**Crude Oil Price Prediction**

**1.INTRODUCTION :-**

**1.1.Overview:-**

Crude oil is a naturally occurring petroleum product composed of hydrocarbon deposits and other organic materials. A type of fossil fuel, crude oil is refined to produce usable products including gasoline, diesel, and various other forms of petrochemicals. It is a [nonrenewable resource](https://www.investopedia.com/terms/n/nonrenewableresource.asp), which means that it can't be replaced naturally at the rate we consume it and is, therefore, a limited resource.It is the raw natural resource that is extracted from the earth and refined into products such as gasoline, jet fuel, and other petroleum products.It is a global commodity that trades in markets around the world, both as spot oil and via derivatives contracts.Many economists view crude oil as the single most important commodity in the world as it is currently the primary source of energy production.

**1.2.Purpose:-**

Crude oil is a liquid fuel source located underground. It is extracted through drilling. Oil is used for transportation, heating and electricity generation, varied petroleum products, and plastics.Crude oil is the base for many products. These include transportation fuels such as jet fuel, gasoline, and diesel fuel. They also include fuel oils used for both heating and electricity generation.Crude oil also creates petroleum products.Petroleum byproducts make tar, asphalt, paraffin wax, and lubricating oils.It is also used in chemicals, such as fertilizer, perfume, insecticides, soap, and vitamin capsules.Oil is the base for plastics used in everything from heart valves to plastic bags.﻿﻿ It's used in carbon fiber in aircraft, PVC pipes, and cosmetics.

**2.LITERATURE SURVEY:-**

**2.1.Existing problem:-**

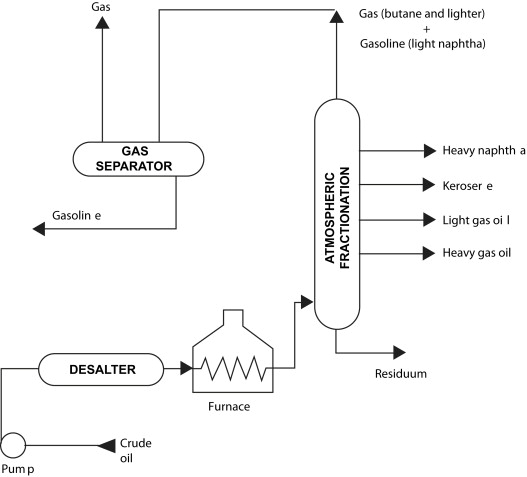
* **Higher prices: adverse impact on fiscal deficit :** India imports 1.5 billion barrels of crude oil each year . This comes up to around 86% of its annual crude oil requirement.
* **Impact on the rupee :** The rise in crude oil prices has a clear impact on the Indian rupee. On 24 May 2018, the rupee closed at 68.34 against the US dollar
* **Impact on Current Account Deficit (CAD):** India’s dependency on crude oil imports has only been increasing over the past few years. The dependency rose from 77.3% in FY2014 to 83.7% in FY2018.
* **Impact on inflation :** Oil is a very important commodity and it is required to meet domestic fuel needs. And in addition to that, it is a necessary raw material used in a number of industries.

**2.2.proposed solution:-**

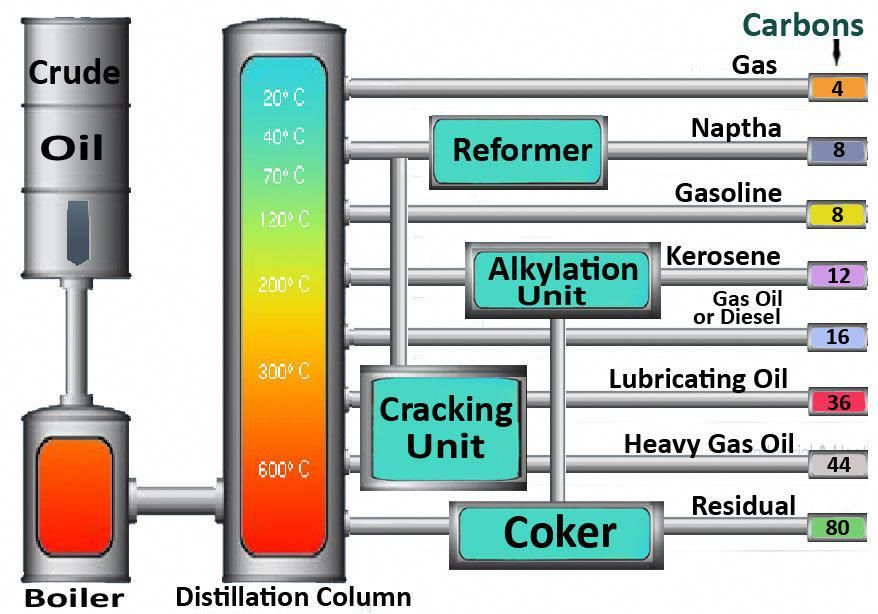
* High oil rate can be solved by decreasing the use of personal vehicle.Stop using vehicle in unnecessary work.Govt should import more and more oil fromForeign.Govt should manage this cost with other things and decrease the cost of oil.
* Lower oil prices mean less drilling and exploration activity because most of the new **oil** driving the economic activity is unconventional and has a higher **cost** per barrel than a conventional source of **oil**. ... Between the job losses and the capital losses, a dip in oil prices can trim the growth of the U.S. economy.
* **Countries** where fuel accounts for more than 90% of total exports include Algeria, Azerbaijan, Brunei Darussalam, Iraq, Kuwait, Libya, Sudan and Venezuela. For an idea of which economies **rely** most heavily on **oil**, this chart using 2012 World Bank data shows **oil** revenue as a share of GDP.

**3.THEORITICAL ANALYSIS:-**

**3.1.Block diagram:-**



**3.2.Technical Architecture:**



**3.3.Hardware/Software designing**

**software Requirements:**

* Os - Windows XP,7,8,10
* Jupyter software
* Spyder software
* Anaconda command prompt

**Hardware Components:**

* Processor - i3
* Hard Disk Storage - 10 GB

**4.EXPERIMENTAL INVESTIGATIONS:-**

There is one input layer, two hidden layers of GRU and one dense layer in our proposed model.illustrate its Tensorflow computation graph which indicate the network architecture in our method. Each input sample is a matrix of *n* × *m*, which is represented by a NumPy array. ‘*n*’ is the lagging order and ‘*m*’ is the number of IMFs and residue. By trial and error, we determine set the number of hidden neurons to 32 and MSE as the loss function. The optimizer of training is adaptive moment estimation (Adam) which solve the problems of other algorithm, such as learning rate disappearing, convergence slowly or loss function fluctuating greatly. The learning rate in following experiment is set to 0.01. We adopt the strategy of one day ahead prediction to carry out our tasks. In other words, the prices of the past *n* days (*p*1, *p*2, … *pn*−1, *pn*) is used to predict the price of the (*n* + 1)th day. Letter *n* is called the lag order which related to the size of neuron of the GRU. We adopt a strategy of grid search to determine the number of lagging order that is important for time series analysis. By trial and error, the lag order was set to 32.

**5.Result:-**

This crude oil price prediction process constitutes of two steps.These steps are mentioned below:

There are two inputs to be considered they are

* + Last ten times changed prices are taken as input from the user.
  + Crude oil price is shown as predicted.

The output is generated related to the analysis based on the data collected related to the prediction of crude oil price of last ten times changed prices.

**6.ADVANTAGES AND DISADVANTAGES:-**

**Advantages:**

* **Crude Oil** is Readily Available. ...
* **Oil** Has Lots of Uses. ...
* It Has a High Energy Density. ...
* **Crude Oil** Can Be Stored Easily. ...
* **Oil** Energy Can Be Constant. ...
* The **Oil** Industry Creates Jobs. ...
* **Oil** Energy Produces Toxic Gases. ...
* **Oil** Leaks Are Possible.

**Disadvantages:**

* **Oil** is a non-renewable source of energy. ...
* Burning **oil** produces carbon dioxide gas. ...
* Burning **oil** can pollute the air.
* Much of our **oil** has to be imported and it is becoming more and more expensive as reserves reduce and imports increase.

**7.Applications:-**

## **\* Plastics**

As noted above, plastic requires crude oil for its production. Since there are so many products that use plastic, this resource is in high demand. You’re probably reading this article on a device that uses plastic in its construction. You may be surrounded by plastic products in your home or at work. It’s hard to get away from this versatile product.

## **\* Clothing materials**

Petroleum is essential for the creation of clothing. It helps to make clothing that is colorful and non-flammable. Polyester, rayon, nylon, and some fake furs all use crude oil.

## **\* Products in your home**

Believe it or not, if you’re sitting on your couch or other furniture with a cushion, you’re probably sitting on a product created using crude oil. Polyurethane foam is a petroleum-based material that is often used for couch cushions. It’s durable, so it can take a beating from years of people sitting and laying on it while being lightweight. Also, your flooring could be another hidden crude oil product in your home. Both linoleum and carpet are made using petroleum

## **\* Food production**

You may want to cringe thinking of crude oil being used in the production of food items, but it is a necessary piece of the process. Some fertilizers used to grow your food need this product. Transportation is a more “tame” use for crude oil in the industry, with commercial vehicles delivering food all over the country. The packaging that your food comes in is also made using plastic.

## **\* Transportation**

Your car is another item that uses crude oil in the manufacturing process. Your car’s bumper, interior, and even engine block components are made with high-performance plastics. These are used because they are lightweight and can help improve the fuel economy of the car. They can also help to enhance the safety of the vehicle.

**8.Conclusion:-**

In this study, a hybrid model called CEEMDAN-ML-GRU for crude oil price forecasting is proposed. This model takes full use of the advantages of the signal processing algorithm CEEMDAN and the multi-layer gated recurrent unit networks (ML-GRU). As mentioned in the previous section, the hybrid model uses CEEMDAN to solve the non-stationarity problem of crude oil price data, and generalizes the nonlinear crude oil prices data by a multi-layered GRU neural network. We conduct a large number of experiments to verify the effect of the proposed method in forecasting task by using the WTI price data as sample data. The experimental results show that our proposed method goes beyond other traditional statistical methods, machine intelligent algorithms and other hybrid models, which include the EEMD-LSTM method proposed in 2019.

In addition to crude oil price forecasts, the introduced CEEMDAN-ML-GRU model can also be extended to solve other complex problems in other areas, such as time series forecasts or risk measurements in financial markets. The main purpose of this approach is to improve the accuracy of short-term crude oil price predictions and help decision makers minimize the risks of the crude oil market. However, the proposed method is mainly applied to short-term forecasts, so only daily data is used. If we need to predict long-term price trends, we need to combine this method with economic theory or measurement methods to play a greater role. This is exactly the research plan that we will follow in our future research.

**9.Future scope:-**

**Crude oil** consumption is expected to grow at a CAGR of 3.60% to 500 million tonnes by 2040 from 221.56 million tonnes in 2017. India's **oil** demand is projected to rise at the fastest pace in the world to reach 10 million barrels per day by 2030, from 5.05 million barrel per day in 2020.The EIA forecast that Brent **crude oil** prices **will** average $68/b in the third quarter of 2021 and $60/b in 2022. Prices are increasing due to higher demand as more people are vaccinated against COVID-19. OPEC is beginning to increase production after limiting it due to decreased demand for **oil** during the pandemic.

**10.BIBLILOGRAPHY:-**

[1] E. Regnier, "Oil and Energy Price Volatility," Energy Econ. **29**, 405 (2007).

[2] J. E. Rentschler, "Oil Price Volatility, Economic Growth and the Hedging Role of Renewable Energy," World Bank, Policy Research Working Paper [6603](http://large.stanford.edu/courses/2018/ph240/ezquerro1/docs/wps6603.pdf), September 2013.

[3] R. McNally, *Crude Volatility: the History and the Future of Boom-Bust Oil Prices* (Columbia University Press, 2017).

**Source code(Model Building):**

import numpy as np #used for numerical analysis

from flask import Flask,render\_template,request #Flask is a application used to run/serve our aplication

# request is used to access the file which is uploaded by the user in our application

#render\_template is used for rendering the html pages

from tensorflow.keras.models import load\_model #we are loading our model from keras

app = Flask(\_\_name\_\_) #our flask app

model = load\_model('crude\_oil.h5',) #loading the model in the flask app

@app.route('/') #rendering html template

def home() :

return render\_template("index.html") #rendering html template

@app.route('/about')

def home1() :

return render\_template("index.html") #rendering html template

@app.route('/predict')

def home2() :

return render\_template("web.html") #rendering html template

@app.route('/login',methods = ['POST']) #route for our prediction

def login() :

x\_input=str(request.form['year']) #requesting the file

x\_input=x\_input.split(',')

print(x\_input)

for i in range(0, len(x\_input)):

x\_input[i] = float(x\_input[i])

print(x\_input)

x\_input=np.array(x\_input).reshape(1,-1)

temp\_input=list(x\_input)

temp\_input=temp\_input[0].tolist()

lst\_output=[]

n\_steps=10

i=0

while(i<1):

if(len(temp\_input)>10):

#print("temp input",temp\_input)

x\_input=np.array(temp\_input[1:])

print("{} day input {}".format(i,x\_input))

x\_input=x\_input.reshape(1,-1)

x\_input = x\_input.reshape((1, n\_steps, 1))

#print(x\_input)

yhat = model.predict(x\_input, verbose=0)

print("{} day output {}".format(i,yhat))

temp\_input.extend(yhat[0].tolist())

temp\_input=temp\_input[1:]

#print(temp\_input)

lst\_output.extend(yhat.tolist())

i=i+1

else:

x\_input = x\_input.reshape((1, n\_steps,1))

yhat = model.predict(x\_input, verbose=0)

print(yhat[0])

temp\_input.extend(yhat[0].tolist())

print(len(temp\_input))

lst\_output.extend(yhat.tolist())

i=i+1

print(lst\_output)

return render\_template("web.html",showcase = 'The next day predicted value is:'+str(lst\_output))

#return str(x)

if \_\_name\_\_ == '\_\_main\_\_' :

app.run(debug = True,port=5000)

**Source code(Application Building):**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

data=pd.read\_excel(r"Crude Oil Prices Daily.xlsx")

data.head()

data.describe()

data.isnull().sum()

data.dropna(axis=0,inplace=True)

data.isnull().sum()

data\_oil=data.reset\_index()['Closing Value']

data\_oil

from sklearn.preprocessing import MinMaxScaler

scaler=MinMaxScaler(feature\_range=(0,1))

data\_oil=scaler.fit\_transform(np.array(data\_oil).reshape(-1,1))

plt.plot(data\_oil)

training\_size=int(len(data\_oil)\*0.65)

test\_size=len(data\_oil)-training\_size

train\_data,test\_data=data\_oil[0:training\_size,:],data\_oil[training\_size:len(data\_oil),::]

training\_size,test\_size

train\_data.shape

#convert an array of values into a dataset matrix

def create\_dataset(dataset, time\_step=1):

dataX, dataY = [], []

for i in range(len(dataset)-time\_step-1):

a = dataset[i:(i+time\_step), 0]

dataX.append(a)

dataY.append(dataset[i + time\_step, 0])

return np.array(dataX), np.array(dataY)

time\_step = 10

x\_train, y\_train = create\_dataset(train\_data, time\_step)

x\_test, ytest = create\_dataset(test\_data, time\_step)

print(x\_train.shape), print(y\_train.shape)

print(x\_test.shape), print(ytest.shape)

x\_train

x\_train = x\_train.reshape(x\_train.shape[0],x\_train.shape[1], 1)

x\_test = x\_test.reshape(x\_test.shape[0],x\_test.shape[1] , 1)

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense

from tensorflow.keras.layers import LSTM

model.add(LSTM(50, return\_sequences=True, input\_shape=(10,1))) model.add(LSTM(50, return\_sequences=True)) model.add(LSTM(50))

**Output:**

