

Beginner Level Task...

Task-2 Stock Market Prediction And Forecasting Using Stacked LSTM :

Dataset: <https://raw.githubusercontent.com/mwilderick/stockprice/master/NSE-TATAGLOBAL.csv>

Name: Neha Tripathi

1.Importing Required Libraries and Packages

```
import numpy as np
import pandas as pd
import tensorflow as tf
import datetime
import math
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import MinMaxScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, LSTM
from sklearn.metrics import mean_squared_error
print("All required packages included successfully!")

All required packages included successfully!
```

2.Importing the Dataset

```
dataset_link='https://raw.githubusercontent.com/mwilderick/stockprice/master/NSE-TATAGLOBAL.csv'
```

```
df= pd.read_csv(dataset_link, parse_dates=True,)
df.reset_index()
df.head(5)
```

	Date	Open	High	Low	Last	Close	Total Trade Quantity	Turnover (Lacs)
0	2018-09-28	234.05	235.95	230.20	233.50	233.75	3069914	7162.35
1	2018-09-27	234.55	236.80	231.10	233.80	233.25	5062899	11899.95
2	2018-09-26	240.00	240.00	232.50	235.00	234.25	2240909	5248.60
3	2018-09-25	233.30	236.75	232.00	236.25	236.10	2349368	5503.90
4	2018-09-24	233.55	239.20	230.75	234.00	233.30	3423509	7999.55
5	2018-09-21	235.00	237.00	227.95	233.75	234.60	5396319	12589.59
6	2018-09-19	235.95	237.20	233.45	234.60	234.90	1362058	3202.78
7	2018-09-18	237.90	239.25	233.50	235.50	235.05	2614794	6163.70
8	2018-09-17	233.15	238.00	230.25	236.40	236.60	3170894	7445.41
9	2018-09-14	223.45	236.70	223.30	234.00	233.95	6377909	14784.50

3. Performing EDA

```
df.sample(10)
```

```
df[['Date', 'Open', 'High', 'Low', 'Last', 'Close', 'Total Trade Quantity', 'Turnover (Lacs)']]
```

```
df.shape
```

```
(2835, 8)
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2835 entries, 0 to 2834
Data columns (total 8 columns):
 #   Column              Non-Null Count  Dtype
---  ---
 0   Date                2835 non-null   object
 1   Open                2835 non-null   float64
 2   High                2835 non-null   float64
 3   Low                 2835 non-null   float64
 4   Last                2835 non-null   float64
 5   Close               2835 non-null   float64
 6   Total Trade Quantity 2835 non-null   int64
 7   Turnover (Lacs)      2835 non-null   float64
dtypes: float64(6), int64(1), object(1)
memory usage: 127.3+ KB
```

```
df.isnull().sum()
```

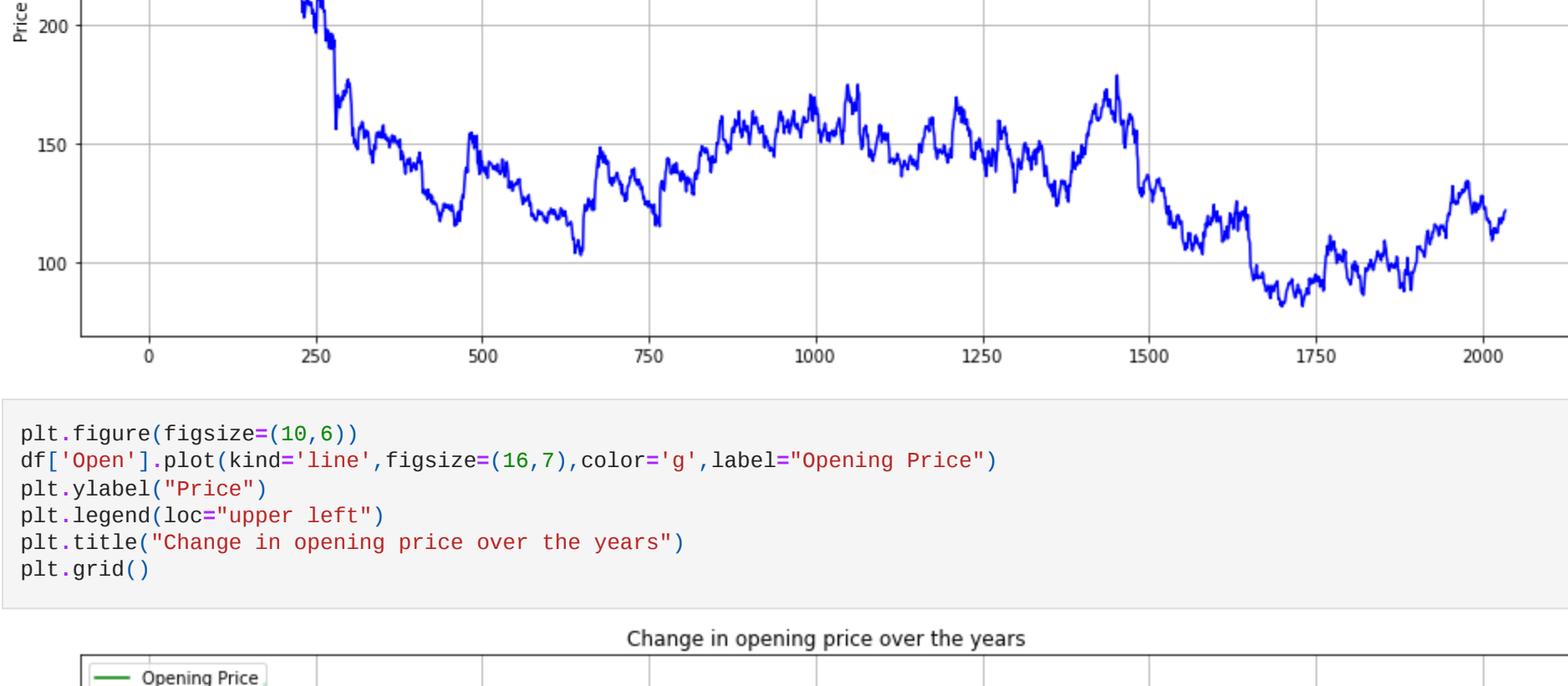
```
Date      0
Open      0
High      0
Low        0
Last       0
Close      0
Total Trade Quantity  0
Turnover (Lacs)      0
dtype: int64
```

```
df.describe()
```

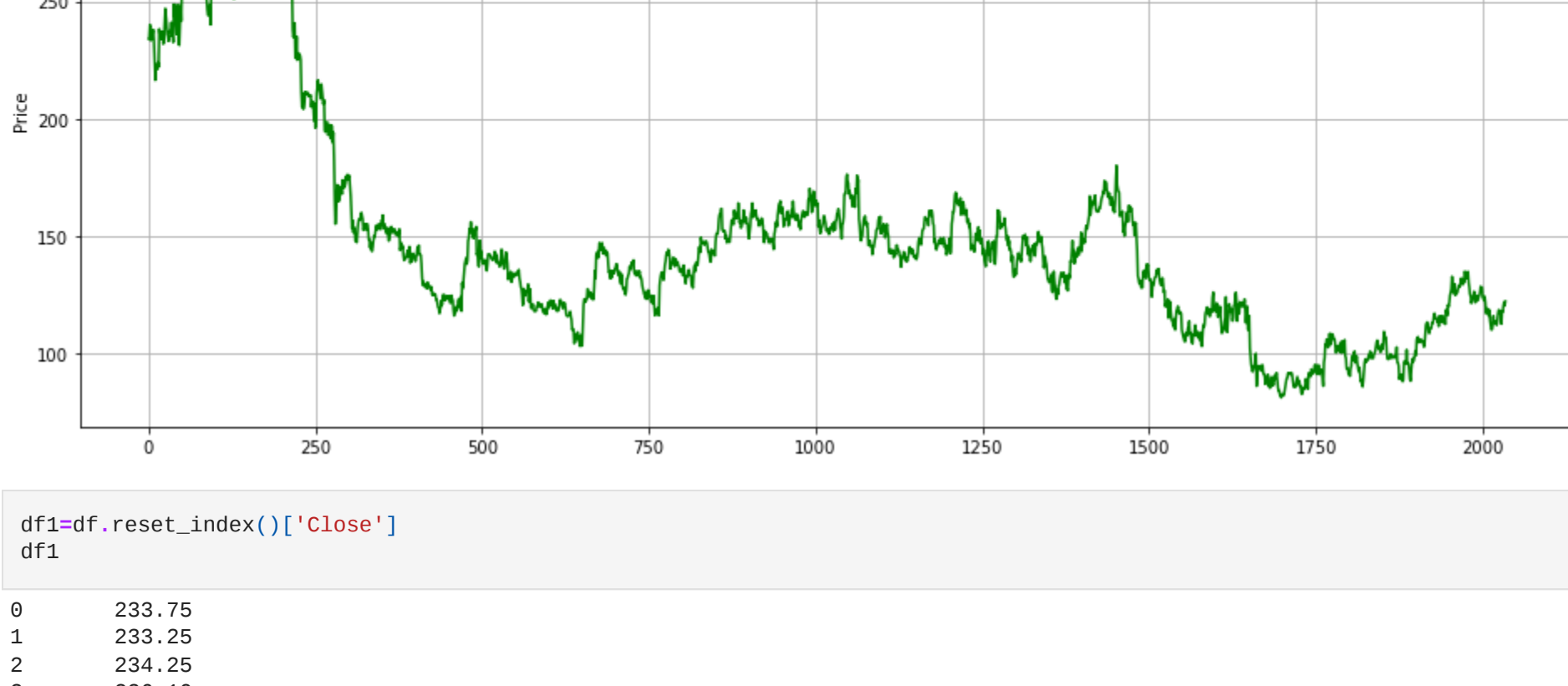
	Open	High	Low	Last	Close	Total Trade Quantity	Turnover (Lacs)
count	2035.000000	2035.000000	2035.000000	2035.000000	2035.000000	2.035000e+03	2035.000000
mean	149.713735	151.992828	147.293931	149.474251	149.45027	2.335681e+06	3899.980565
std	48.664509	49.413109	47.931958	48.732570	48.71204	2.091778e+06	4570.767877
min	81.000000	82.800000	80.000000	81.000000	80.95000	3.961000e+04	37.040000
25%	102.05000	122.100000	118.300000	120.075000	120.05000	1.146444e+06	1427.460000
50%	141.500000	143.400000	139.600000	141.100000	141.25000	1.783456e+06	2512.030000
75%	157.175000	159.400000	155.150000	156.925000	156.90000	2.813994e+06	4539.010000
max	327.700000	328.750000	321.650000	325.950000	325.75000	2.919102e+07	95795.080000

4. Parametric Visualization

```
plt.figure(figsize=(10,6))
df['Close'].plot(kind='line',figsize=(16,7),color='b',label="Closing Price")
plt.xlabel("Price")
plt.legend(locs="upper right")
plt.title("Change in closing price over the years")
plt.grid()
```



```
plt.figure(figsize=(10,6))
df['Open'].plot(kind='line',figsize=(16,7),color='g',label="Opening Price")
plt.xlabel("Price")
plt.legend(locs="upper left")
plt.title("Change in opening price over the years")
plt.grid()
```



```
df1=df.reset_index()['Close']
df1
```

```
0    233.75
1    233.25
2    234.25
3    236.10
4    233.30
...
2830   118.65
2831   117.60
2832   120.65
2833   120.90
2834   121.55
Name: Close, Length: 2835, dtype: float64
```

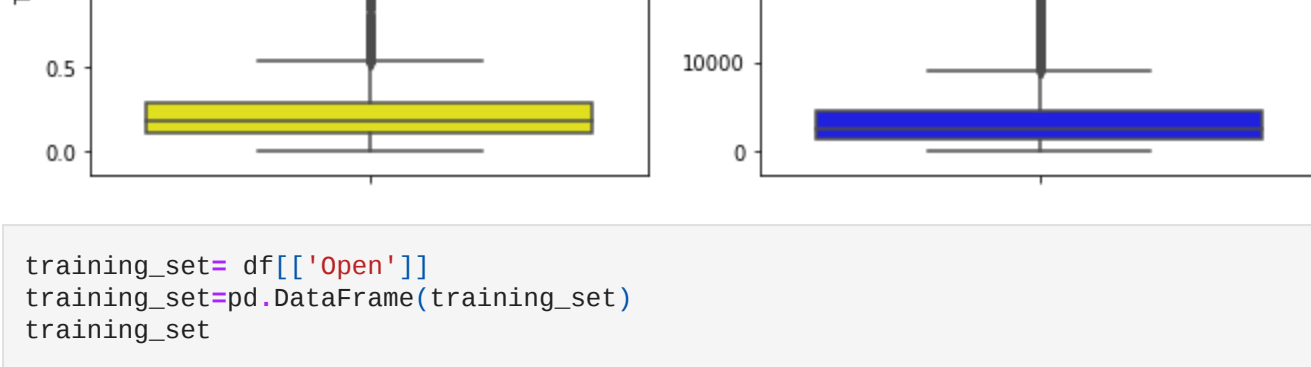
```
plt.figure(figsize=(12,6))
sns.heatmap(df.corr(),annot=True,cmap='BuPu')
```

```
<AxesSubplot:~>
```



```
plt.figure(figsize=(11,5))
plt.subplot(1,2,1)
sns.boxplot(data=df,y='Total Trade Quantity',color='yellow')
plt.subplot(1,2,2)
sns.boxplot(data=df,y='Turnover (Lacs)',color='blue')
```

```
<AxesSubplot:~>
```



```
training_set= df[['Open']]
training_set=pd.DataFrame(training_set)
training_set
```

```
Open
```

```
0    234.05
```

```
1    234.55
```

```
2    240.00
```

```
3    233.30
```

```
4    233.55
```

```
...
```

```
2830   117.60
```

```
2831   120.10
```

```
2832   121.80
```

```
2833   120.90
```

```
2834   122.10
```

```
2035 rows x 1 columns
```

5.Splitting and Transforming the Dataset

```
scaler=MinMaxScaler(feature_range=(0,1))
training_set=scaler.fit_transform(np.array(df1).reshape(-1,1))
```

```
train_size= int(len(training_set_scaler)*0.65)
test_size=int(len(training_set_scaler)-train_size)
train_data1,test_data1=training_set_scaler[0:train_size,1],training_set_scaler[train_size:len(df1),:1]
```

```
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=100
x_train,y_train=create_dataset(train_data1, time_step)
x_test,y_test= create_dataset(test_data1, time_step)
```

```
print(x_train.shape,y_train.shape)
```

```
(1221, 100) (1221, 1)
```

```
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)
```

6.Building the Model

```
model = Sequential()
model.add(LSTM(50, return_sequences=True, input_shape=(100,1)))
model.add(LSTM(50, return_sequences=True, input_shape=(100,1)))
model.add(LSTM(50))
model.add(Dense(1))
model.compile(loss='mean_squared_error', optimizer='adam', metrics='acc')
```

```
model.summary()
```

```
Model: "sequential_1"
Layer (type)                 Output Shape         Param #
-----
lstm_3 (LSTM)                 (None, 100, 50)      28400
lstm_4 (LSTM)                 (None, 100, 50)      28400
lstm_5 (LSTM)                 (None, 50)           28200
dense_1 (Dense)               (None, 1)             51
-----
Total params: 56,851
Trainable params: 56,851
Non-trainable params: 0
```

```
model.fit(x_train,y_train, validation_data = (x_test,y_test), epochs = 75, batch_size = 64, verbose = 1)
```

```
Epoch 1/75
28/20 [=====] - 10s 202ms/step - loss: 0.0260 - acc: 8.1900e-04 - val_loss: 0.0063 - val_acc: 0.0016
Epoch 2/75
28/20 [=====] - 3s 138ms/step - loss: 0.0033 - acc: 8.1900e-04 - val_loss: 0.0014 - val_acc: 0.0016
Epoch 3/75
28/20 [=====] - 3s 145ms/step - loss: 0.0018 - acc: 8.1900e-04 - val_loss: 0.0012 - val_acc: 0.0016
Epoch 4/75
28/20 [=====] - 3s 147ms/step - loss: 0.0015 - acc: 8.1900e-04 - val_loss: 0.0011 - val_acc: 0.0016
Epoch 5/75
28/20 [=====] - 3s 143ms/step - loss: 0.0016 - acc: 8.1900e-04 - val_loss: 0.0011 - val_acc: 0.0016
Epoch 6/75
28/20 [=====] - 3s 138ms/step - loss: 0.0016 - acc: 8.1900e-04 - val_loss: 0.0012 - val_acc: 0.0016
Epoch 7/75
28/20 [=====] - 3s 152ms/step - loss: 0.0013 - acc: 8.1900e-04 - val_loss: 0.9739e-04 - val_acc: 0.0016
Epoch 8/75
28/20 [=====] - 3s 154ms/step - loss: 0.0013 - acc: 8.1900e-04 - val_loss: 8.2228e-04 - val_acc: 0.0016
Epoch 9/75
28/20 [=====] - 3s 156ms/step - loss: 0.0014 - acc: 8.1900e-04 - val_loss: 8.8522e-04 - val_acc: 0.0016
Epoch 10/75
28/20 [=====] - 3s 142ms/step - loss: 0.0012 - acc: 8.1900e-04 - val_loss: 8.2590e-04 - val_acc: 0.0016
Epoch 11/75
28/20 [=====] - 3s 153ms/step - loss: 0.0011 - acc: 8.1900e-04 - val_loss: 8.1912e-04 - val_acc: 0.0016
Epoch 12/75
28/20 [=====] - 3s 156ms/step - loss: 0.0011 - acc: 8.1900e-04 - val_loss: 6.7901e-04 - val_acc: 0.0016
Epoch 13/75
28/20 [=====] - 3s 144ms/step - loss: 0.0011 - acc: 8.1900e-04 - val_loss: 8.1020e-04 - val_acc: 0.0016
Epoch 14/75
28/20 [=====] - 3s 147ms/step - loss: 8.8078e-04 - acc: 8.1900e-04 - val_loss: 7.5177e-04 - val_acc: 0.0016
Epoch 15/75
28/20 [=====] - 3s 150ms/step - loss: 8.5999e-04 - acc: 8.1900e-04 - val_loss: 8.9449e-04 - val_acc: 0.0016
Epoch 16/75
28/20 [=====] - 3s 147ms/step - loss: 9.5655e-04 - acc: 8.1900e-04 - val_loss: 7.4808e-04 - val_acc: 0.0016
Epoch 17/75
28/20 [=====] - 3s 149ms/step - loss: 8.9458e-04 - acc: 8.1900e-04 - val_loss: 6.1651e-04 - val_acc: 0.0016
Epoch 18/75
28/20 [=====] - 3s 144ms/step - loss: 8.8339e-04 - acc: 8.1900e-04 - val_loss: 6.1371e-04 - val_acc: 0.0016
Epoch 19/75
28/20 [=====] - 3s 157ms/step - loss: 8.8328e-04 - acc: 8.1900e-04 - val_loss: 7.1401e-04 - val_acc: 0.0016
Epoch 20/75
28/20 [=====] - 3s 168ms/step - loss: 7.9292e-04 - acc: 8.1900e-04 - val_loss: 6.7901e-04 - val_acc: 0.0016
Epoch 21/75
28/20 [=====] - 3s 161ms/step - loss: 7.3923e-04 - acc: 8.1900e-04 - val_loss: 7.9189e-04 - val_acc: 0.0016
Epoch 22/75
28/20 [=====] - 3s 164ms/step - loss: 7.4088e-04 - acc: 8.1900e-04 - val_loss: 6.7126e-04 - val_acc: 0.0016
Epoch 23/75
28/20 [=====] - 3s 156ms/step - loss: 7.7677e-04 - acc: 8.1900e-04 - val_loss: 7.6438e-04 - val_acc: 0.0016
Epoch 24/75
28/20 [=====] - 3s 159ms/step - loss: 6.6252e-04 - acc: 8.1900e-04 - val_loss: 6.0745e-04 - val_acc: 0.0016
Epoch 25/75
28/20 [=====] - 3s 154ms/step - loss: 7.7655e-04 - acc: 8.1900e-04 - val_loss: 6.5055e-04 - val_acc: 0.0016
Epoch 26/75
28/20 [=====] - 3s 152ms/step - loss: 8.8357e-04 - acc: 8.1900e-04 - val_loss: 6.8460e-04 - val_acc: 0.0016
Epoch 27/75
28/20 [=====] - 3s 150ms/step - loss: 6.6036e-04 - acc: 8.1900e-04 - val_loss: 6.5111e-04 - val_acc: 0.0016
Epoch 28/75
28/20 [=====] - 3s 160ms/step - loss: 6.3912e-04 - acc: 8.1900e-04 - val_loss: 6.9801e-04 - val_acc: 0.0016
Epoch 29/75
28/20 [=====] - 3s 157ms/step - loss: 6.1519e-04 - acc: 8.1900e-04 - val_loss: 6.2612e-04 - val_acc: 0.0016
Epoch 30/75
28/20 [=====] - 3s 154ms/step - loss: 5.8753e-04 - acc: 8.1900e-04 - val_loss: 6.2238e-04 - val_acc: 0.0016
Epoch 31/75
28/20 [=====] - 3s 149ms/step - loss: 5.9932e-04 - acc: 8.1900e-04 - val_loss: 4.6276e-04 - val_acc: 0.0016
Epoch 32/75
28/20 [=====] - 3s 148ms/step - loss: 5.5727e-04 - acc: 8.1900e-04 - val_loss: 5.2977e-04 - val_acc: 0.0016
Epoch 33/75
28/20 [=====] - 4s 188ms/step - loss: 5.5910e-04 - acc: 8.1900e-04 - val_loss: 5.3496e-04 - val_acc: 0.0016
Epoch 34/75
28/20 [=====] - 3s 172ms/step - loss: 5.1168e-04 - acc: 8.1900e-04 - val_loss: 6.9071e-04 - val_acc: 0.0016
Epoch 35/75
28/20 [=====] - 3s 146ms/step - loss: 4.9454e-04 - acc: 8.1900e-04 - val_loss: 5.2911e-04 - val_acc: 0.0016
Epoch 36/75
28/20 [=====] - 3s 156ms/step - loss: 4.9212e-04 - acc: 8.1900e-04 - val_loss: 5.6185e-04 - val_acc: 0.0016
Epoch 37/75
28/20 [=====] - 3s 148ms/step - loss: 5.2935e-04 - acc: 8.1900e-04 - val_loss: 5.7607e-04 - val_acc: 0.0016
Epoch 38/75
28/20 [=====] - 3s 157ms/step - loss: 4.6468e-04 - acc: 8.1900e-04 - val_loss: 4.8099e-04 - val_acc: 0.0016
Epoch 39/75
28/20 [=====] - 3s 148ms/step - loss: 4.4259e-04 - acc: 8.1900e-04 - val_loss: 5.3878e-04 - val_acc: 0.0016
Epoch 40/75
28/20 [=====] - 3s 144ms/step - loss: 4.5824e-04 - acc: 8.1900e-04 - val_loss: 4.0745e-04 - val_acc: 0.0016
Epoch 41/75
28/20 [=====] - 3s 144ms/step - loss: 4.5254e-04 - acc: 8.1900e-04 - val_loss: 5.3405e-04 - val_acc: 0.0016
Epoch 42/75
28/20 [=====] - 3s 153ms/step - loss: 3.9192e-04 - acc: 8.1900e-04 - val_loss: 6.1533e-04 - val_acc: 0.0016
Epoch 43/75
28/20 [=====] - 3s 146ms/step - loss: 4.7294e-04 - acc: 8.1900e-04 - val_loss: 4.0906e-04 - val_acc: 0.0016
Epoch 44/75
28/20 [=====] - 3s 148ms/step - loss: 4.6050e-04 - acc: 8.1900e-04 - val_loss: 3.9033e-04 - val_acc: 0.0016
Epoch 45/75
28/20 [=====] - 3s 143ms/step - loss: 3.5533e-04 - acc: 8.1900e-04 - val_loss: 3.4979e-04 - val_acc: 0.0016
Epoch 46/75
28/20 [=====] - 3s 149ms/step - loss: 3.6811e-04 - acc: 8.1900e-04 - val_loss: 4.5556e-04 - val_acc: 0.0016
Epoch 47/75
28/20 [=====] - 3s 146ms/step - loss: 3.7284e-04 - acc: 8.1900e-04 - val_loss: 3.7616e-04 - val_acc: 0.0016
Epoch 48/75
28/20 [=====] - 3s 144ms/step - loss: 4.2066e-04 - acc: 8.1900e-04 - val_loss: 3.4603e-04 - val_acc: 0.0016
Epoch 49/75
28/20 [=====] - 3s 146ms/step - loss: 3.9496e-04 - acc: 8.1900e-04 - val_loss: 3.2456e-04 - val_acc: 0.0016
Epoch 50/75
28/20 [=====] - 3s 141ms/step - loss: 3.2932e-04 - acc: 8.1900e-04 - val_loss: 3.1912e-04 - val_acc: 0.0016
Epoch 51/75
28/20 [=====] - 3s 145ms/step - loss: 3.1815e-04 - acc: 8.1900e-04 - val_loss: 3.2586e-04 - val_acc: 0.0016
Epoch 52/75
28/20 [=====] - 3s 149ms/step - loss: 3.1228e-04 - acc: 8.1900e-04 - val_loss: 3.1151e-04 - val_acc: 0.0016
Epoch 53/75
28/20 [=====] - 3s 153ms/step - loss: 3.4316e-04 - acc: 8.1900e-04 - val_loss: 3.1151e-04 - val_acc: 0.0016
Epoch 54/75
28/20 [=====] - 3s 147ms/step - loss: 3.0306e-04 - acc: 8.1900e-04 - val_loss: 2.6610e-04 - val_acc: 0.0016
Epoch 55/75
28/20 [=====] - 3s 143ms/step - loss: 3.2810e-04 - acc: 8.1900e-04 - val_loss: 2.5491e-04 - val_acc: 0.0016
Epoch 56/75
28/20 [=====] - 3s 146ms/step - loss: 3.6841e-04 - acc: 8.1900e-04 - val_loss: 3.5210e-04 - val_acc: 0.0016
Epoch 57/75
28/20 [=====] - 3s 146ms/step - loss: 3.1213e-04 - acc: 8.1900e-04 - val_loss: 3.9873e-04 - val_acc: 0.0016
Epoch 58/75
28/20 [=====] - 3s 150ms/step - loss: 3.0978e-04 - acc: 8.1900e-04 - val_loss: 3.9033e-04 - val_acc: 0.0016
Epoch 59/75
28/20 [=====] - 3s 147ms/step - loss: 2.8359e-04 - acc: 8.1900e-04 - val_loss: 3.6507e-04 - val_acc: 0.0016
Epoch 60/75
28/20 [=====] - 3s 142ms/step - loss: 2.7195e-04 - acc: 8.1900e-04 - val_loss: 2.3409e-04 - val_acc: 0.0016
Epoch 61/75
28/20 [=====] - 3s 145ms/step - loss: 2.6361e-04 - acc: 8.1900e-04 - val_loss: 4.4546e-04 - val_acc: 0.0016
Epoch 62/75
28/20 [=====] - 3s 145ms/step - loss: 4.4556e-04 - acc: 8.1900e-04 - val_loss: 2.8315e-04 - val_acc: 0.0016
Epoch 63/75
28/20 [=====] - 3s 146ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 64/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 65/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 66/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 67/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 68/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 69/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 70/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 71/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 72/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 73/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 74/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 75/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 76/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 77/75
28/20 [=====] - 3s 145ms/step - loss: 2.7079e-04 - acc: 8.1900e-04 - val_loss: 2.8212e-04 - val_acc: 0.0016
Epoch 78/75
28/20 [=====] - 3s 145ms/step - loss: 2.707
```