ANALYSIS AND FORECAST OF COMMODITY PRICE USING MACHINE LEARNING AND DEPLOYMENT IN WEB APPLICATION

VIKRAM VIJAYARAJ M, SRIRAM R, SAI PRASANTH G, SRIKARTHIKEYAN V,BTECH.
INFORMATION TECHNOLOGY,
PANIMALAR ENGINEERING COLLEGE,
vikramvijayaraj31@gmail.com , sriram135790@gmail.com ,prasa6167@gmail.com ,
riyan17082@gmail.com

ABSTRACT

Prediction of financial market accurately is certainly significant. Commodity price fluctuations affects the global economic activity. Earning in export industry rely mainly on primary commodity and the movements of commodity prices has significant impact on overall economic progress for all countries.
The method of forecasting plays a vital role in predicting adverse movements in case of commodity price prediction. In today's world, the growth in deep learning outperforms in several demonstration in fields of financial market analysis. In this paper, we present a productionized commodity price analysis and prediction using LSTM model.

- The LSTM model gathers data in a sequential order periodically of the commodity value.
- Datasets used in our model are collected in real time from Yahoo Finance via API to reduce local storage.
- The critical task is to transform a machine learning model into user accessible real time environment. [2]
- Manual conversion of a machine learning model source code into a software is a time-consuming and error-prone task.
- Investors and traders can buy and sell commodities directly in the spot market.

KEYWORDS

Price Prediction, Commodity, Deep Learning, Finance, LSTM, RNN, Django-REST, Deployment, Productionised ML Model.

I. INTRODUCTION

Finance plays a vital role in our day-to-day life and growth of economy can depend on precise prediction in the financial markets. A commodity is a basic good used in commerce that is interchangeable with other commodities of the same type. Analysts and economists still use manual methods in predicting financial movements using mathematical methods [1]. Traditional statistic tools like Naive Bayes algorithm, helps the financial industries to forecast the trends of financial assets. Financial fields should be handled cautiously as it may leads to capital loss. Deep learning is built on complex neural networks with deeper level of hidden layers. Deep learning network learn by discovering intricate structures in the data they experience. It is also known as deep structured learning.

II. METHODOLOGY

A. RNN

Neural Networks are a set of algorithms that mimics the human brain and are designed to recognize patterns. Recurrent Neural

Network (RNN) is type of neural network used to deal specifically with sequential data. The principle of RNN is storing the output of a specific layer and is given as aninput to the next layer to predict the outputof that layer.

B. LSTM

A modified RNN, Long Short-Term Memory (LSTM) is used for predicting the price of the commodity.

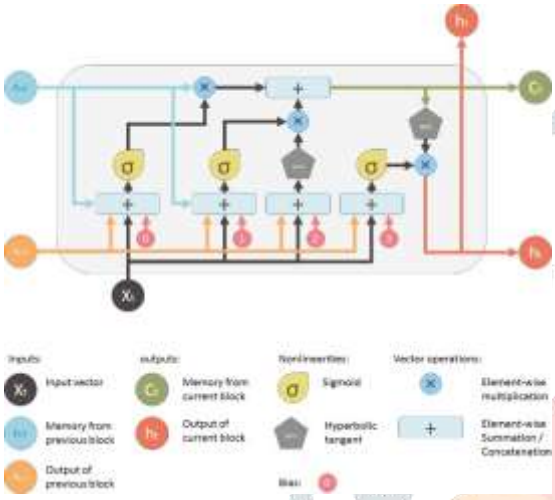

Fig. 1 (LSTM Flow Diagram)

LSTM model stores information sequentially over a period of time as shownin Fig. 1. LSTM model decides what information will be stored and what discarded. It is done using the “gates”.

III. ARCHITECTURE

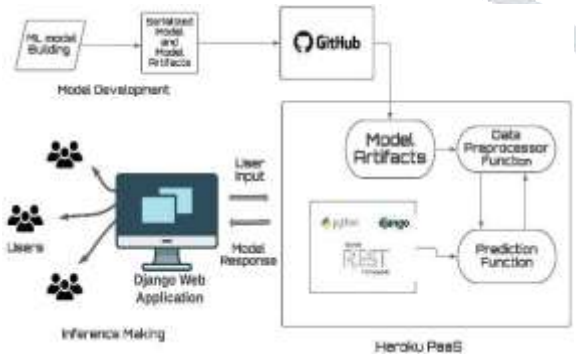

Fig. 2.a (Architectural Diagram)

- A machine learning model is build based on LSTM algorithm.
- The unordered data is reshaped into a chain of tokens from which the actual data is rebuild.
- The source code of the model will be available as a repository in GitHub.
- Refer Fig. 2.b for the ML model flow diagram.
- A Django web application is developed and integrated with the model.
- This web application is productionized to reach the user.

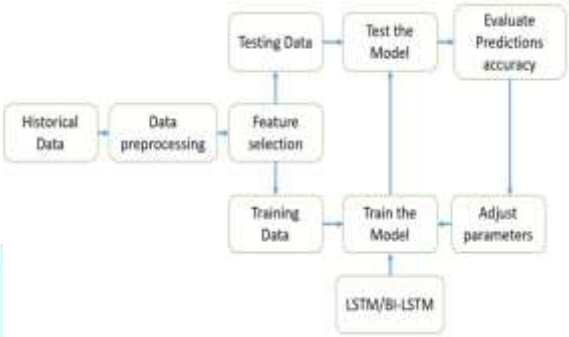

Fig. 2.b (ML Model Flow Diagram)

A. GATHERING DATA AND SCRAPPING PROCESS

1. SCRAPPING

Scrapping is the process of extracting data from a data source and processing it into a data that is suitable for the system to display.
The Web sources in online helps in extracting the price data that is used as data source/input. This data source is given as an input to the machine learning model where the get trained on top of this data source.

2. KERAS

An API which we used for extracting data from online sources is **Keras**. Keras has potential to decrease the amount of

cognitive load. It furnishes standard & simple APIs which is used to turn down the number of user actions and also packed with comprehensible & controllable error messages.
TensorFlow has full deployment capabilities that can be used by Keras.
Keras models are more scalable as it can be converted into a JavaScript file to run in web browsers.

B. DATA PREPROCESSING

Machine learning models can only be trained with the data having the suitable format but the data obtained from the web sources basically hold noises, missing values or Not a Number (NaN) values. To convert the data into a suitable format, a technique called Data pre-processing is used by which the data is cleaned and reformatted that leads to higher accuracy of the model.

C. FEATURE SELECTION

To reduce the computational cost and to increase performance of the model, the input variables are reduced. This can be achieved by Feature selection technique where the features are divided in such a way that it focuses more on the prediction variable.
This technique helps in three different ways before data modelling. They are,

- **Reducing Overfitting:** Decision making increases with decrease in redundant data.
- **Improves Accuracy:** Irrelevant and unwanted data attributes are removed to improve the accuracy.
- **Reduces Training Time:** As the number of input variable decreases the model takes less time to train.

D. TRAINING AND TESTING

Training set – The data subset used for training the model is termed as Training set.
Test set - The data subset used for testing the model is termed as Testing set.

The method to find the accuracy of themodel is **Train/Test set** method. The Train/Test set method generally splits the data into two sets. One is training set (80%)and the other is testing set (20%).

IV. DEPLOYMENT

This model is integrated into a website for which a high-level and rapid development python framework is used called **Django.**
Django is a well-structured framework thatis flexible and powerful enough for building advanced web APIs.

1. REST API

The API which we used in our proposal is **Representational State Transfer (REST)** which provides a connection between computer systems on the web (refer Fig. 2.a), which makes it easier for the systems and to communicate easily with each other while an **Application Programming Interface (API)** is an application used to interact with the components of the system and delivers the responses between users and the system.

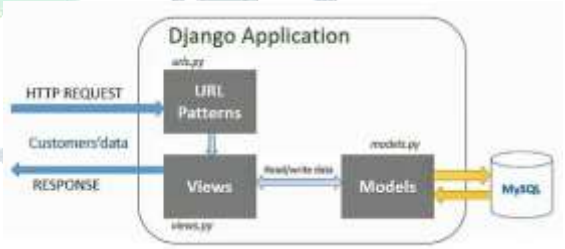

Fig. 3 (Django-REST Framework)

APIs are developed based on the collection of protocols that REST defines.
Whenever a **request** is made by calling the URL, a **response** is sent back as a JSON file (refer Fig. 3).

2. API REQUESTS

- **Endpoint**: Endpoints helps in interacting with server-side web APIs, as they denote stored resources which can be acquired by third party software.
- **Method**: A method is the request that is send to the server. GET and POST methods are the most commonly used methods.
- **Headers**: The REST headers contain information that helps to track down the issues when they are encountered.
- **The Data**: To get a response, the data is sent to an API request.

V. WORKFLOW

To deploy the model, we have trained via an API, we have to structure our API to:

- Accept data and pre-process it.
- Load the model we have trained using **Pickle**.
- Predict the outcome based on the data we have passed.
- Return the prediction and confidence score as a JSON (JavaScript Object Notation) file.

We created a virtual environment and installed the required libraries and it is activated. The required packages are installed using **pip** instead of installing them one after the other. The **Django** and **Django-REST** API are setup and navigated to the folder directory creating a Django app. The "rest_framework" and "API" files are added under the "installed_apps" list, under "model_deploy" folder.

Migrate the packages by running the server. Building the API by creating the simple index page with data pre-processing which converts variable to float data type since it is represented as string data type.

Configure the url path and create "urls.py" under the "model_deploy" folder, check if nothing breaks by running the server. Thus, the API is integrated into the website.

VI. RESULT

The end result of the machine learning model is integrated and deployed in a website. The interface of the web application is shown in Fig. 4 and Fig. 5.

ANALYSIS - The end user has access to set the time line of the dataset given to be trained in the model. User can also do their own research by referring the historical data provided in the website itself. Different commodities are listed in the side menu that can be selected based on user needs. Daily returns, daily change and cumulative returns are visualized in a graph chart. Assets high, low, open, close and adjusted close prices are displayed in table format.

Fig. 4.a (Analysis Interface)

FORECAST - The trained and tested value is visualized in charts. The output is also analysed with a real time dataset and suggested based on the forecasted values.

The charts are visualised holding the theme of providing user friendly, understandable analyzation to hold or close their commodities.

Fig. 4.b (Forecast Interface)

VII. CONCLUTION

From this project, it is evident that deep Learning algorithms plays a vital role in enhancing modern technologies. Investors and traders can attain much financial gains and recover the losses with the help of this website. As this website build upon this deep learning model is used in real time analysis and forecast. This website used as an analysis tool for Investors and Traders. Daily returns can be predicted by the individuals and investors accordingly.
Future prices of the assets will be forecasted for better investment.

REFERENCES

[1] Gold Price Forecast based on LSTM- CNN Model, Zhanhong He, Macau University of Science and Technology, Macau.

[2] Multi-target Compiler for the Deployment of Machine Learning Models Oscar Castro-Lopez Facultad de Informatica Universidad Autonoma de Sinaloa Culiacan, Mexico.

[3] Understanding LSTM and its diagrams, Shi Yan, 2016, Medium, https://miro.medium.com/max/2312/1*laH0_xXEkFE0lKJu54gkFQ.png

[4] Md. Arif Istiake Sunny. "Deep Learning-Based Stock Price Prediction Using LSTM and Bi-Directional LSTM Model", 2020 2nd Novel Intelligent and Leading Emerging Sciences Conference (NILES), 2020.