akaigraham / module_4_project Public

0 stars 0 forks

Star Unwatch

Code Issues Pull requests Actions Projects Wiki Security

main

akaigraham updated README ... 3 minutes ago 46

View code

README.md

Phase 4 Final Project - Twitter Sentiment / Text Classification (Kai Graham)

README Outline

Within this `README.md` you will find:

1. Introduction
2. Overview of Repository Contents
3. Project Objectives
4. Overview of the Process
5. Findings & Recommendations
6. Conclusion / Summary

Introduction

Build a classifier to identify whether a tweet is of positive, negative, or neutral sentiment. Ultimate goal is to create a classifier that can be used by product managers / public relations stakeholders to track public sentiment in real-time, especially as new products and updates are released.

Repository Contents

1. `README.md`
2. `sentiment_nlp.ipynb` - clean jupyter notebook containing all code
3. `glove.6B.50d.txt` - Global Vectors for Word Representation file for leveraging word embeddings
4. `judge-1377884607_tweet_product_company.csv` - raw dataset
5. `best.pickle` - best identified model, fit to training set
6. `tweet_sentiment.pdf` - non-technical presentation

Project Objectives

Build a classifier to predict whether a tweet is of positive, negative, or neutral sentiment, in context of product managers / public relations personnel tracking public sentiment related to certain events, releases, updates, etc. Follow CRISP-DM machine learning process to explore dataset, prepare data for modeling, modeling, and post-model evaluation. Will be focused on accuracy as our main performance metric as context of false positives vs. false negatives is not as relevant with tweet data, especially given volume of it.

Overview of the Process:

Following CRISP-DM, the process outlined within `sentiment_nlp.ipynb` follows 5 key steps, including:

1. Business Understanding: Outlines facts and requirements of the project. Specifically a clssifier will be built and trained on twitter text data to predict whether that tweet is of positive, negative, or neutral sentiment. Understanding public sentiment surround product releases, updates, other pushes will be beneficial to product managers and public relations professionals to track the public's responses.
2. Data Understanding: focused on unpacking data available to us and leveraged throughout classification. Section will focus on the distribution of our data, any imbalances within our target predictor, etc.

3. Data Preparation: further preprocessing of our data to prepare for modeling. This includes splitting into training and testing sets, text processing, vectorization, and other techniques.
4. Modeling: this section iteratively trains a number of machine learning models, primarily focused on random forest classifiers, LinearSVC classifiers, and neural networks.
5. Evaluation: Final / optimal model is selected and final performance metrics of final model are discussed and evaluated. Focused on accuracy as our performance metric.

Findings & Recommendations

The best performing model we saw was a neural network trained on count vectorized data. With a testing accuracy score of ~67% our model is expected to generalize fairly well and produce proper class labeling 67% of the time. Given imbalanced dataset to begin with, and lack of negative class labels, it may be beneficial to find labels for more negative tweets / go back in time when it was known that public sentiment was not at it's highest. More training dat would be beneficial in this scenario. I recommend using this classifier in conjunction with other tools to monitor public relations surrounding product launches and other releases or events.

Conclusions & Summary

Through an iterative modeling and data preparation process, we were able to tune a model with ~67% accuracy. Through this process accuracy score was selected as the main performance metric. Training times were also considered as part of the evaluation phase.

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

Languages

- **Jupyter Notebook** 100.0%