Predicting the Effects of News Sentiments on the Stock Market

M Dhathri Praveenya
AP19110010217
SRM AP

Neeraja Devireddy
AP19110010193
SRM AP

Naveena Bonu
AP19110010146
SRM AP

Ravitarun Dasari
AP19110010201
SRM AP

Supraja
AP19110010300
SRM AP

Abstract

Stock market forecasting is very important in the planning of business activities. Stock market variations depend on various factors/situations that happen around the world. Language Processing algorithms like Bag Of Words, and TF-IDF construct vectors of each and every document present in the Dataset. Our model learns vector representations of sentiments in Headlines of newspapers. After the construction of Vectors for each and every document, they are preprocessed and given to machine Learning Classification Models like Random forest, and naive Bayes to learn the patterns hidden behind the stock prices based on Headlines. So that our model can predict the sentiments of the stock price by previously learned patterns.

Introduction

These days predictive analytics has become the key subject of study and research as Predictive analytics uses many techniques from data mining, statistics, modeling, machine learning, and artificial intelligence to analyze current data to make predictions about the future. Many researchers across the globe are experimenting with a variety of techniques for increasing the speed of the process of processing online as well as offline documents with the aim of producing valuable insights. There are plenty of algorithms for doing various analytical activities in order to extract actionable insights in time out of huge datasets. Nowadays we know that Machine learning is one of the hot topics in education as well as in industries to make analytical tasks smoother. Sentiment analysis systems are being applied in almost every business and social domain because opinions are central to almost all human activities and are key influencers of our behaviors. Our beliefs and perceptions of reality, and the choices we make, are largely conditioned on how others see and evaluate the world. For this reason, when we need to make a decision we often seek out the opinions of others. This is true not only for individuals but also for organizations.

Now that we have all the data in the form of vectors, we are training our machine learning model.

Problem Survey

As newspapers make a lot of impact on the stock market, this prototype is based on given headlines in the respective newspaper on a particular date. We predict whether your stock prices will increase or decrease. Positive news will normally cause individuals to buy stocks. Good earnings reports, an announcement of a new product, a corporate acquisition, and positive economic indicators all translate into buying pressure and an increase in stock prices. This is to extract sentiment from an object web-wide and need to automate opinion-mining systems to do it. The existing techniques for sentiment analysis include machine learning (supervised and unsupervised), and lexical-based approaches. Hence, the main aim of this paper presents a survey of sentiment analysis (SA) and opinions written by newspapers.

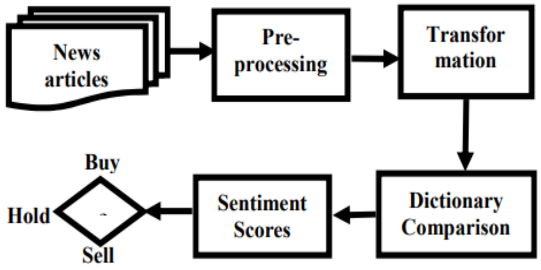


Fig 1. Sentimental Model

Dataset Description

We use the data set named “df”, which consists of records like date, label, and 25 headlines. There are 4102 data points and 27 features in the dataset. The text consists of predictions of the stock market. We can predict the changes in stock prices by the sentiments in the headlines. In case there is an increase in prices, the label is written as 1 else 0 if there is either no change or decrease in stocks.

The training dataset will be 3975.

The testing datasets will be 126

Proposed model

We took a dataset that contains Records of the top25 headlines of the day from 2000-to 2015. We considered all the records before 31122014 as train data but then we extracted records after 01012015 to perform predictions and named it a test (we call it test data). Train data has 3975 records.

Now we preprocess our data by changing the indexes for our convenience and converting everything to small case letters and all the special characters are removed. Now we merge all the headlines irrespective of the year and ranking into a list of strings. Now that we pre-processed, cleaned, and merged our data, we have to model our data from text to a set of vectors using BOW (Bag of Words). Bag of Words is an NLP technique that is used to create vector representation for the given textual data and those vectors were given to the ML algorithm to do predictions depending on what the use case(i.e Logistic Regression) is. This model can understand data in the format of integer/float values. But our dataset contains the headlines on various days which are in the form of text. So we convert these text data in the form of vectors which consists of the frequency of the words of that particular review.

We perform BOW using a countVectorizer and random forest classifier and Multinomial naive Bayes. The random forest classifier is a classification method used in the process of training a machine learning model consisting of many decision trees. It uses bagging when building individual trees to create correlated forests as it is more accurate than having individual trees when making decisions. Count vectorizer is a library in python used to convert text to vectors on the basis of the frequency of appearance of each word in a given text. A bag of words gives us the vector of frequency of words in the text.

Result

After training the model, the next step is to test the accuracy of the model using the test data. After testing, we construct a confusion matrix to check the accuracy and performance of our model. A confusion matrix is a table that is constructed on the basis of the results of data of which true values are known. The confusion matrix has four sections(true positive, true negative, false-positive, and false-negative). *Our model has given accuracy of 84% using Naive Bayes and Random Forest Classification.*

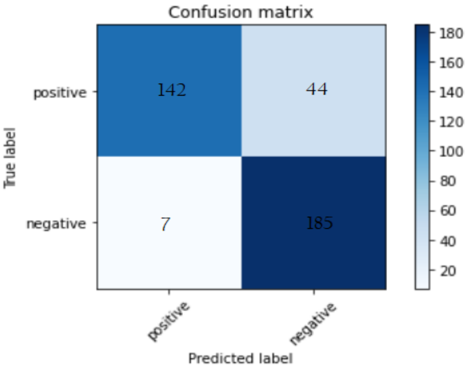


Fig 2. Confusion matrix- Random Forest

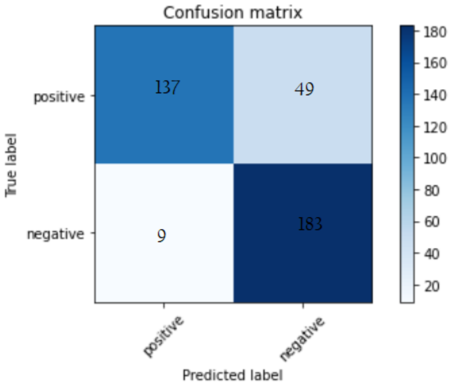
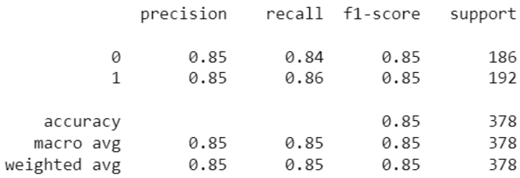


Fig 3.Confusion matrix-Multinomial NB

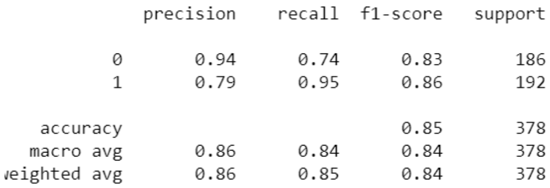


```
              precision    recall  f1-score   support

           0       0.85      0.84      0.85       186
           1       0.85      0.86      0.85       192

    accuracy                           0.85       378
   macro avg       0.85      0.85      0.85       378
weighted avg       0.85      0.85      0.85       378
```

Fig 4. Classification Report - Random Forest



```
              precision    recall  f1-score   support

           0       0.94      0.74      0.83       186
           1       0.79      0.95      0.86       192

    accuracy                           0.85       378
   macro avg       0.86      0.84      0.84       378
veighted avg       0.86      0.85      0.84       378
```

Fig 5. Classification Report - Mulnominal NB

From these values, we try to get the classification report by finding values like accuracy, precession, recall and F1 score values. The formulas to find these are as follows:

Accuracy :

(TN+TP) / (TN+TP+FN+FP)

Precession :

(TP)/(TP+FP)

Recall :

(TP) / (TP+FN)

F1 Score

2* [(Precision *Recall) / (Precision + Recall)]

*TP- true positive
*FP- false positive
*TF- true negative
*FN- false negative

Conclusion:

We trained our model using bow (bag of words)vectors for the prediction of stock markets. After vigorous training using algorithms, our model could produce predictions with accuracy up to mining (OM) approaches, and various techniques used in this field.

References

*Godbole, N., Srinivasaiah, M., & Skiena, S. (2007). Large-Scale Sentiment Analysis for News and Blogs. *Icwsm*, *7*(21), 219-222.

*Balahur, A., Steinberger, R., Kabadjov, M., Zavarella, V., Van Der Goot, E., Halkia, M., ... & Belyaeva, J. (2013). Sentiment analysis in the news. *arXiv preprint arXiv:1309.6202*.

*Vinodhini, G., & Chandrasekaran, R. M. (2012). Sentiment analysis and opinion mining: a survey. *International Journal*, *2*(6), 282-292.

*Mehler, A., Bao, Y., Li, X., Wang, Y., Skiena, S.:Spatial analysis of news sources. IEEE Trans.Visualization and Computer Graphics 12 (2006) 765–772

*ArXiv. (n.d.). Retrieved May 4, 2022, from https://arxiv.org/ftp/arxiv/papers/1309/1309.6202.pdf