a weighted average precision of .73. When my model predicts positive cases, it is correct 76% of the time which is acceptable. A limitation is that the model is not as great in predicting non-positive sentiments. We can improve the model by collecting more data.

Summary & Recommendation

In conclusion, we have successfully developed a sentiment analysis model with an average weighted precision of 76%, prioritizing the accurate identification of positive tweets to minimize false positives. This enables my customer to act on positive tweets and leverage this information. However, it's worth noting that the model performs better in predicting positive tweets compared to negative ones. To further enhance its performance, we recommend collecting more data and addressing the class imbalance in the dataset. This model represents a critical step in harnessing the power of data science to gain actionable insights from social media data, enabling our client to refine their strategies and maintain a competitive edge in the dynamic digital landscape.

This repository

My technical code is stored in this jupyter notebook

My presentation can be found here

My github repository is here

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

● **Jupyter Notebook** 100.0%