**INTRODUCTION**

1.1 **Stock Market Prediction**

Stock market investment is a process of predicting the future values of stocks.The investment decisions are mainly driven by the market information available to investor.These predictions helps any significant market to yield profit.

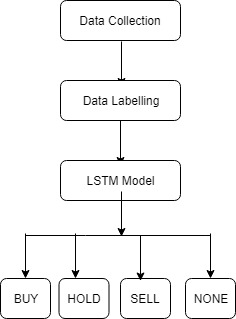
1.2 **More about the Project**

Predicting the Stock Market has been the bane and goal of investors since its existence.Entire day companies rise and fall everyday based on the behaviour of the market.Despite its prevalence,it is quite secretive and deceptive art. The chief goal of this project is to enhance the understanding of stock market prediction.The better understanding includes how the market moves ,what investors thinks about every ups and downs of stocks.In this project the main motive was to predict the prediction of stocks through social network analysis. Basically social network analysis means we looked out for comments, messages or reviews on the website moneycontrol.com where many investors related to particular market predict the stock movement through commenting their views.These comments highlight three basic prediction either to “Sell”or”Buy”or “Hold”. The project will make no attempt to deciding how much money to allocate to each prediction.

**Objective**

To predict the stock movements using LSTM network.

**Proposed Model**



**DATASETS AND FEATURES**

**2.1 Data-set and Preparation**

The Data-set was collected from moneycontrol.com by using one handy software called Octo-parse which has a feature of web scrapping.The four different company’s stocks were scrapped Airtel,PC jeweller,Bajaj Finance and Tata\_Steel.

**Columns Extracted:-**

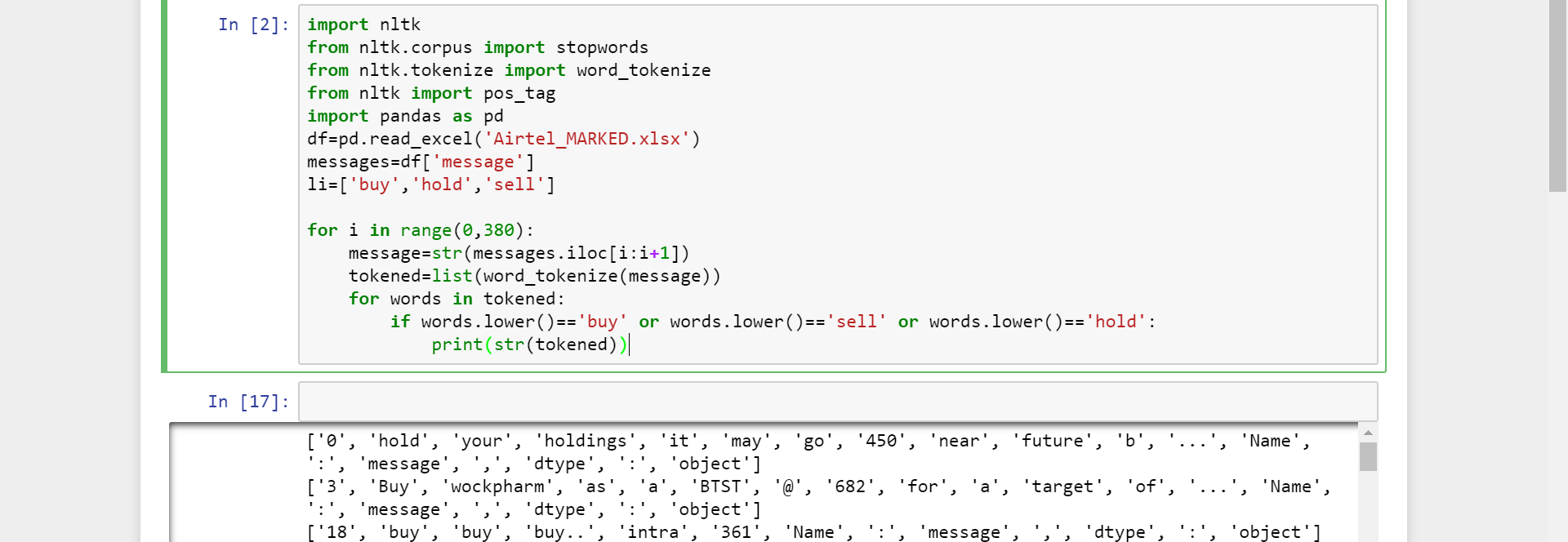
user\_name,num\_messages,stocks\_predicted,user\_level,followers,messages,time ,ratings and offensive.

The main feature that was used in this project was ‘messages;.

**2.2 Process of Labeling**

**Two ways of labeling:-**

**1.By coding in Python**



**2.Manual labeling**

1. By reading the messages which were left and noticing the target that has been posted by the particular user.
2. If the target given by the user is greater than the current price then-BUY and vice versa for SELL.
3. If the target given by the user is near about the current price in range(+ 5or -5) then HOLD.
4. If the message doesn't seems to give any sort of information then NONE.
5. Label Code word used for BUY-0,HOLD-1,NONE-2,SELL-3 as deep learning algorithm can only understand numbers.

**Overview of Data -Sample**



**Methods**

**What exactly Deep Learning is?**

-Deep learning is an aspect of artificial intelligence ([AI](https://searchenterpriseai.techtarget.com/definition/AI-Artificial-Intelligence)) that is concerned with emulating the learning approach that human beings use to gain certain types of knowledge. At its simplest, deep learning can be thought of as a way to automate [predictive analytics](https://searchbusinessanalytics.techtarget.com/definition/predictive-analytics).

A type of advanced machine learning algorithm, known as neural networks, underpins most deep learning models. Neural networks come in several different forms, including recurrent neural networks, convolutional neural networks, artificial neural networks and feed forward neural networks, and each has their benefit for specific use cases. However, they all function in somewhat similar ways, by feeding data in and letting the model figure out for itself whether it has made the right interpretation or decision about a given data element.

Neural networks involve a trial-and-error process, so they need massive amounts of data to train on. It's no coincidence that neural networks became popular only after most enterprises embraced big data analytic and accumulated large stores of data. Because the model's first few iterations involve somewhat-educated guesses on the contents of image or parts of speech, the data used during the training stage must be labeled so the model can see if its guess was accurate. This means that, though many enterprises that use big data have large amounts of data, unstructured data is less helpful. Unstructured data can be analyzed by a deep learning model once it has been trained and reaches an acceptable level of accuracy, but deep learning models can't train on unstructured data.

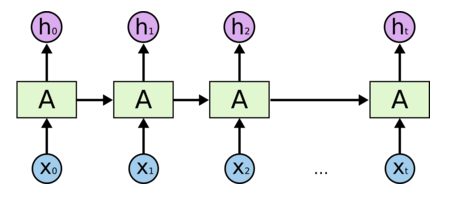
### **Limitations of deep learning**

### The biggest limitation of deep learning models is that they learn through observations. This means they only know what was in the data they trained on. If a user has a small amount of data or it comes from one specific source that is not necessarily representative of the broader functional area, the models will not learn in a way that is generalizable.

Sequence prediction problems have been around for a long time. They are considered as one of the hardest problems to solve in the data science industry. These include a wide range of problems; from predicting sales to finding patterns in stock markets’ data, from understanding movie plots to recognizing your way of speech, from language translations to predicting your next word on your phone’s keyboard.

With the recent breakthroughs that have been happening in data science, it is found that for almost all of these sequence prediction problems, Long short Term Memory networks, a.k.a LSTMs have been observed as the most effective solution.

LSTMs have an edge over conventional feed-forward neural networks and RNN in many ways. This is because of their property of selectively remembering patterns for long duration of time.The purpose of this article is to explain LSTM and enable you to use it in real life problems.



A typical LSTM network is comprised of different memory blocks called **cells**(the rectangles that we see in the image)**.** There are two states that are being transferred to the next cell; the **cell state** and the**hidden state**. The memory blocks are responsible for remembering things and manipulations to this memory is done through three major mechanisms, called **gates.**

**Structure Of LSTM Model Created For This Project**

* Tokenizer API
* Encoding with one\_hot
* Sequence Padding
* SMOTE
* Embedding Layer
* Activation Function('Softmax')
* Loss Function( 'Categorical\_crossentropy')
* Optimizer Function('RMSprop')

**EXPERIMENTS AND RESULTS**

|  |  |
| --- | --- |
| **Language** | **Python 3.5** |
| **Framework** | **Keras** |
| **Windows** | **10** |
| **Software** | **Jupyter Notebook** |

Fig.1 Airtel

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Types of Data | Activation function and loss | Optimizer | Epochs | Batch\_size | F1\_score | Precision | Recall |
| Airtel(replicated) | Softmax& categorical | RMS\_prop | 75 | 20 | 0.82 | 0.84 | 0.85 |
| Airtel(SMOTE) | Softmax& categorical | RMS\_prop | 50 | 16 | 0.66 | 0.66 | 0.66 |
| Airtel(without SMOTE) | Softmax& categorical | adam | 51 | 20 | 0.70 | 0.68 | 0.73 |
| Airtel(replicate with k-fold=10) | Softmax& categorical | RMS\_prop | 50 | 20 | 0.95 | 0.97 | 0.95 |

Fig.2 Tata\_steel

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Types of Data | Activation function and loss | Optimizer | Epochs | Batch\_size | F1\_score | Precision | Recall |
| Tata\_steel(replicated) | Softmax& categorical | RMS\_prop | 75 | 20 | 0.82 | 0.83 | 0.85 |
| Tata\_steel(SMOTE) | Softmax& categorical | RMS\_prop | 50 | 16 | 0.49 | 0.51 | 0.51 |
| Tata\_steel(without SMOTE) | Softmax& categorical | adam | 75 | 20 | 0.69 | 0.65 | 0.70 |
| Tata\_steel(replicate with k-fold=10) | Softmax& categorical | RMS\_prop | 50 | 20 | 0.93 | 0.95 | 0.96 |

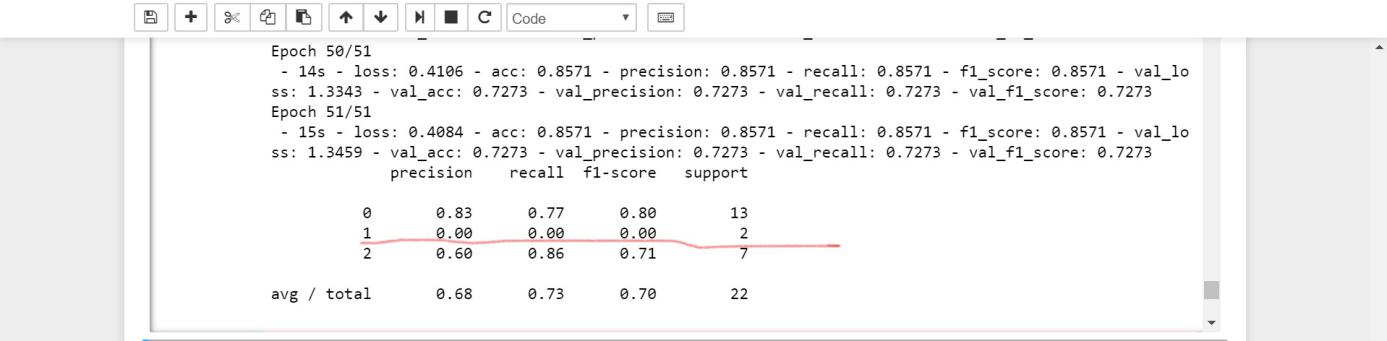
Fig.3 Pc\_Jewellers

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Types of Data | Activation function and loss | Optimizer | Epochs | Batch\_size | F1\_score | Precision | Recall |
| Pc\_Jewellers(replicated) | Softmax& categorical | RMS\_prop | 75 | 20 | 0.79 | 0.80 | 0.82 |
| Pc\_Jewellers(SMOTE) | Softmax& categorical | RMS\_prop | 50 | 16 | 0.54 | 0.59 | 0.58 |
| Pc\_Jeweller(without SMOTE) | Softmax& categorical | adam | 51 | 20 | 0.61 | 0.60 | 0.60 |
| Pc\_Jewellers(replicate with k-fold=10) | Softmax& categorical | RMS\_prop | 50 | 20 | 0.90 | 0.91 | 0.91 |

**STEPS INVOLVED IN EXPERIMENTS:-**

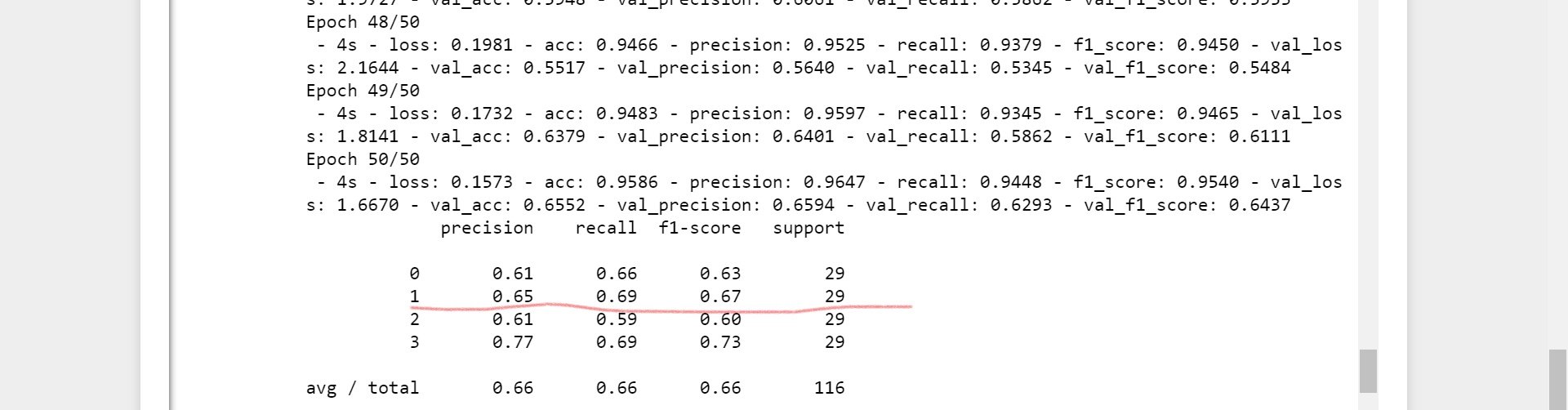
1.As we started the experiment with Airtel Data set we took the data set without applying smote and k-fold and using ‘adam’ as optimizer and ‘softmax’ and ‘categorical\_crossentropy’ as activation function and loss function respectively. The result that we found is shown in table 1. But we faced a problem i.e the class ‘HOLD’(HOLD-1) didn’t show any result. As you can see down here in the image class 1 holds no calculation

for precision ,recall and f1-score.



2.So next step that we took to solve this problem was we applied k-fold in our fold but the same thing happened with the HOLD class though there other class showed some better result but we didn’t get our desired result.

3.In next step we applied SMOTE to the data set and by the help of this method we actually got better result i.e the class HOLD had some calculation.



**CONCLUSION**

* In this project, we have built an algorithm for prediction of stock movements using LSTM model.
* We have calculated performance metrics of the system using this model.
* The experiment results show that precision, recall and f1-score values for SMOTE data with ‘Sofmax’ activation function and ‘Categorical’ cross entropy are 66%, 66% and 66% respectively.
* Data without SMOTE ‘Softmax’ activation function and ‘Categorical’ cross entropy are 68%, 73% and 70% respectively. Our next step is to further increase the performance metrics of the system.

