ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/346089339

NLP Pranshu Tiwari

Experiment Findings · November 2020
DOI: 10.13140/RG.2.2.23893.04322

CITATIONS
0

READS
229

1 author:

Pranshu Tiwari
IBM

5 PUBLICATIONS **0** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project NLP for Stock Market Prediction View project

Project Ethical & Legal Issues in IoT and Artificial Intelligence -Case Study on Health Insurance View project

All content following this page was uploaded by Pranshu Tiwari on 23 November 2020.

The user has requested enhancement of the downloaded file.

Comparison of Different Natural Language Processing Models and Neural Networks for Predicting Stock Prices: A Case Study

Pranshu Tiwari,
Northeastern University

***Abstract*—The dataset consists of 25 heading of newspapers from 2008-08-08 to 2014-12-31. The data is transformed, cleaned to produce two datasets containing numeric values. One Data set consist of difference of count of Positive and Negative words creating a sentiment score for each News Paper for each Date. Then models are created to compare Linear Discriminant Analysis, Quadratic Discriminant Analysis and Logistic Regression. Similarly another data set is created where words are converted numeric values through one hot encoding and then used embedded size construct to pass through RNN many to one Architecture. Further This paper then considers inclusion of financial data and RNN architecture (Many to One RNN) and many to Many Architecture, We then compare different models to find the accuracy of models. The model wants to evaluate how much of text analytics contributes directly to the label. The second part of paper investigates if Sentimental score along with Financial data increases the accuracy of the Prediction**

I. Introduction

[Science Direct Volume 55] refers that stock price prediction based on textual information in financial news can be improved. Accordingly, they enhance existing text mining methods by using more expressive features to represent text and by employing market feedback as part of their feature selection process. Lot of research has been done in this area and this paper presents the different modelling techniques on text mining as well as deep learning based on text as well as market information

II. Novelty of Paper

Earlier techniques provided [Shreymas et.al] created a positivity, negativity Neutrality, Objectivity and Subjectivity for each Day They have combined the data with volume of Stock price, Opening and closing Price to compute and predict the label based on Random Forest, XGB Boost and Principal Component Analysis and Linear Discriminant Analysis . However they combined news of each day into one news segment. This approach may not work as different heading in reddit post have different viewership and it standardizes the viewership on the website.We use Natural Language processing by combining news of a given day as well as keeping separate headlines for each day. This is because each headline may have a higher visibility as compared to other news.

Our analysis consists of both text mining at both news level as well as combined perception of news for each day. The first part of analysis is solely based on text mining to compare how much of stock prediction can be only associated to text. This shows how text mining using Neural Networks and Sentimental analysis can predict directly the stock ending gain /loss based on heading of Newspaper.

In-order to improve accuracy further we then use the other attributes like volumes of Stocks, Open, High, Low, Close Volume and then again compare the analysis However later on we have combined sentimental analysis along with financial data from S&P[S&P data] which has been done by different researchers. In particular we, want to examine the accuracy of RNN prediction on Output label as well Stock Price Prediction for test data

III. Dataset Description

The dataset 1 consist of 1989 observations with p predictors. Hence if we create a matrix we end up creating a data matrix [n*p matrix].

S.No	Label	Type of Variable
1	Date	Numeric Character
2	Indicator for High Or Low Stock (1 or 0)	Nominal Variable
3	News Paper heading 1	String
4	News Paper heading 2	String
i	News Paper heading i	String
25	News Paper Heading 24	String

TABLE I: Data Set Description

This data set is transformed by creating a function which creates a numerical value associated to each sentiment. The net sentiment score is number of Positive words used – number of negative word used in newspaper. Each predictor is associated a value of sentimental score based on count of words [Positive]- Count of Words Negative. This database is called Sentimental Score Data. Hence, we end up creating a matrix n*p matrix with sentimental score for each headline news for the day. Since there are 24 headlines in reddit news there are sentimental score for each of 24 headlines.

IV. Design & Construct

A. Text Data only

The function of Sentimental Score is calculated as below

$$Sentimental\ Score\ (i,p) \sum Positive\,Words - \sum Negitive\,Words \tag{1}$$

for i day heading at p th time /location of website. Please note there are 25 news each day

```
def function(words):
            count=0
            for word in words:
                 if word in p3:
                   count=count +1

            return count

def something(words):
            count2=0
            for word in words:
               if word in q3:
                   count2=count2+1
            count3= function(words)-count2
            return count3
```

Fig. 1: Function for Sentimental analysis for each heading of news

P3= [‘Word 1’,’Word2….. Word n] which corresponds to list of positive words[Github,1]

Q3=[‘Word 1’,’Word2….. Word n] which correspond to negative words[Github,2]

Once the data is created we use Linear Discriminant Analysis , Logistic Regression and Quadratic Discriminant Analysis.

Let X= M -n*p matrix where n=1989 list of observations and p represents the headings of news paper. Referencing Figure 1 we create a frequency of words for each heading in reddit news which creates a Net Score as evident in Function in Figure 1. Hence we get a numeric matrix X.

We now create pooled covariance matrix for

$$\sum = \sum_{k=1}^{K} \sum_{i=1}^{N_k} (n_{ik} - 1) * \sum k \tag{2}$$

$n_{ik}\ -\ number\, of\, observations\, in\, class\, k$

$k\ -\ class$

$K\ -\ Total\, classes$

$i\ -\ Observations\, belonging\, to\, class\, k$

$\sum k\ =\ X{\cdot}t(X)\ \ where\ X\ is\ matrix\ for\ class\ k$ $containing\, n_k\, observations\, with\, p\, predictors.$

Formally the multivariate Gaussian density is defined as f(x) where X-is multivariate Gaussian Distribution representing sentimental score from Reddit website at different site position and time with $\mu\, mean\, and \sum\, is\, the\, p*p\, covariance\, matrix.$ f(x) denotes the probability density function for x vector with p length

Construct of Model 1-This is the odd ration of a output category depending on X matrix

$$M1 \sim (\log(\frac{p(X)}{1-p(X)})) = \beta_0 + \beta * X \tag{3}$$

where: where is n*p size and β is column vector of size p ..eq 6

Construct of Model 2:

$$f(x) = \frac{1}{2*\pi*\sum^{0.5}} exp(-\frac{1}{2}(x-\mu)^T * (x-\mu)) \tag{4}$$

$$\theta(x) = x^T \sum^{-1} * \mu_K - 0.5 * \mu_K^T * \sum^{-1} * \mu_K + \log(\Pi_k) \tag{5}$$

$$\Pi_k = P(Y=k) = \frac{Count\, of\, k}{n} \tag{6}$$

μ_k: lass Specific or Label Specific Mean vector. Hence if there are p predictors μ_k will have a vector length p

Leveraging Equation 3,4,5 we arrive at $\theta(x)$ *which separates into different labels.*

Construct of Model 3:

$$M3\ :\ \theta(x) = x^T \sum\nolimits_k^{-1} * \mu_K - 0.5 * \mu_K^T * \sum\nolimits_k^{-1} * \mu_K + \log(\Pi_k) - 0.5 \log(\sum\nolimits_k) \tag{7}$$

Similarly, we can create model on QDA Quadratic Discriminant Analysis leveraging equation where $\sum_k$ is covariance of each sub-class [Each sub class refers to output Label Stock Closure]

Modelling Characteristics

S. No	Model	Input Parameters	Intermediate State Model	Target State	Summary	Characteristics
1	M1	Text T for each part of website	Sentimental Score(S)creating a Matrix X-Equation 1	Y (based on equation 2)	Logistic Regression based on news headlines	X is n*p where p is score for day or position on site
2	M2	Text T for each part of website	Sentimental Score(S)creating a Matrix X-Equation 1	Y (based on equation 3 ,4,5)	Linear Discriminant Analysis	X is n*p where p is score for day or position on site
3	M3	Text T for each part of website	Sentimental Score(S)creating a Matrix X-Equation 1	Y (based on equation 6)	Quadratic Discriminant Analysis	X is n*p where p is score for day or position on site

TABLE II: Modelling based on Sentimental Score of News paper Heading for that day only

Confusion Matrix /Output on Test Data

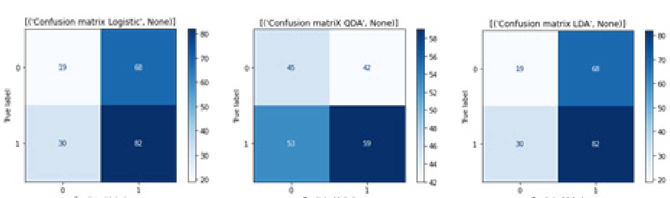

Fig. 2: Confusion Matrix based on Logistic ,QDA and LDA respectively

Accuracy on Test Data

The accuracies on test data of sample test size was about 64-65%

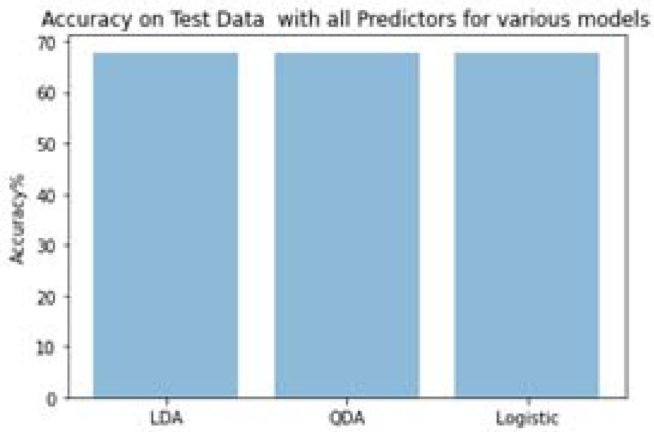

Fig. 3: Accuracy of all of them were about 60%-64%

Drawback of this approach.

1) The drawback of sentimental score is it does not consider the position of words and grammatical rules to analyse each news. [Sentiment Analysis and Subjectivity ,Bing Liu,UIC] The study done in UIC observes that sentimental analysis based on frequency of word count does not cover the following:
   a) Two Negative connotation can make a Positive sentiment
   b) Relative decrease in negative emotion-Example :Decrease in Death count is mild positive news
2) [Sentiment Analysis and Subjectivity ,Bing Liu,UIC] mentions that ”Clearly negation words are important because their appearances often change the opinion orientation. For example, the sentence “I don’t like this camera” is negative. However, negation words must be handled with care because not all occurrences of such words mean negation. For example, ”not” in ”not only . . . but also” does not change the orientation direction”
3) This model does not consider the previous news data. There may be a case that previous news may be more overwhelming that current news and hence previous days news would pull stock in same direction

B. Textual Analysis Using Neural Network Based Analysis

(Text Based Analytics Using Embedding Size and Encoder through LSTM process and Sigmoid Function) Text mining is the process of converting unstructured text /word data to numeric variables which act as independent variables or covariates to predict the response variable. We first do feature engineering by removing all stop words, punctuations and then convert the words to vector of Numbers using Embedding Matrix. We prefer one hot coding method over other natural language processing algorithms. Example: Term Documentation Matrix (frequency of words-based method is an existing approach to create a matrix per document or review. However the draw back of this approach the numeric are based only on frequency, not on position of words, meanings of similar words. - TF-IDF is based on the bag-of-words (BoW) model, therefore it does not capture position in text, semantics, co-occurrences in different documents, etc.

Word 2 Vector Architecture

In this case Word 2 Vec Architecture has size(—v—*N) where v-is the vocabulary size of All the headlines and N is the number of dimensions which we want to represent the word vector.
Each word has a one hot coding and we pass that word vector through embedding matrix which has dimensions (Vocabulary Size, Number of Dimensions of each word). A dot product is performed to extract the corresponding hot encoding of the particular word. This is passed through Neural Architecture. The final Neuron Network layer will have a Sigmoid Activation function for categorization of Output Label.

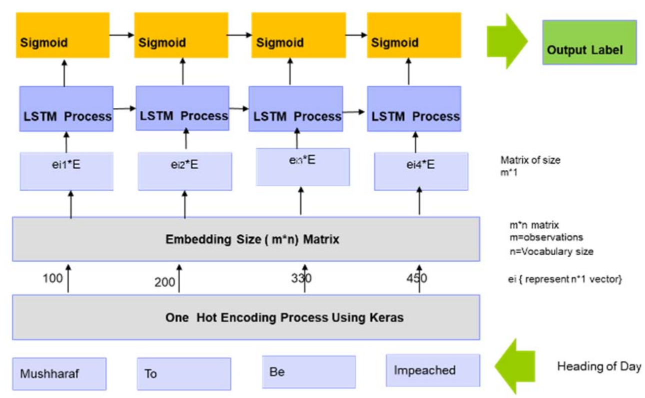

Fig. 4: Word 2 Vector is passed through RNN architecture

When we leverage RNN only on text analysis while ensuring the total Text Length from all headlines of day is 791- we get an accuracy of 57% on test Date.
We first do feature engineering by removing all stop words, punctuations and then convert the words to vector of Numbers using Embedding Matrix.
Mathematically the model works in the following way Perceptron’s Training using RLU activation.

$$w_0 + w_1x_1 + w_2x + w_nx_n = w^T x \tag{8}$$

Then We vary the weights through the vector transformation as below:

$$w_{J+1} = w_J + (y - y_1)x_i \tag{9}$$

$$l(w) = -\sum (y, -y)x_1^T w \tag{10}$$

Loss function for weights and we keep on changing weights till $\frac{\partial}{dw} l(w)' = 0$

$$w = w + \eta \sum_{1}^{n} (y_i - y)x \tag{11}$$

$$y = step(w_0 + w_1x_1 + w_2x + w_nx_n) = w^T x \tag{12}$$

The optimised weights can be achieved through batch Gradient Process.
Finally, in last neural network layer we activate sigmoid activation function which would help is classification of Output Label. Cross Entropy Function is optimized using Batch Gradient Optimization Process to arrive at optimum weights. This model works well if we want to predict the label given stock news for that day.

S. No	Model	Input Parameters	Intermediate State Model	Target State	Summary	Characteristics
1	M4	Word Index denoting matrix e(i)	e(i)*E E-Embedding Matrix	Y (based on equation 2)	Loss Calculation Entropy Loss. Shows Accuracy of 56%	Sigmoid Activation Function Calculating Output label after all the key words have been published for that day. Batch Gradient Process

TABLE III: RNN based Model using Word 2 Vector Architecture

Drawbacks of this model

While this model considers the position and spacing of word -this model does not consider previous days post. Hence in case retrospectively there has been a major factor to increase the volume of trade and increase in close value of the stock.

Results & Summary of Model Loss value at training data with various epochs

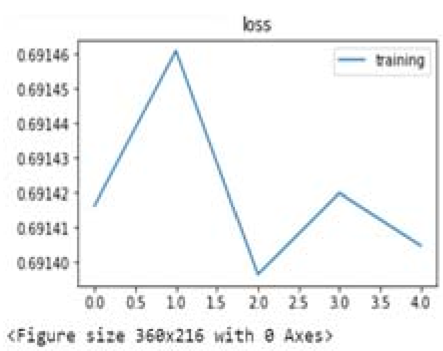

Fig. 5

The overall accuracy of the model is 56% on Test Data

C. Inclusion of Financial Data in the existing Data Set

The next part of analysis includes both textual analysis as well as time-series financial data

1	Date	Categorical
2	Stock Open	Continuous
3	Stock Close	Continuous
4	Stock High	Continuous
5	Stock Low	Continuous
6	Sentimental Score	Continuous

TABLE IV: Features for Model

Explanatory Analysis with Financial Data

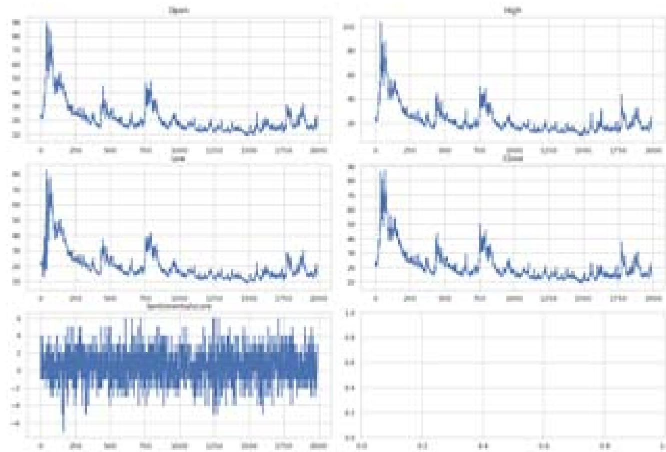

Fig. 6: Explanatory Analysis

Model 5, Model 6 & Model 7
RNN based Modelling Considering Financial Characteristics along with Sentimental Score including last 4 days data. [DataScience:RNN,2] Long Short-Term Memory (LSTM) maintains a cell state as well as a carry for ensuring that the signal (information in the form of a gradient) is not lost as the sequence is processed. Hence this model is used in model 4, 5 and will also be used in Model 6.If we remove the first feature (Date) , we can create a Matrix X of size 1989*5 with 5 features. However since we have to take data for first 5 and predict the 6 th Label . Hence we need to reshape the matrix to 1989*1*5 and then add feature data @t-5,t-4,t-3,t-2,t-1,t to dimension 2 to understand the patterns. This will follow Many to one RNN architecture

S. No	Model	Input Parameters	Hyper Parameters	Target State	Summary	Characteristics
1	M5	Input Steps of 1 S-sentimental Sc X- Financial Feat	LTSM	Y(t+1)	Loss Calculation Entropy Loss. Shows Accuracy of 56%	SEQ Length of 1
1	M6	Input data for 5 time-steps: S-sentimental Sc X- Financial Feat	Sentimental Score of words [Equation 1] LTSM technique with Sequence Length Data	Y(t+5)	Loss Calculation Entropy Loss. Shows Accuracy of 56%	Calculates Y for t+5 given X for t,t+1,t+2,t+3, t+4
1	M7	4 Time steps data S-sentimental Sc X- Financial Feat Y -output (t) is merged with X	LTSM technique for 4 time step	Y(t+4)	Loss Calculation Entropy Loss. Shows Accuracy of 56%	Calculates Y for t+4 given X for t,t+1,t+2,t+3,

TABLE V: LTSM with different time step memory using RLU activation in initial layers & Entropy Loss as Optimization Parameter using Batch Gradient Process

Confusion Matrix: The below plot represent confusion matrix for 3 models. Please note that test size varies as the time-steps/sequence length changes as per the model selected.

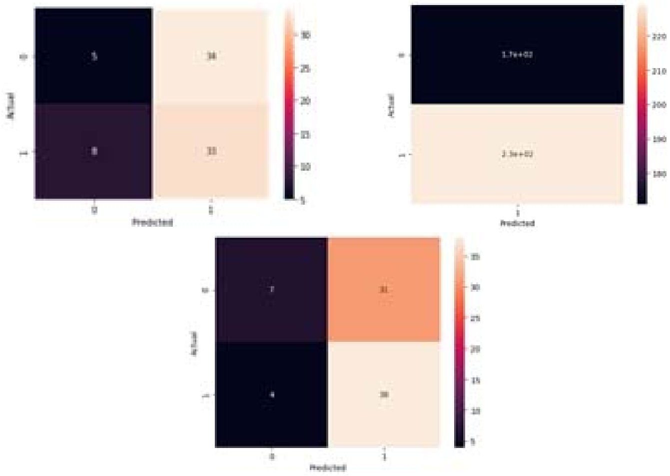

Fig. 7: Confusion Matrix

Model 7 RNN analysis LTSM technique considers the output variable as input variable each state/time as input variable

By this methodology we will be considering output Y label as input as well and will be using many to many RNN architecture. Hence in this case X matrix would include Output label as a learning parameter to predict the 5th label. In this case we used timesteps of 4 to predict 5th label. Hence unlike previous Model , XN matrix would concatenate X matrix [n*5] and Y[n*1] feature. Then we reshape XN matrix to (-1,4,6) which would be fed in the LTSM model.

Drawback of the model

1) Does not consider the y label of t-5,t-4,t-3,t—2,t-1 ,t respectively. It just predicts label y for t+1 given the conditions at t to t-5 points. However, y is itself dependent on Stock open and Stock closure values and hence they together will have some correlation

Accuracy Comparison of Model 5, 6, 7 on Test Data is as below

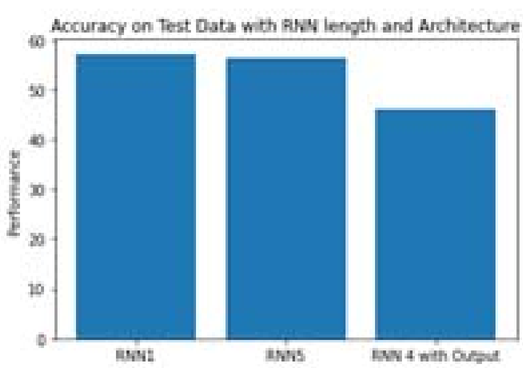

Fig. 8: Accuracy of Prediction of Output label

Leveraging Neural networks, we are achieving up to maximum 60% of accuracy based on textual and financial data

Now output label of reddit news is 0 or 1. When stock is higher than day start it is 1 and when close is less than start is zero. Towards this another approach is predicting Price output and that can automatically help us decide if it is higher or lower than day start. This can be done by removing y label and predicting the X(Financials for next day) for a batch of n timestep

S. No	Model	Input Parameters	Intermediate State Model	Target State	Summary	Characteristics
1	M8	X-Financial Feature for t,t+1,t+2,t+3, t+4	LTSM Inclusion of 3 dimension matrix to include time steps	X -Financial (t+5)	MSE-Mean Square Error Loss	SEQ Length of 5

TABLE VI: RNN Architecture using L2 Loss as Optimizer Function optimized through Batch Gradient Descent Methodology. The input data for 4 time-step data is considered as input to predict output

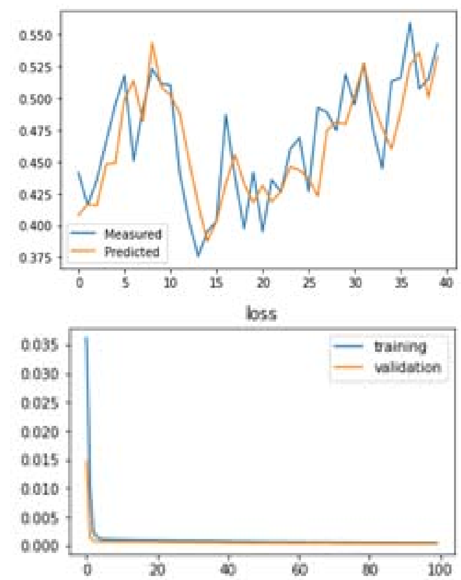

Fig. 9: Confusion Matrix

V. Recommendation

The following recommendation based on the above Study

Use Case	Mathematical Construct	Rationality
User wants to predict label (Label indicates if closure is higher than start price) by textual data/ News only.	Leverage Logistic Regression/LDA based on sentimental Analysis Use RNN one hot encoding and Embedded Size Matrix to predict Stock Label	Both Models have similar accuracy of test data Logistic Regression is having slightly better performance potentially because sentiments are captured for each news. The 25 news are segments on viewership. RNN uses hot encoding for entire day than for each news segment However past day news is not considered which will lead to some inaccuracy
User wants to predict Stock Price only for Next Day	Leverage RNN architecture using MSE and adam optimizer with sequence some sequence length	Based on training data the next day price is predicted with very high accuracy
User wants to predict label based on textual data and Sentimental Analysis	Leverage RNN architecture with Adam Optimizer and Sigmoid function with 5 Sequence Length /Time Steps . No need to input Y label as input	Time Steps approach works has about 58% accuracy

VI. Future Research and Limitations of the Research

The following additional methods could be further used to enhance the study

(a) Leverage Cross validation Approach to find the optimum time steps as well use other functions like RELU to bring accuracy of model
(b) Use Two Step based LTSM technique to predict Output Variable. This could be done by merging LTSM prediction of text data separately using embedded size and one hot encoding and LTSM technique for predicting the output label using financial data only
(c) Leverage Logistic Regression for both Financial & Text Data as well as consider time-series ARIMA models

VII. Conclusion

A. Stock Prediction

This paper presents two different approaches of doing natural language processing for each part of heading as well as for the day. Further studies needs to be done to improve the model prediction. This paper wants to show that stock prices can be predicted through newspaper headlines as they capture the macro-economic and politcal context which has a strong bearing to stock prices. Further study and machine learning algorithms can be developed by data scientist to improve the accuracy of model. We get high accuracy of predicting financial features for next day leveraging RNN many to many architecture.MSE¡5% on training data

B. Label Prediction

We get 67% accuracy through Probabilistic Models. However, it is to be noted those models consider sentimental analysis of each news. However, if we sum the sentimental score for each day and use financial data to predict the future data we receive 58% accuracy through RNN architecture

References

[1] Minqing Hu and Bing Liu. ”Mining and Summarizing Customer Reviews.”,*Proceedings of the ACM SIGKDD International Conference on Knowledge, Discovery and Data Mining (KDD-2004)*, Aug 22-25, 2004, Seattle, Washington, USA,
[2] Bing Liu, Minqing Hu and Junsheng Cheng. ”Opinion Observer: Analyzing and Comparing Opinions on the Web.” *Proceedings of the 14th International World Wide Web conference (WWW-2005)*, May 10-14, 2005, Chiba, Japan.
[3] Will Koehrsen. ”Recurrent Neural Networks by Example in Python” *Towards Data Science*, Nov 15, 2018
[4] Bing Liu. ”Sentiment Analysis and Subjectivity” *Handbook of Natural Language Processing* Second Edition, 2010

View publication stats