Capstone Project Write Up

Domain Background

Sentiment analysis is an algorithmic approach used to understand the emotional context of a body of text and is used in a wide range of domains including but not limited to:

- Understanding polarity of financial texts and news articles
- Interpreting product reviews from customers
- Identifying racism / sexism in social media posts
- Understanding movie popularity based on reviews

In the context of sentiment analysis for domain specific texts, historical approaches [1] can broadly be categorised as follows:

- generic dictionary based methods
- domain specific dictionary based
- statistical or machine learning based methods

This project will attempt to classify sentiment of Financial based texts.

Problem Statement

Sentiment of financial news articles and headlines, as well as other financial texts, can be an indicator of company stock returns and volatility.

Other than reading articles and manually tracking sentiment, it is not possible for retail investors to get an overview of sentiment from the financial news towards a company. Such a method would also be susceptible to bias as categorisation of article sentiment maybe altered based on underlying opinion towards the company in question.

Solution Statement

The aim of this project is to address the problem statement by giving retail investors access to statistics describing the sentiment of headlines for companies of interest, over a given historical time period. As such, the news article will need be classified as positive, negative or neutral, from the point of view of a retail investor.

To achieve this functionality a Sentiment Classifier will need to be trained, tested and deployed and a web application will need to be built to enable the user to retrieve predictions.

The solution is described in detail under the heading Project Design.

Dataset and Inputs

The dataset that will be used for both training and testing the Sentiment Analysis algorithm is known as the "Financial Phrase Bank" dataset. It contains around 5000 sentences described by three features "Sentiment", "News Headline" and "Confidence." **Sentiment** is the target variable containing our positive, negative and neutral classes. **Confidence** is a percentage which defines the level of agreement on sentiment between the annotators. **News Headline** is body of text to be analysed.

- https://www.researchgate.net/publication/251231364_FinancialPhraseBank-v10

This core dataset will be supplemented with third party dictionaries to help determine sentiment of technical financial terminology.

- https://github.com/jperla/sentiment-data/tree/master/finance

The trained model will be applied to articles headlines from a financial news API.

- https://stocknewsapi.com/

Evaluation Metrics

The problem statement requires classification so accuracy will be used to evaluate the benchmark and solution model. All scores will be cross-validated using the K-Fold cross validation technique.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

F-Score will be used alongside accuracy to help understand the performance of the model during development.

$$FScore = \frac{2 * Precision * Recall}{Precision + Recall}$$

$$Precision = \frac{TP}{TP + FN}$$

$$Recall = \frac{TP}{TP + FP}$$

Where,

$$TP: True\ Positives,\ TN: True\ Negatives,$$
$$FN: False\ Negatives,\ FP: FalsePositives$$

Since the problem is a multi-class classification problem, the definitions above are somewhat simplified and will be extended to the three dimensional case as part of the project.

Benchmark Model

Since no official benchmark exists, a simple benchmark is proposed. The dataset is imbalanced so an approximate 60% accuracy can be obtained by assuming all headlines are neutral due to the distribution in the dataset.

State of the art approaches have obtained over 90% accuracy [1] on the "Financial Phrase Bank" dataset. This level of accuracy may be difficult to replicate given my previous experience working with sentiment analysis.

Project Design

The project can be broken down into two sub components namely, a **Sentiment Classifier** and a **Web Application**, both of which are described in detail below.

Sentiment Classification:

Get Data:

Exploratory Analysis:

1. Distribution of classes
2. Distribution of confidence
3. Further analysis after feature engineering steps.

Potential PreProcessing / Feature Engineering Approaches:

1. Stemming
2. Bag-of-words
3. N-grams
4. Sentiment words
5. TF-IDF
6. Parts of Speech [4]
7. Negation [4]
8. Domain specific dictionary [3]
9. Domain specific opinion words [2]
10. Text tagging [1]

Potential Classification Approaches:

1. Benchmark model
2. SVM

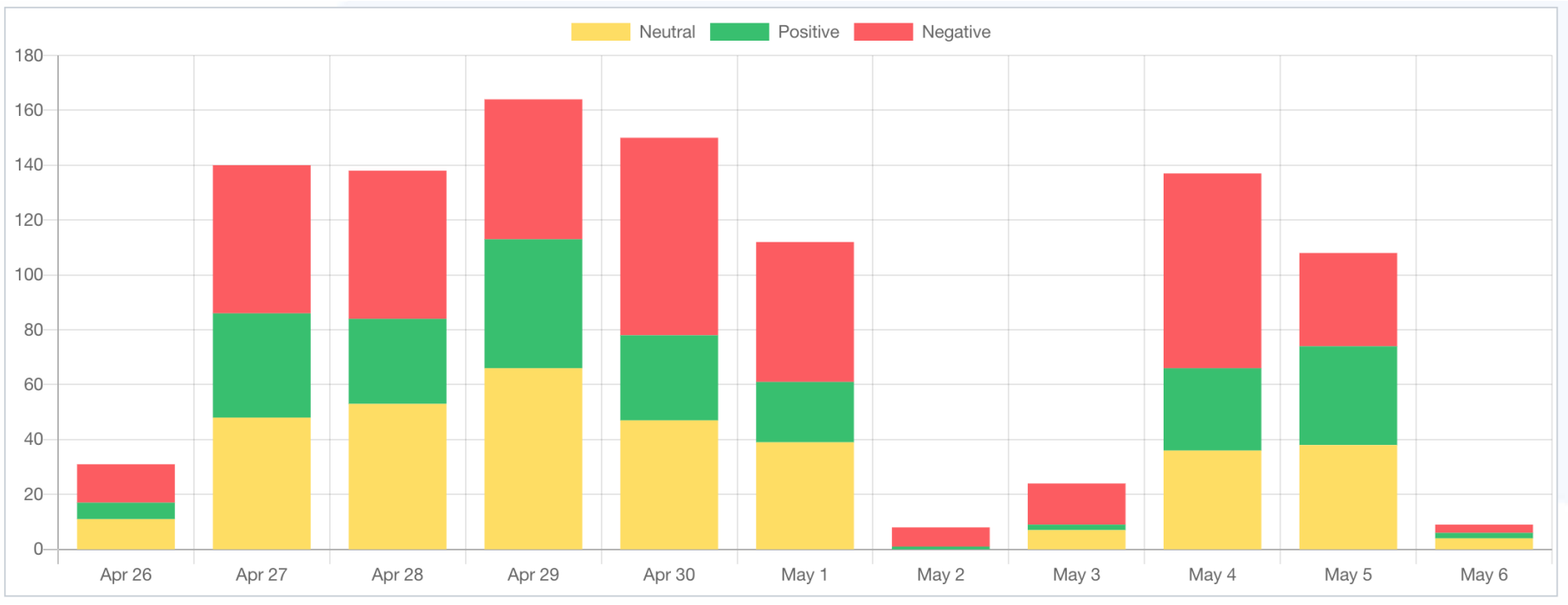
3. Naive Bayes
4. Recurrent Neural Networks [5]
5. LSTM [5]
6. Doc2Vec [5]
7. Hierarchical classifier / association rule mining [1]

Classification Workflow:

1. Select final model
2. Train model using AWS instance
3. Deploy endpoint to AWS instance
4. Test model by calling endpoint

Web Application:

1. User enters stock symbol for company of interest.
2. Web app calls financial news API and retrieves articles for given stock symbol over a given time period.
3. Headlines passed to sentiment classifier endpoint.
4. Positive, negative or neutral predictions returned and aggregated.
5. Descriptive statistics returned to user.



Resources

[1] Srikumar Krishnamoorthy, Sentiment Analysis of Financial News Articles using Performance Indicators, 2017

[2] Ilia Chetviorkin, Extraction of Domain-specific Opinion Words for Similar Domains

[3] Tim Loughrabn and Bill McDonald, When Is a Liability Not a Liability?

[4] Zhichao Han, Data and text mining of financial markets using news and social media

[5] Big Data: Deep Learning for financial sentiment analysis