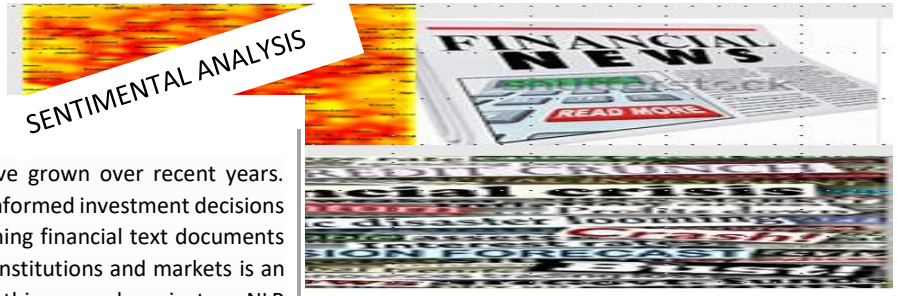
FINANCIAL TEXT MINING (NLP PROJECT IN PYTHON)



STUDENT NAME: *MUWANI ROBSON*

SUPERVISROR NAME: *VIJAY KUMAR*

Objectives

The aim is to build a predictive model and find whether the news may have positive, negative, or neutral influence on the stock price. As a result, sentences which have a sentiment that is not relevant from an economic or financial perspective are considered neutral. Using this model, the financial phrase bank will help prospective investor to know which news headlines will have an impact on the stock market prices.

Abstract

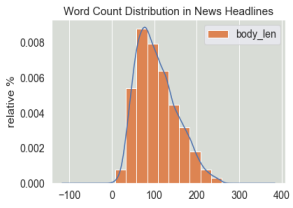
The trend of financial reports, news and publications have grown over recent years. Assessing the relevance of financial information in making informed investment decisions has become an important aspect to modern investors. Mining financial text documents and understanding the sentiments of individual investors, institutions and markets is an important and challenging problem in financial markets. In this research project, an NLP model has been proposed for opinion mining in financial text documents or new headlines based on multiple features. The proposed model(s) has been evaluated based on the performance parameters of the precision, recall and polarity-based accuracy assessment, which gives the overall perspective of the overall accuracy. The proposed model has been clearly defined as being better than other models, when assessed on the given parameters.

Hypothesis

Positive sentiments: news headlines containing positive sentiments are on average less verbose (contain words < 50 on average) **Negative sentiments:** Negative reports, negative profits, decline in earnings per share and negative news reports results in negative sentiments hence decline in stock market prices. Investors will recommend to sell or to liquidate the asset. **Neutral sentiments:** generally financial news with no impact on stock prices are more verbose and contain more words (greater than 100), Investors tend to take wait and see scenario or just a hold on position.

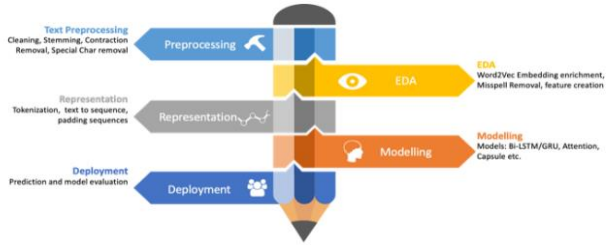
Text Preprocessing and Data Exploration.

The data preprocessing entailed cleaning the financial text using regular expression programming tool. Some of the cleaning tasks was to change numeric values to read as money, year of figures, changing phone numbers to be read as just phone numbers and replacing a variety of money symbols. After this, special characters and white space in the text was removed. Stemming and lemmatization was applied.



Neutral sentiments constitute 59% of the total dataset while negative sentiments take the baseline record of 12% from the data set. Negative news headlines contained more words than the rest of the class. Negative class top for most news headlines with words exceeding 50. Neutral statements top for news headlines less than 50. Negative news headlines contained more words than the rest of the class.

NLP TEXT CLEANING AND MODEL PROCESS



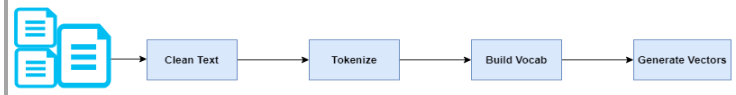
- Logistic regression performed better than all the models, recording an accuracy of ~72%.
- SVM, SGD, Random Forests and KNN all recorded an accuracy of ~70%~70%,~69% and ~65% respectively
- Worst performing models were LDA and NB, all below
- There was greater variability in model performance as read from k fold model stdv, the validation was set at cv=10 for all the models that performed better.

FINDINGS

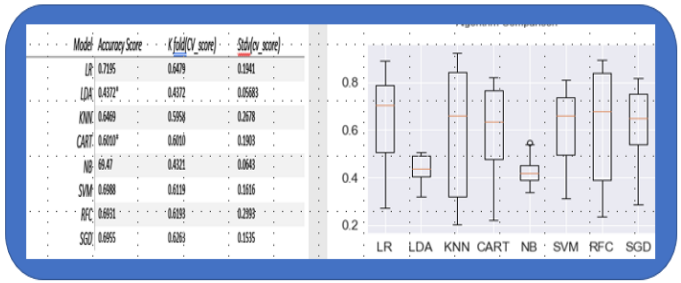
- Irrelevant financial text /news or document are more verbose, contain more words on average exceeding 100 words compared to financial text with positive sentiments.
- Logistic regression presented as a better model in predicting the polarity of financial news headlines. However, the score signifies need for improvement through use of best hyper parameters (Gridsearch).
- Model improvement was applied via ensemble method, score went up to 73%.
- NB, SGD, RFC and SVC presented alternative accuracy scores but needs great improvement.
- Great variability was observed in K-fold accuracy scores in all models that performed better, test scores and accuracy score almost the same for all the models.
- K fold score stdv was too high for all models that performed better (variance between the k-fold scores, for cv=10).
- LDA, KNN and CART models proved to be the worst models in

Feature Engineering: creating a bag of words

After cleaning the data, we tokenized and encoded the text, converted the words into a sequence or padding sequence. This was basically encoding the given texts using internal vocabulary with optionally applied encoding options. Now that we have tokenized our data and have a word to numeric representation mapping of our vocabulary. Since neural networks work by performing computation on numbers, passing in a bunch of words, the algorithm will not understand it. Training data set was created and separated from the test data set, test sample size was set 25%.



Model Results:



Model	Accuracy Score	K fold(CV score)	Stdv(cv score)
LR	0.7195	0.6479	0.1941
LDA	0.4372*	0.4372	0.05683
KNN	0.6469	0.5956	0.2678
CART	0.6010*	0.6010	0.1303
NB	69.47	0.4321	0.0643
SVM	0.6988	0.6119	0.1616
RFC	0.6931	0.6193	0.2393
SGD	0.6955	0.6263	0.1535

Model improvement:

To improve the model , an ensemble method was applied. Ensemble methods are **meta-algorithms that combine several machine learning techniques into one predictive model** in order to decrease variance (bagging), bias (boosting), or improve predictions (stacking).

Ensemble Results:

Model	Accuracy Score		Precision	Recall
Ensemble	0.73	Positive(2)	0.72	0.50
		Neutral (1)	0.74	0.89
		Negative (0)	0.62	0.47

Model results improved to 73%, precision as a measure of relevant and correct classifications by the model was 62% and recall i.e. a measure all relevant/ correct classifications for Negative sentiments was very low at 47%.

Recommendations

- The obtaining results show suitability of Logistic regression as a better model in predicting the polarity of financial text for this data set. Therefore, I recommend using this model with great caution to avoid overfitting. The model accuracy can further be improved by selecting best parameters and hyper parameters.
- **Use of ensemble:** thus, select all models with average score of 70% and run an ensemble method with only better models (L R,NB,SGD ,RFC and SVC).
- **Add more data**-for models with high variance high variance of predictions it is ideal to increase the training set size. Try increasing