**Assignment -4**

|  |  |
| --- | --- |
| **Assignment Date** | 05-11-22 |
| **Student Name** | G.samson |
| **Student Roll Number** | 814619106301 |
| **Maximum Marks** | 2 marks |

***SMS SPAM Classification***

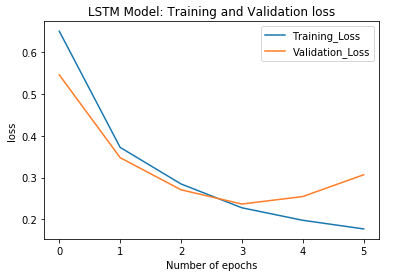
**Importing of Libraries**

import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns  
import scipy as sp  
from google.colab import drive  
from sklearn import feature\_extraction, model\_selection, naive\_bayes, metrics, svm  
from sklearn.ensemble import RandomForestClassifier  
from sklearn.model\_selection import train\_test\_split  
from sklearn.metrics import precision\_recall\_fscore\_support as score  
%matplotlib inline  
drive.mount('/content/drive')

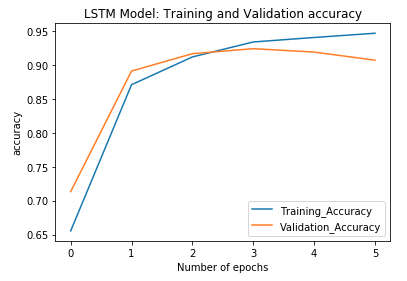
**ADD LAYERS**

# Create a dataframe  
metrics = pd.DataFrame(history.history)# Rename column  
metrics.rename(columns = {'loss': 'Training\_Loss', 'accuracy': 'Training\_Accuracy',  
 'val\_loss': 'Validation\_Loss', 'val\_accuracy': 'Validation\_Accuracy'}, inplace = True)  
def plot\_graphs1(var1, var2, string):  
 metrics[[var1, var2]].plot()  
 plt.title('LSTM Model: Training and Validation ' + string)  
 plt.xlabel ('Number of epochs')  
 plt.ylabel(string)  
 plt.legend([var1, var2])plot\_graphs1('Training\_Loss', 'Validation\_Loss', 'loss')  
plot\_graphs1('Training\_Accuracy', 'Validation\_Accuracy', 'accuracy')

**COMPILATION MODEL**



**Tested Model**



Keeping track of accuracy over epochs: LSTM