Machine Learning Engineer Nanodegree

Capstone Proposal

@KT12
11/19/2016

Proposal for Study

Domain Background

The Dow Jones Industrial Average ("DJIA") tracks the movement of the 30 largest publicly traded companies by market capitalization, and has been a barometer for the US equity market since the late 19th century. Reddit is a large internet community which aggregates content and information, including news. Using data provided by Jiahao Sun at Kaggle, one can attempt to predict the movement of the DJIA using natural language programming techniques on the titles of prominent Reddit news articles from that day.

Since the 2008 Great Financial Crisis, market participants and regulators have been sensitive to 'headline risk.' Analyzing the relationship between news and market moves can inform the investing public of what types of headlines and words have an impact on the market. As a portfolio manager, I am interested to see if certain keywords in headlines can indicate whether a set of headlines is likely to result in a higher or lower market.

While this study will not serve as a direct test of the efficient market hypothesis (EMH), it will be interesting to document the nature of pertinent headlines, whether unforseeable tail risk (natural disaster, terror, political or regulatory event), a change in expectations (Fed meeting, earnings surprise, new economic datapoints), or something else.

Prior studies have investigated the relationship between news and the market using sentiment of news to predict the S&P 500, natural language processing of earnings calls/reports to infer volatility, and unigrams and shallow semantic relations to infer single stock movement.

Problem Statement

The machine learning problem that will be investigated is the binary classification of the daily movement of the DJIA based on features extracted from Reddit world news headlines using NLP techniques. A classifier (SVM or Neural Net in `sklearn` ok `keras`) will fit the extracted textual features to the binary DJIA response variable. With an expanded corpora of headlines, the same analysis could be extended to different markets or securities.

Datasets and Inputs

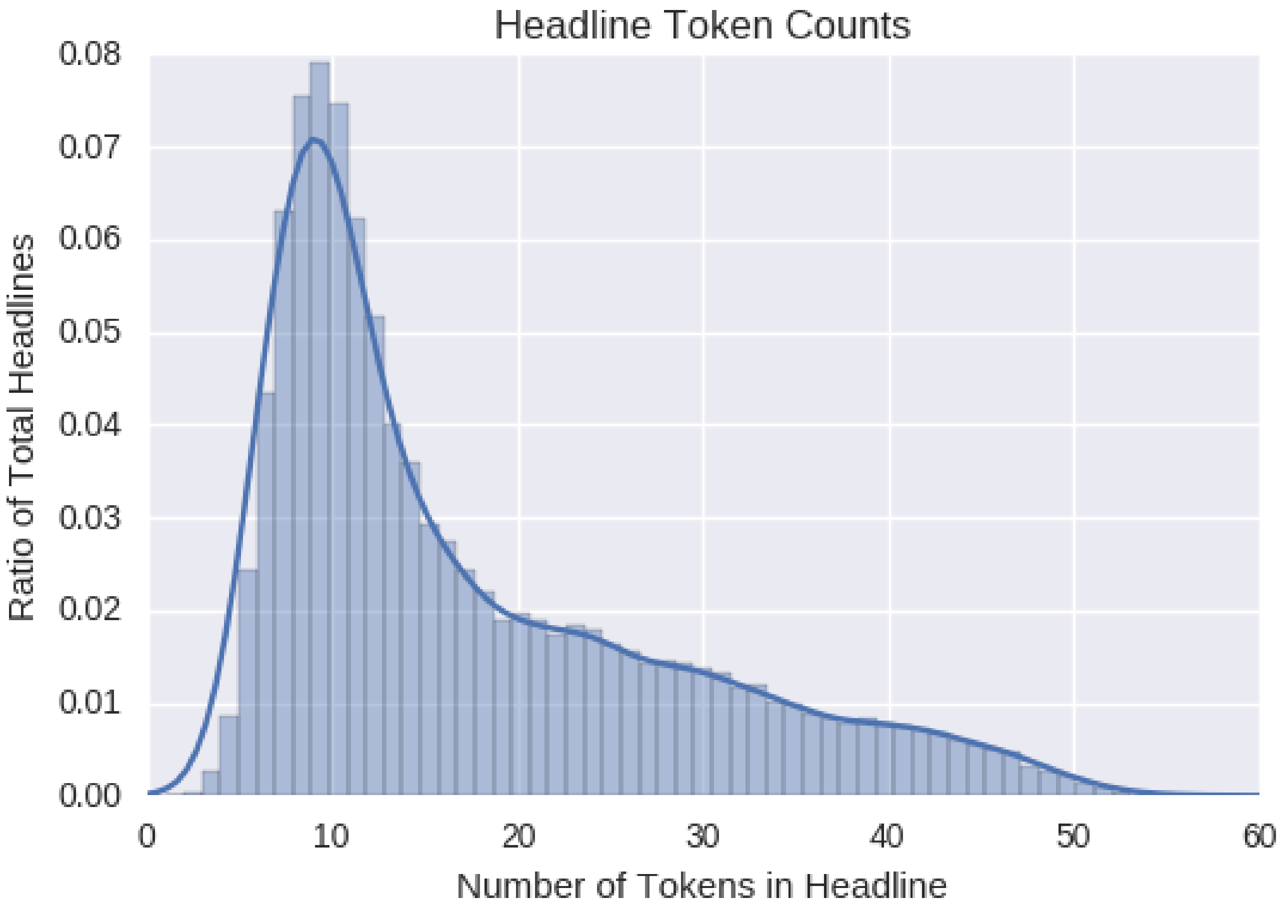
Datasets for both index prices and headlines are provided from Kaggle in CSV form. The relevant CSV includes the trading date, the label indicating DJIA movement, and headlines of the top 25 world news articles from Reddit.

The data includes nearly eight years of both DJIA and headline data from August 8th, 2008 until July 1, 2016. There are 1,989 examples with 25 headline features.

The DJIA classification label has been extracted from a CSV log of DJIA data which includes trading date, open, close, high, low, and trading volume. Over the sample of 1,989 trading days, 1,065 resulted in the index finishing flat or higher. These will be labeled `1`. Days in which the index finsihed lower than the previous day's close will be labeled `0`. The ratio of 'up' days is 0.5354. The classes are well balanced.

The headlines from Reddit are assigned to 25 different cells, in decreasing order of importance (as decided by the community). NLP techniques will be used on these headlines to extract features.

Over the entire data set of headlines, the median token count is 13 and the mean is 17.3 tokens. The maximum is 59 tokens, the minimum is 1 token, and the standard deviation is 11 tokens.



Both feature engineering and selection will be crucial steps to ensure the classifier is successful. The `nltk` and `sklearn` packages will be utilized to evaluate and extract features from the headlines.

Solution Statement

The solution to the problem would be to fit an support vector machine classifier or neural network classifier with features extracted from the various headlines. The feature engineering aspect of the project will be crucial in order to fit a classifier which is accurate.

Benchmark Model

A benchmark to compare the classifier against would be a classifier which guessed randomly. This is similar to the oft-cited [monkey throwing darts at a paper]. The random classifier will be compared to the headline-informed classifier using the metrics below.

Evaluation Metrics

The model will be evaluated using both F1 score and area under the receiver operating characteristic curve (AUROC). Both metrics are very useful in evaluating binary classification models.

F1 score is the harmonic mean of precision and recall. Precision is the ratio of true positives to all positive predictions. Recall is the ratio of true positives to all positive examples.

$$F_1 Score = \frac{(2 \cdot Precision \cdot Recall)}{(Precision + Recall)}$$

Where:

$$Precision = \frac{true\ positives}{true\ positives\ +\ false\ positives}$$

and

$$Recall = \frac{true\ positives}{true\ positives\ +\ false\ negatives}$$

AUROC measures the trade-off between the true positive rate and the false positive rate as the classifier's threshold is varied. The metric is only sensitive to rank ordering of the two classes, and essentially measures the probability that a randomly chosen positive example and randomly chosen negative example will be classified correctly. In contrast to the F1 score, the AUROC metric is not sensitive to inbalance classes.

While easier classification exercises with more data to test and train can attain a high level of accuracy, the stock market has been analyzed very carefully by many actors with an assortment of techniques. A randomly guessing algorithm would be expected to get about half of its guesses correct. While the situation that is under investigation is slightly contrived (the news headlines are not timestamped so one cannot ascertain if the market closed before or after their release), being able to 'predict' market moves with even 2-3% greater accuracy than random has value.

Project Design

A theoretical workflow for this project would be:

Setup and Data Preprocessing

- Import all necessary packages and modules into Jupyter Notebook
- Load the CSV into Python into `NumPy` arrays.
- Split the data into fifths. The chronologically earlier data will serve as a training set to cross validate the 'unseen' test set on a rolling basis. To avoid data leakage in cross validation, each training set will undergo feature engineering separately.

Data Exploration and Feature Engineering

- Utilize the `nltk` package to:
  - remove stop words with `stopwords`
  - inspect collocations with `collocations`, n-grams with `ngrams`
  - create positive/negative sentiment scores with `vader`

- Utilize `sklearn.feature_extraction.text` to analyze term frequency-inverse document frequency (TF-IDF) of words
- Potentially use the `word2vec` module from the `gensim` package to preprocess and create 'neural word embeddings.'
- Explore the data of the target variable and headlines
    - Create histogram plots of headline length and other summary statistics.
    - Check for unbalanced class distribution in the response variable
- Create new features by combining text headlines for each date or only using the n^th^ most important headline of the day

Training Classifier

- Use `PassiveAggressiveClassifier`from `sklearn` to take a first pass over the data. The algorithm is very fast compared to a neural net and should provide a good idea of which features are most informative for classification.
    - Observe the useful features and possible engineer new features based on similar attributes.
    - Review the weights of the classifier. Check that they make intuitive sense and pass the 'smell test.'
- Use `keras` with a `tensorflow` backend to construct a neural network classifier which will train on a subset of the engineered features. Conduct a gridsearch with cross validation to both optimize the parameters and reduce overfitting.
- Use the `DummyClassifier` from `sklearn` to create a baseline classifier which guesses randomly.

Classification Prediction and Evaluation

- Predict the classes of the test set using the neural net and the random classifier on each training set using each training set-derived classifier.
- Using the cross validation set on a rolling basis:
    - Report accuracy of the classifier
    - Display the confusion matrix
    - Evaluate the neural network using both F1 Score and AUROC analyses. Compare the results to the baseline.