E6893 Big Data Analytics (Project ID: 201812-23)

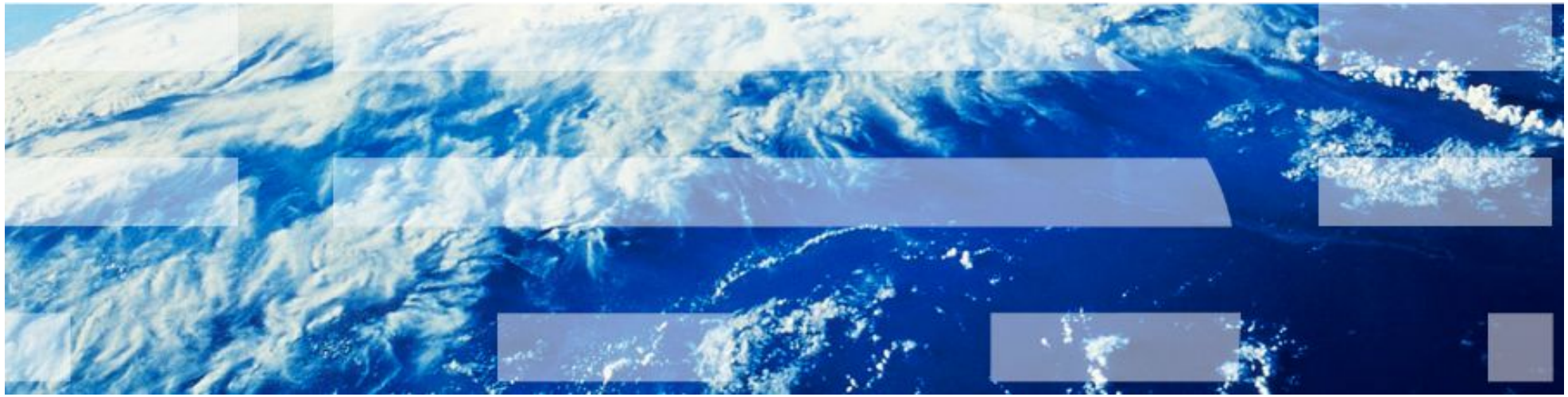
Stock Performance Predictions based on News Analytics

Team Members:

Fan Gao (fg2432)

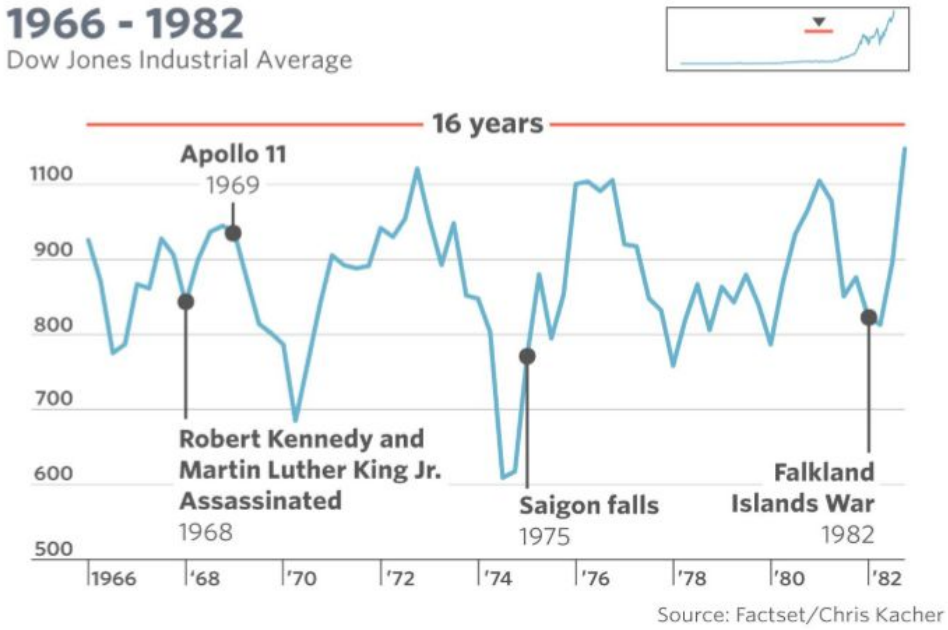
Connor Gatlin (ceg2195)

Ruisi Wang (rw2720)



Motivation

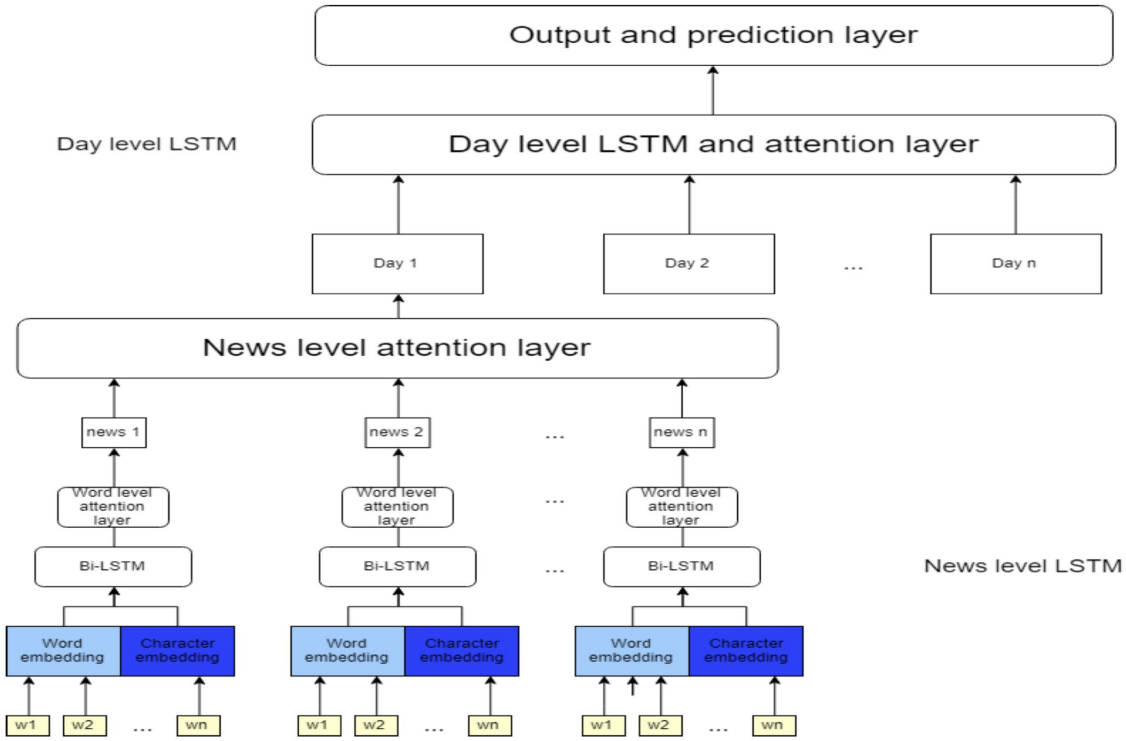
- Historically, news and stock performance are highly correlated
- Understand the predictive power of news
- Compare stock performance predictive models



Related Work

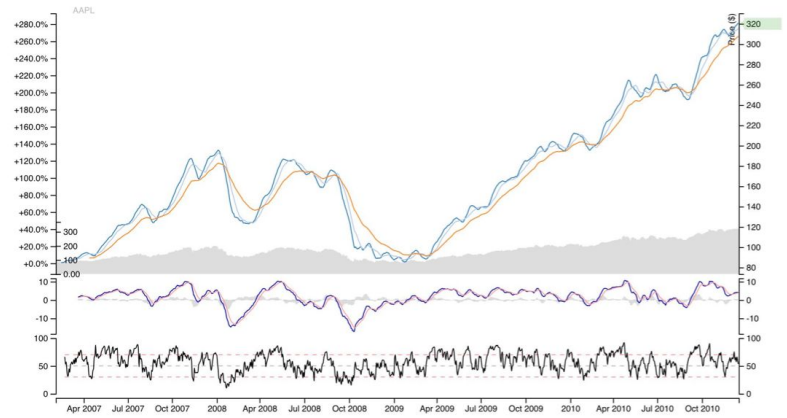
1. Liu (2018) proposed an attention-based LSTM model (At-LSTM).
a. It predicts the directional movements of Standard & Poor's 500 index and stock price of individual companies using financial news titles.
b. Results are competitive with the state-of-the-art model which incorporate knowledge graph into the learning process of event embeddings.

2. Joshi, Bharathi, Rao (2016) used polarity detection algorithm for initially labelling news and making the training set.
a. Based on this data, they showed that a tree-based model works very well for all test cases with accuracy ranging from 88% to 92%.



Datasets

- Market data
  - 2007 to present (4,072,956 samples - 1.2GB)
  - Provided by Intrinio
  - Stock Market data
    - Asset Code (Stock Ticker)
    - Open/close stock price
    - Trading Volume
    - Returns (1D, 10D)
    - Label - Future Return (10D)



- News data
  - 2007 to present (9,328,750 samples - 4.7GB)
  - Provided by Thomson Reuters
  - News articles and alerts published about assets
    - Headline
    - Asset Code mentions
    - Sentiment
    - Market Commentary

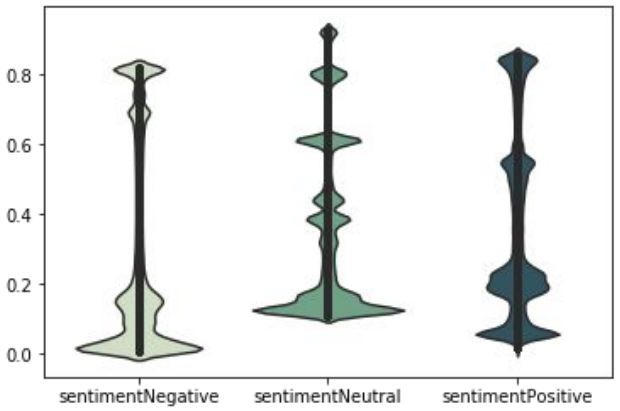
Tools

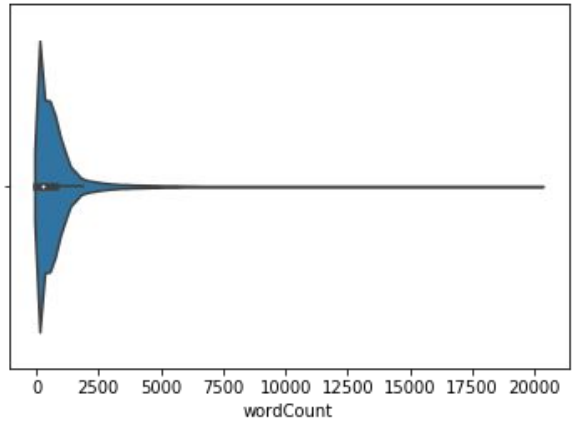
- Languages/Packages:
  - Python, scikit-learn, pandas, nltk, DMTK LightGBM, Catboost, Keras
- Visualization:
  - Node.js, d3.js, pyplot, matplotlib, seaborn
- Platforms:
  - Jupyter Notebook, Google Cloud

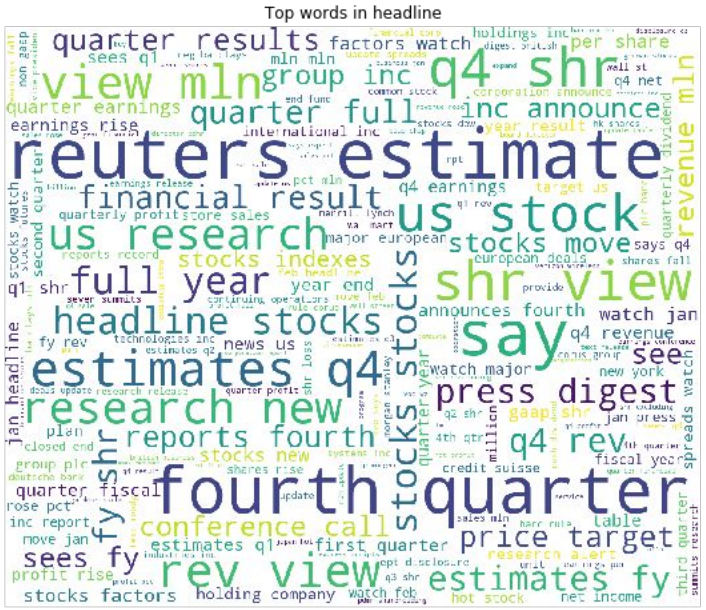
Data Visualization

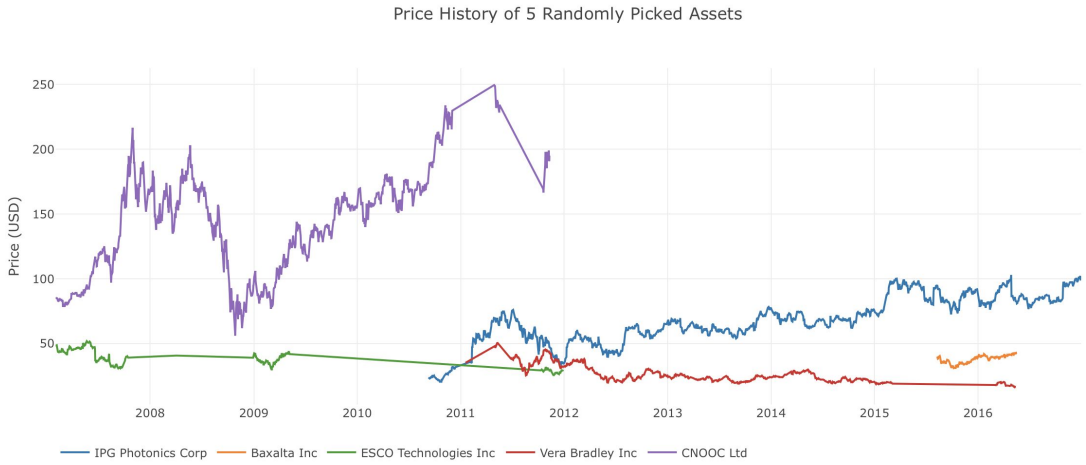
- Website

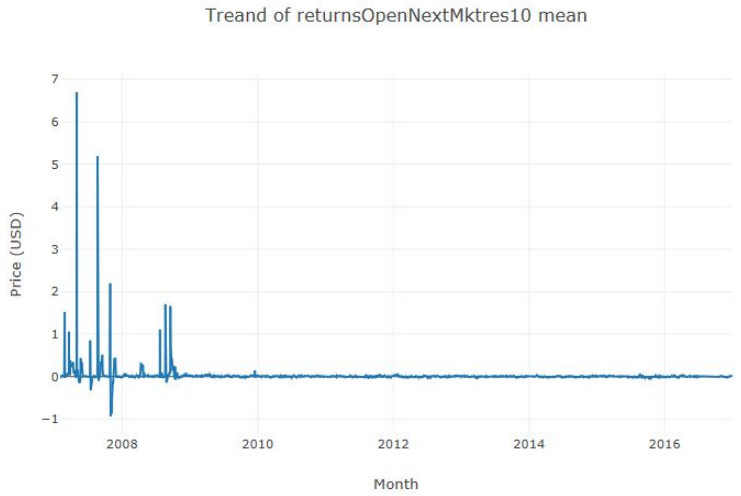
Data Exploration





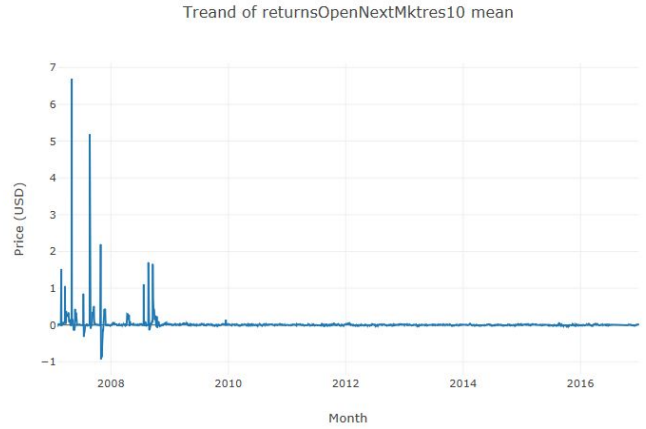


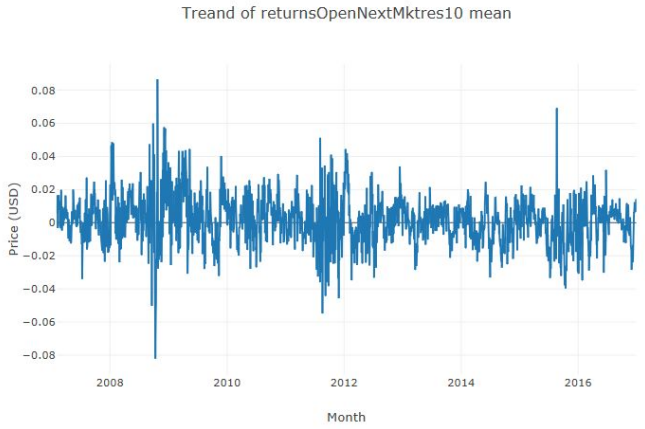


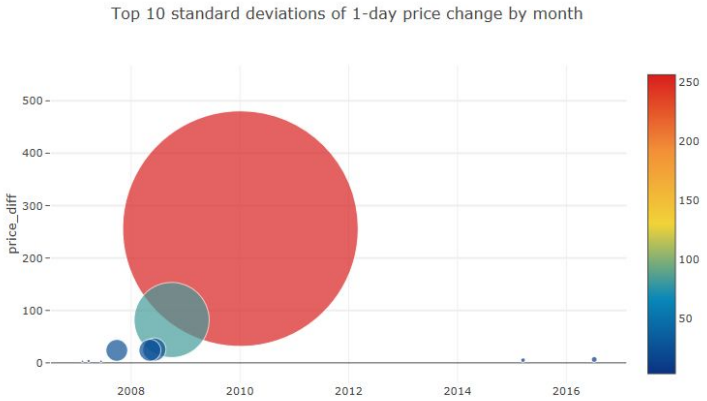


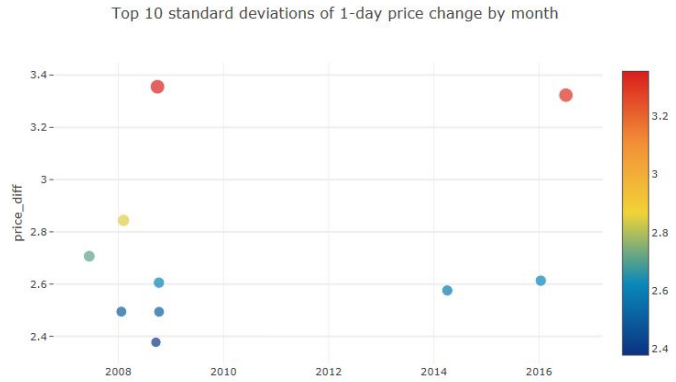
Data Preparation

- Data Cleaning and Outlier Removal









Feature Extraction

- Market Feature Extraction
  - Extract derived numerical features
    - ex: close-to-open ratio, volume-to-average-volume ratio, etc.
  - Categorical feature encoding (assetCode, assetName)

- News Feature Extraction
  - Convert headlines/subjects to numerical word vectors
  - Convert other categorical features to numerical
  - Flatten news articles to one row per day/assetCode

```
'sentimentNegative': ['min', 'max', 'mean', 'std'],
'sentimentNeutral': ['min', 'max', 'mean', 'std'],
'sentimentPositive': ['min', 'max', 'mean', 'std'],
```

- Data Merge
  - Flatten News Data and Merge with Market Data on “Asset Code” and “Time”
    - 114 Features
    - 4,072,956 Samples

	assetCode	assetName	volume	close	open	returnsClosePrevRaw1	returnsOpenPrevRa
count	3.000000e+06	3.000000e+06	3.000000e+06	3.000000e+06	3.000000e+06	3.000000e+06	3.000000e+06
mean	9.592328e+02	9.242142e+02	2.551221e+06	4.201477e+01	4.200672e+01	5.804256e-04	5.839471e-04
std	6.729093e+02	6.455006e+02	6.932746e+06	4.611590e+01	4.611178e+01	2.148449e-02	2.164497e-02
min	-1.000000e+00	-1.000000e+00	0.000000e+00	7.000000e-02	8.000000e-02	-1.000000e-01	-1.000000e-01
25%	4.170000e+02	4.010000e+02	4.552430e+05	1.778000e+01	1.778000e+01	-9.587897e-03	-9.740259e-03
50%	8.400000e+02	8.210000e+02	9.635360e+05	3.181000e+01	3.180000e+01	4.649000e-04	5.576545e-04
75%	1.385000e+03	1.339000e+03	2.338324e+06	5.270000e+01	5.268000e+01	1.066351e-02	1.084645e-02
max	3.250000e+03	3.124000e+03	1.226791e+09	1.578130e+03	1.584440e+03	1.000000e-01	1.000000e-01

Modeling - Evaluation

- Models predict signed confidence value:

$$y_{ti} \in [-1,1]$$

- For each day, we calculate:

$$x_t = \sum_i y_{ti} r_{ti} u_{ti}$$

r_{ti} is the 10-day market-adjusted leading return for day t for instrument i
u_{ti} is a 0/1 universe variable that controls whether a particular asset is included in scoring on a particular day.

- Final score for model is calculated with:

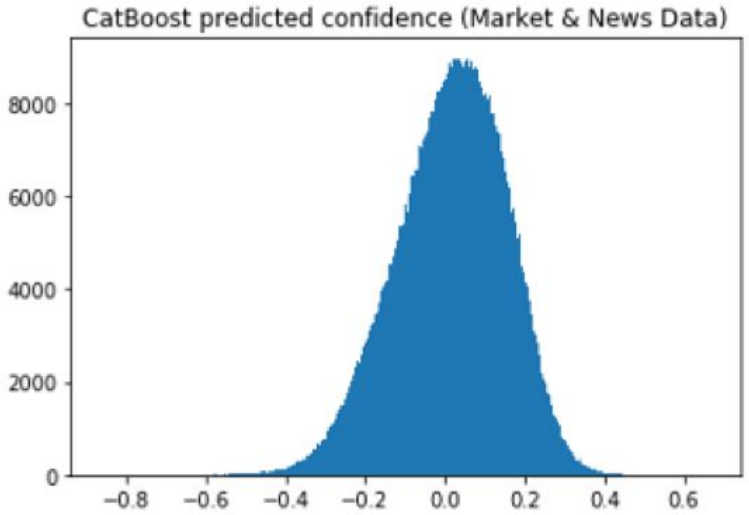
$$score = \frac{\bar{x}_t}{\sigma(x_t)}$$

Modeling - Algorithms

- Benchmark
  - Confidence values all equal to 1
  - Equivalent to buying all assets in the market on each day

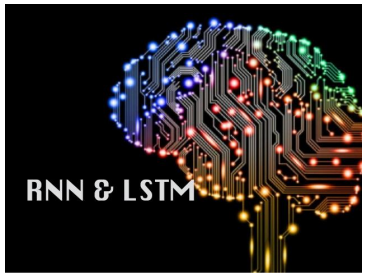
- CatBoost Classifier
  - Gradient boosted decision trees
  - Returns a confidence value

Yandex CatBoost



- LightGBM Regressor
  - Gradient boosted decision trees
  - Returns an estimated 10 day leading return

- LSTM Neural Network
  - Recurrent neural network
  - Beneficial for time-based series data

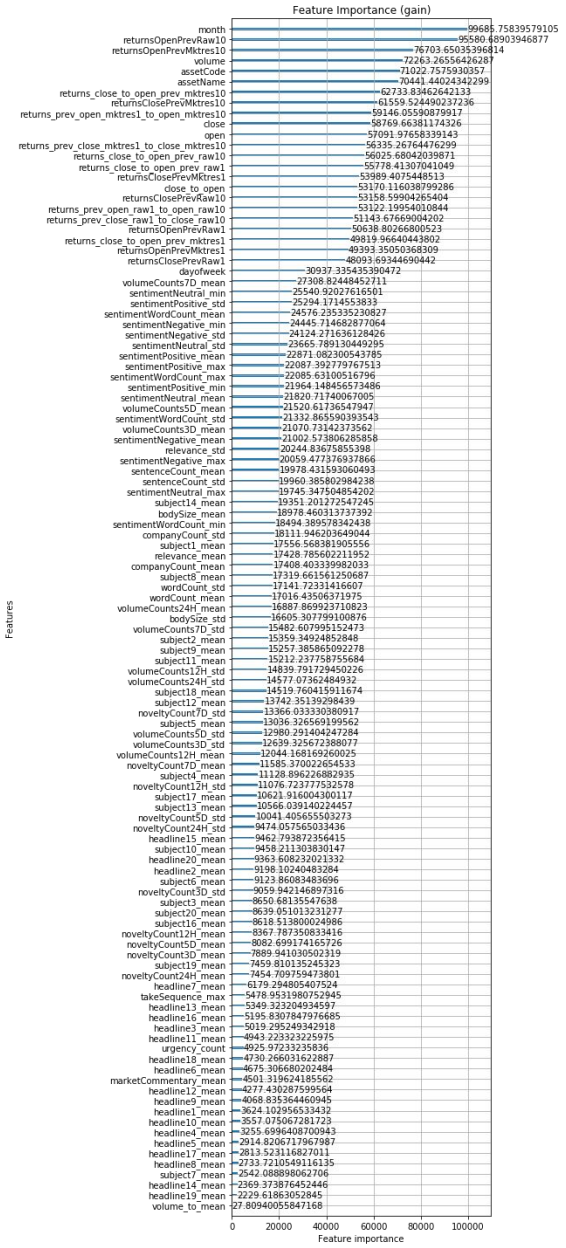


Results

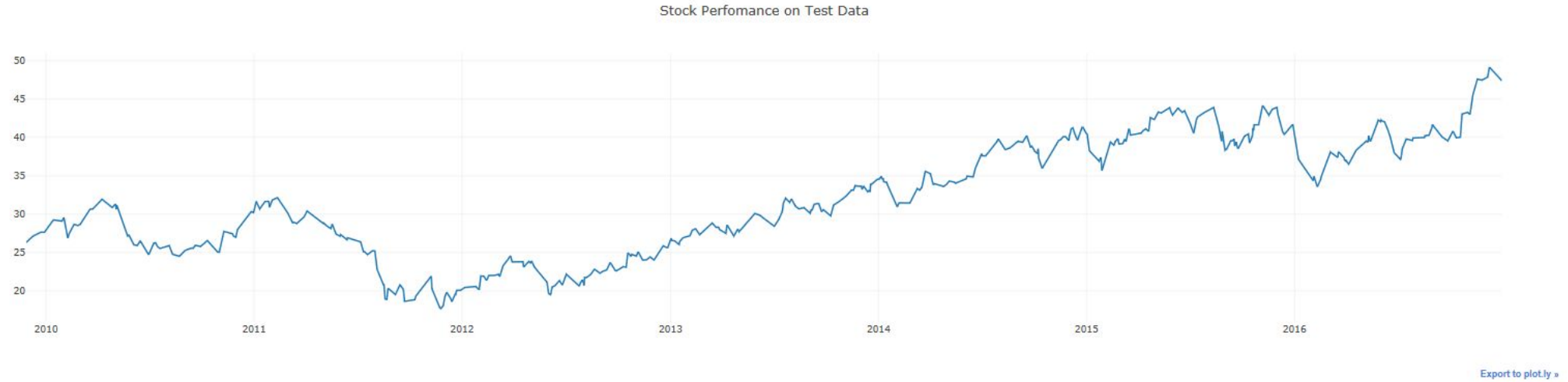
- LightGBM Regressor provided the best score on the test set
- All models provided significant improvement over benchmark
- Classification Model was improved by 34.9% with added news data

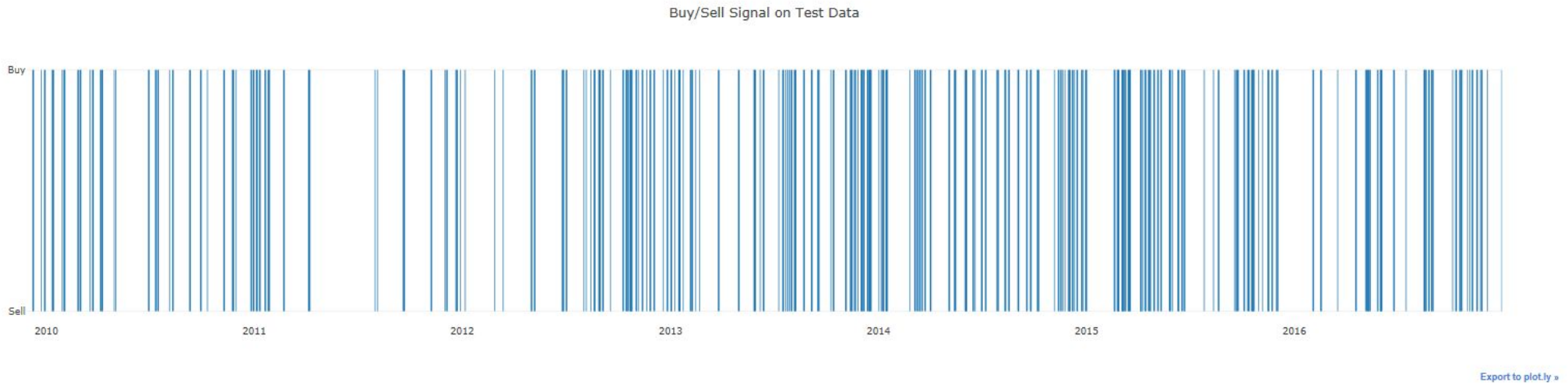
Training (70%)	Validation (20%)	Test (10%)

Model	**Score on Test Data**
Benchmark	0.016
CatBoost Classifier w/out News Data	0.630
CatBoost Classifier	0.850
LightGBM Regressor	0.916
LSTM Neural Network	TBD



Results





Reference

- Huicheng Liu. “Leveraging Financial News for Stock Trend Prediction with Attention-Based Recurrent Neural Network”. In: arXiv:1811.06173 (2018).
- Kalyani Joshi, Bharathi H. N, and Jyothi Rao. “Stock Trend Prediction Using News Sentiment Analysis”. In: arXiv:1607.01958 (2016).
- Liudmila Prokhorenkova, et al. “CatBoost: unbiased boosting with categorical features”. In: arXiv: 1706.09516v4 (2018).
- Guolin Ke, et al. “LightGBM: A Highly Efficient Gradient Boosting Decision Tree”. In: Advances in Neural Information Processing Systems 30 (NIPS 2017) (2017).
- Kaggle. “Two Sigma: Using News to Predict Stock Movements”. In: Featured Code Competition (2018).

YouTube Link

- https://youtu.be/8oagRyBbris