In [1]:
```
from __future__ import print_function
#from sklearn.cross_validation import train_test_split
import pandas as pd
import numpy as np
np.random.seed(1337)  # for reproducibility
from keras.preprocessing import sequence
from keras.utils import np_utils
from keras.models import Sequential
from keras.layers import Dense, Dropout, Activation, Embedding
from keras.layers import LSTM, SimpleRNN, GRU
from keras.datasets import imdb
from keras.utils.np_utils import to_categorical
from sklearn.metrics import (precision_score, recall_score,f1_score, accuracy_sc
from sklearn import metrics
from sklearn.preprocessing import Normalizer
import h5py
from keras import callbacks
from keras.callbacks import ModelCheckpoint, EarlyStopping, ReduceLROnPlateau, CS
from sklearn.preprocessing import LabelEncoder,MinMaxScaler
```

In [2]:
```
traindata = pd.read_csv(r'C:\Users\elroofey\OneDrive\Documents\KDDTrain.csv', hea
testdata = pd.read_csv(r'C:\Users\elroofey\OneDrive\Documents\KDDTest.csv', heade
```

In [3]:
```
traindata
```

Out[3]:

	0	1	2	3	4	5	6	7	8	9
0	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot
1	0	tcp	ftp_data	SF	491	0	0	0	0	0
2	0	udp	other	SF	146	0	0	0	0	0
3	0	tcp	private	S0	0	0	0	0	0	0
4	0	tcp	http	SF	232	8153	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...
1995	0	tcp	ftp_data	SF	20658	0	0	0	0	0
1996	0	udp	domain_u	SF	44	132	0	0	0	0
1997	37749	tcp	private	RSTR	1	0	0	0	0	0
1998	0	tcp	ftp_data	SF	3676	0	0	0	0	0
1999	0	tcp	http	SF	301	10063	0	0	0	0

2000 rows × 42 columns

In [4]:
```
testdata
```

Out[4]:

	0	1	2	3	4	5	6	7	8	9
0	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot
1	0	tcp	private	REJ	0	0	0	0	0	0
2	0	tcp	private	REJ	0	0	0	0	0	0
3	2	tcp	ftp_data	SF	12983	0	0	0	0	0
4	0	icmp	eco_i	SF	20	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...
495	0	tcp	discard	RSTO	0	0	0	0	0	0
496	0	udp	other	SF	1	1	0	0	0	0
497	0	icmp	eco_i	SF	20	0	0	0	0	0
498	0	udp	domain_u	SF	43	83	0	0	0	0
499	0	tcp	courier	S0	0	0	0	0	0	0

500 rows × 42 columns

In [5]:
```
#We will use LabelEncoder for our Data Encoding
for i in traindata.select_dtypes('object').columns:
  le = LabelEncoder().fit(traindata[i])
  traindata[i] = le.transform(traindata[i])
```

In [6]:
```
traindata=traindata.astype(float)
```

In [7]:
```
#We will use LabelEncoder for our Data Encoding
for i in testdata.select_dtypes('object').columns:
  le = LabelEncoder().fit(testdata[i])
  testdata[i] = le.transform(testdata[i])
```

In [8]:
```
testdata=testdata.astype(float)
```

In [9]:
```
X = traindata.iloc[:,1:42]
Y = traindata.iloc[:,0]
C = testdata.iloc[:,0]
T = testdata.iloc[:,1:42]
```

In [10]:
```
scaler = Normalizer().fit(X)
trainX = scaler.transform(X)
# summarize transformed data
np.set_printoptions(precision=3)
print(trainX[0:5,:])
```

```
[[0.001 0.051 0.01  0.449 0.697 0.001 0.003 0.001 0.013 0.003 0.002 0.008
  0.002 0.002 0.005 0.003 0.001 0.002 0.001 0.001 0.002 0.325 0.177 0.035
  0.021 0.033 0.018 0.064 0.045 0.049 0.243 0.245 0.107 0.087 0.095 0.045
  0.044 0.025 0.085 0.048 0.001]
 [0.005 0.047 0.018 0.855 0.    0.    0.    0.    0.    0.    0.    0.
  0.    0.    0.    0.    0.    0.    0.    0.    0.254 0.142 0.
  0.    0.    0.    0.15  0.    0.    0.134 0.37  0.045 0.008 0.045 0.
  0.    0.    0.013 0.    0.005]
 [0.015 0.196 0.036 0.371 0.    0.    0.    0.    0.    0.    0.    0.
  0.    0.    0.    0.    0.    0.    0.    0.    0.175 0.    0.
  0.    0.    0.    0.041 0.072 0.    0.748 0.    0.    0.273 0.387 0.
  0.    0.    0.    0.    0.01 ]
 [0.008 0.163 0.015 0.    0.    0.    0.    0.    0.    0.    0.    0.
  0.    0.    0.    0.    0.    0.    0.    0.    0.103 0.54  0.118
  0.068 0.    0.    0.019 0.027 0.    0.551 0.563 0.038 0.019 0.    0.
  0.148 0.084 0.    0.    0.   ]
 [0.003 0.03  0.01  0.235 0.819 0.    0.    0.    0.    0.    0.001 0.
  0.    0.    0.    0.    0.    0.    0.    0.    0.351 0.189 0.023
  0.014 0.    0.    0.08  0.    0.    0.213 0.207 0.135 0.    0.004 0.006
  0.004 0.001 0.    0.001 0.003]]
```

In [11]:
```
scaler = Normalizer().fit(T)
testT = scaler.transform(T)
# summarize transformed data
np.set_printoptions(precision=3)
print(testT[0:5,:])
```

```
[[0.003 0.068 0.024 0.466 0.521 0.003 0.008 0.003 0.016 0.005 0.005 0.008
  0.005 0.005 0.008 0.008 0.005 0.008 0.003 0.003 0.005 0.411 0.186 0.045
  0.031 0.05  0.039 0.071 0.055 0.097 0.293 0.301 0.186 0.11  0.105 0.06
  0.079 0.068 0.149 0.079 0.003]
 [0.016 0.202 0.    0.    0.    0.    0.    0.    0.    0.    0.    0.
  0.    0.    0.    0.    0.    0.    0.    0.    0.631 0.008 0.
  0.    0.146 0.113 0.032 0.032 0.    0.51  0.008 0.032 0.049 0.    0.
  0.    0.    0.453 0.235 0.   ]
 [0.02  0.253 0.    0.    0.    0.    0.    0.    0.    0.    0.    0.
  0.    0.    0.    0.    0.    0.    0.    0.    0.253 0.    0.
  0.    0.182 0.142 0.01  0.04  0.    0.637 0.    0.    0.061 0.    0.
  0.    0.    0.566 0.293 0.   ]
 [0.016 0.088 0.056 0.136 0.    0.    0.    0.    0.    0.    0.    0.
  0.    0.    0.    0.    0.    0.    0.    0.    0.    0.    0.
  0.    0.    0.    0.207 0.    0.    0.128 0.869 0.343 0.032 0.191 0.016
  0.    0.    0.    0.    0.016]
 [0.    0.042 0.042 0.225 0.    0.    0.    0.    0.    0.    0.    0.
  0.    0.    0.    0.    0.    0.    0.    0.    0.    0.388 0.
  0.    0.    0.    0.158 0.    0.218 0.401 0.564 0.425 0.    0.237 0.085
  0.    0.    0.    0.    0.   ]]
```

In [12]:
```
y_train = np.array(Y)
y_test = np.array(C)
```

In [13]:
```
# reshape input to be [samples, time steps, features]
X_train = np.reshape(trainX, (trainX.shape[0], 1, trainX.shape[1]))
X_test = np.reshape(testT, (testT.shape[0], 1, testT.shape[1]))
```

In [14]:
```
print(X_train.shape)
```

```
(2000, 1, 41)
```

In [15]:
```
print(X_test.shape)
```

```
(500, 1, 41)
```

In [16]:
```
batch_size = 32
```

In [17]:
```
# 1. define the network
model = Sequential()
model.add(LSTM(4,input_dim=41))  # try using a GRU instead, for fun
model.add(Dropout(0.1))
model.add(Dense(1))
model.add(Activation('sigmoid'))
print(model.get_config())
```

```
{'name': 'sequential', 'layers': [{'class_name': 'InputLayer', 'config': {'batc
h_input_shape': (None, None, 41), 'dtype': 'float32', 'sparse': False, 'ragge
d': False, 'name': 'lstm_input'}}, {'class_name': 'LSTM', 'config': {'name': 'l
stm', 'trainable': True, 'batch_input_shape': (None, None, 41), 'dtype': 'float
32', 'return_sequences': False, 'return_state': False, 'go_backwards': False,
'stateful': False, 'unroll': False, 'time_major': False, 'units': 4, 'activatio
n': 'tanh', 'recurrent_activation': 'sigmoid', 'use_bias': True, 'kernel_initia
lizer': {'class_name': 'GlorotUniform', 'config': {'seed': None}}, 'recurrent_i
nitializer': {'class_name': 'Orthogonal', 'config': {'gain': 1.0, 'seed': Non
e}}, 'bias_initializer': {'class_name': 'Zeros', 'config': {}}, 'unit_forget_bi
as': True, 'kernel_regularizer': None, 'recurrent_regularizer': None, 'bias_reg
ularizer': None, 'activity_regularizer': None, 'kernel_constraint': None, 'recu
rrent_constraint': None, 'bias_constraint': None, 'dropout': 0.0, 'recurrent_dr
opout': 0.0, 'implementation': 2}}, {'class_name': 'Dropout', 'config': {'nam
e': 'dropout', 'trainable': True, 'dtype': 'float32', 'rate': 0.1, 'noise_shap
e': None, 'seed': None}}, {'class_name': 'Dense', 'config': {'name': 'dense',
'trainable': True, 'dtype': 'float32', 'units': 1, 'activation': 'linear', 'use
_bias': True, 'kernel_initializer': {'class_name': 'GlorotUniform', 'config':
{'seed': None}}, 'bias_initializer': {'class_name': 'Zeros', 'config': {}}, 'ke
rnel_regularizer': None, 'bias_regularizer': None, 'activity_regularizer': Non
e, 'kernel_constraint': None, 'bias_constraint': None}}, {'class_name': 'Activa
tion', 'config': {'name': 'activation', 'trainable': True, 'dtype': 'float32',
'activation': 'sigmoid'}}]}
```

In [18]:
```
# try using different optimizers and different optimizer configs
model.compile(loss='binary_crossentropy',optimizer='adam',metrics=['accuracy'])
checkpointer = callbacks.ModelCheckpoint(filepath="results/checkpoint-{epoch:02d}
csv_logger = CSVLogger('results/training_set_iranalysis.csv',separator=',', apper
model.fit(X_train, y_train, batch_size=batch_size, epochs=10, validation_data=(X_
model.save("results/lstm1layer_model.hdf5")

loss, accuracy = model.evaluate(X_test, y_test)
print("\nLoss: %.2f, Accuracy: %.2f%%" % (loss, accuracy*4000))
y_pred = model.predict(X_test)
classes_x=np.argmax(y_pred, axis=1)
np.savetxt('lstm1predicted.txt', y_pred, fmt='%01d')
```

```
Epoch 1/10
63/63 [==============================] - 10s 41ms/step - loss: 0.4661 - accur
acy: 0.0355 - val_loss: 0.3914 - val_accuracy: 0.0220

Epoch 00001: val_accuracy improved from -inf to 0.02200, saving model to resu
lts\checkpoint-01.hdf5
Epoch 2/10
63/63 [==============================] - 1s 11ms/step - loss: 0.1485 - accura
cy: 0.0210 - val_loss: 0.1436 - val_accuracy: 0.0220

Epoch 00002: val_accuracy did not improve from 0.02200
Epoch 3/10
63/63 [==============================] - 1s 12ms/step - loss: -0.2031 - accur
acy: 0.0205 - val_loss: -0.1546 - val_accuracy: 0.0220

Epoch 00003: val_accuracy did not improve from 0.02200
Epoch 4/10
63/63 [==============================] - 1s 12ms/step - loss: -0.6149 - accur
acy: 0.0205 - val_loss: -0.4780 - val_accuracy: 0.0220

Epoch 00004: val_accuracy did not improve from 0.02200
Epoch 5/10
63/63 [==============================] - 1s 11ms/step - loss: -0.9897 - accur
acy: 0.0205 - val_loss: -0.7989 - val_accuracy: 0.0220

Epoch 00005: val_accuracy did not improve from 0.02200
Epoch 6/10
63/63 [==============================] - 1s 9ms/step - loss: -1.5237 - accura
cy: 0.0205 - val_loss: -1.1529 - val_accuracy: 0.0220

Epoch 00006: val_accuracy did not improve from 0.02200
Epoch 7/10
63/63 [==============================] - 1s 12ms/step - loss: -2.0467 - accur
acy: 0.0205 - val_loss: -1.5156 - val_accuracy: 0.0220

Epoch 00007: val_accuracy did not improve from 0.02200
Epoch 8/10
63/63 [==============================] - 1s 13ms/step - loss: -2.3999 - accur
acy: 0.0205 - val_loss: -1.8598 - val_accuracy: 0.0220

Epoch 00008: val_accuracy did not improve from 0.02200
Epoch 9/10
```

```
63/63 [==============================] - 1s 11ms/step - loss: -3.0352 - accur
acy: 0.0205 - val_loss: -2.1708 - val_accuracy: 0.0220

Epoch 00009: val_accuracy did not improve from 0.02200
Epoch 10/10
63/63 [==============================] - 1s 14ms/step - loss: -3.3730 - accur
acy: 0.0205 - val_loss: -2.4759 - val_accuracy: 0.0220

Epoch 00010: val_accuracy did not improve from 0.02200
16/16 [==============================] - 0s 5ms/step - loss: -2.4759 - accura
cy: 0.0220

Loss: -2.48, Accuracy: 88.00%
```

In [19]:
```
# Import library for synthetic data generation using CTGAN
from ctgan import CTGANSynthesizer
```

In [20]:
```
ctgan = CTGANSynthesizer(epochs=10)
```

In [21]:
```
# Train CTGAN on KDDTrain data
ctgan.fit(traindata)
```

```
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (4) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:277: ConvergenceWarning: Initialization 1 did not converge. Try d
ifferent init parameters, or increase max_iter, tol or check for degenerate d
ata.
  warnings.warn(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:277: ConvergenceWarning: Initialization 1 did not converge. Try d
ifferent init parameters, or increase max_iter, tol or check for degenerate d
ata.
  warnings.warn(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:277: ConvergenceWarning: Initialization 1 did not converge. Try d
ifferent init parameters, or increase max_iter, tol or check for degenerate d
ata.
  warnings.warn(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (2) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (4) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (2) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (4) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (3) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (8) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (3) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (3) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (6) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
```

```
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (4) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (2) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (3) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (2) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (2) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (3) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:277: ConvergenceWarning: Initialization 1 did not converge. Try d
ifferent init parameters, or increase max_iter, tol or check for degenerate d
ata.
  warnings.warn(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:277: ConvergenceWarning: Initialization 1 did not converge. Try d
ifferent init parameters, or increase max_iter, tol or check for degenerate d
ata.
  warnings.warn(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:277: ConvergenceWarning: Initialization 1 did not converge. Try d
ifferent init parameters, or increase max_iter, tol or check for degenerate d
ata.
  warnings.warn(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:277: ConvergenceWarning: Initialization 1 did not converge. Try d
ifferent init parameters, or increase max_iter, tol or check for degenerate d
ata.
  warnings.warn(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:277: ConvergenceWarning: Initialization 1 did not converge. Try d
ifferent init parameters, or increase max_iter, tol or check for degenerate d
ata.
  warnings.warn(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:277: ConvergenceWarning: Initialization 1 did not converge. Try d
ifferent init parameters, or increase max_iter, tol or check for degenerate d
ata.
  warnings.warn(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:277: ConvergenceWarning: Initialization 1 did not converge. Try d
```

```
ifferent init parameters, or increase max_iter, tol or check for degenerate d
ata.
  warnings.warn(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:277: ConvergenceWarning: Initialization 1 did not converge. Try d
ifferent init parameters, or increase max_iter, tol or check for degenerate d
ata.
  warnings.warn(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (3) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
```

In [22]:
```
# generate synthetic copy
sample_train = ctgan.sample(2000)
```

In [24]:
```
# Train CTGAN on KDDTest data
ctgan.fit(testdata)
```

```
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (4) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:277: ConvergenceWarning: Initialization 1 did not converge. Try d
ifferent init parameters, or increase max_iter, tol or check for degenerate d
ata.
  warnings.warn(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:277: ConvergenceWarning: Initialization 1 did not converge. Try d
ifferent init parameters, or increase max_iter, tol or check for degenerate d
ata.
  warnings.warn(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (2) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (4) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (2) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (7) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (3) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (3) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (4) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (3) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (3) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (4) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
```

```
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (4) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (3) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (4) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (2) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (2) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (3) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
C:\Users\elroofey\AppData\Roaming\Python\Python38\site-packages\sklearn\mixtu
re\_base.py:143: ConvergenceWarning: Number of distinct clusters (3) found sm
aller than n_clusters (10). Possibly due to duplicate points in X.
  cluster.KMeans(
```

In [25]:
```
# generate synthetic copy
sample_test = ctgan.sample(500)
```

In [27]:
```
#We will use LabelEncoder for our Data Encoding
for i in sample_train.select_dtypes('object').columns:
  le = LabelEncoder().fit(sample_train[i])
  sample_train[i] = le.transform(sample_train[i])
```

In [28]:
```
sample_train=sample_train.astype(float)
```

In [29]:
```
#We will use LabelEncoder for our Data Encoding
for i in sample_test.select_dtypes('object').columns:
  le = LabelEncoder().fit(sample_test[i])
  sample_test[i] = le.transform(sampletest[i])
```

In [30]:
```
sample_test=sample_test.astype(float)
```

In [31]:
```
X = sample_train.iloc[:,1:42]
Y = sample_train.iloc[:,0]
C = sample_test.iloc[:,0]
T = sample_test.iloc[:,1:42]
```

In [32]:
```
scaler = Normalizer().fit(X)
trainX = scaler.transform(X)
# summarize transformed data
np.set_printoptions(precision=3)
print(trainX[0:5,:])
```

```
[[ 9.033e-03  9.596e-02  3.247e-02 -5.144e-02 -1.034e-01  4.406e-06
   2.087e-03 -3.395e-06 -6.626e-05 -1.075e-05  4.487e-03  5.644e-05
  -2.207e-05  7.646e-06  6.774e-05  2.295e-05  6.929e-06  1.471e-05
  -2.454e-06  2.597e-06 -2.460e-05  1.634e-02  1.968e-01 -6.479e-03
  -1.632e-03  3.433e-04  2.675e-04  9.692e-02 -2.935e-03  2.326e-01
   6.633e-01  6.377e-01  5.323e-03  2.824e-02  2.043e-02 -1.476e-03
  -3.079e-03  7.869e-04  1.604e-01 -8.530e-04  9.035e-03]
 [ 1.186e-02  4.247e-02  4.175e-02  2.693e-02  2.461e-02 -1.274e-05
   1.842e-06 -2.295e-05 -4.340e-05 -1.276e-05  5.541e-03  7.700e-05
  -7.134e-06  9.588e-06  5.348e-05 -8.623e-06  7.514e-06  7.286e-06
   6.600e-06 -2.743e-06  3.677e-05 -2.039e-02  2.734e-01 -6.889e-03
   1.121e-01  3.484e-03  1.457e-03  3.370e-01 -3.867e-03 -3.663e-03
   8.650e-01 -9.484e-02  6.922e-02 -1.039e-02 -6.086e-03 -1.772e-03
   1.152e-01  1.341e-01 -2.236e-04  1.675e-03  1.244e-02]
 [ 6.793e-03  1.741e-01  3.795e-02 -5.306e-03 -6.037e-02  2.133e-06
  -4.717e-05 -5.896e-06 -1.363e-04  2.002e-06 -8.034e-05  5.209e-05
  -4.556e-06 -4.036e-06  4.718e-05  9.563e-06  4.314e-06  2.370e-06
  -7.209e-06 -4.373e-06  1.834e-05  7.754e-01  1.308e-01 -3.466e-03
   5.255e-02 -5.038e-04  5.577e-02  1.562e-01 -8.509e-04  7.270e-04
   4.990e-01  1.191e-01  1.872e-01 -3.466e-03 -3.802e-03  3.866e-02
   1.265e-01  2.078e-04 -7.330e-03  6.999e-04  6.833e-03]
 [ 4.352e-03  5.050e-02  1.574e-02  6.793e-03  5.930e-01  3.569e-06
  -9.288e-06 -6.875e-06 -2.032e-05  8.803e-06  2.113e-03  2.272e-05
  -8.116e-08  5.864e-06  2.602e-05  1.208e-05  6.912e-06  4.880e-06
   8.101e-06  8.220e-08  5.717e-06  5.059e-01  3.689e-01 -2.659e-04
  -1.350e-03  7.656e-02  8.102e-04  1.283e-01 -5.963e-04  1.734e-03
   3.120e-01  2.915e-01  2.205e-01  1.115e-02 -2.221e-03 -8.660e-04
   2.214e-03  1.486e-03 -1.007e-03  9.659e-04  4.389e-03]
 [ 8.994e-03  9.559e-02  3.200e-02  3.141e-02 -9.444e-02  7.121e-06
  -3.657e-05 -1.037e-05  8.267e-05  1.263e-06  4.457e-03  4.246e-06
  -2.110e-05  1.950e-05  6.629e-05  1.585e-05  1.558e-05 -1.136e-05
   4.701e-06 -8.457e-06 -6.864e-06  2.228e-01  2.534e-01  1.248e-01
   1.799e-03  1.660e-03  7.767e-02  2.502e-01  1.394e-02  4.898e-03
   6.287e-01  6.193e-01 -2.568e-02  4.164e-02 -1.482e-02 -1.069e-03
  -4.926e-03 -1.761e-03 -7.135e-03 -1.165e-03  8.370e-03]]
```

In [33]:
```
scaler = Normalizer().fit(T)
testT = scaler.transform(T)
# summarize transformed data
np.set_printoptions(precision=3)
print(testT[0:5,:])
```

```
[[ 1.441e-02  1.470e-01  4.589e-03 -9.146e-02  7.617e-02  4.908e-06
   1.620e-04 -1.275e-05  1.122e-02 -4.355e-05  7.025e-03 -1.491e-04
  -6.596e-05 -2.974e-05  2.818e-05  1.361e-04  5.356e-05 -6.036e-06
   5.264e-06 -1.857e-05  7.157e-05  1.482e-02  3.223e-01  6.308e-02
   3.294e-02  1.403e-01  8.743e-02 -3.128e-02  5.229e-03  1.354e-01
   4.607e-01  6.397e-01  1.909e-01  3.835e-01  1.208e-02  4.840e-02
   2.892e-03 -2.581e-03 -8.478e-03  1.943e-02 -6.686e-05]
 [ 1.337e-02  8.240e-02  2.640e-02  9.857e-02  1.351e-01  2.170e-06
  -2.445e-05  3.008e-05  1.792e-05  1.158e-04  6.046e-05 -1.835e-04
  -5.357e-05  3.423e-05  6.946e-05  1.155e-02  2.830e-05  4.171e-06
  -1.286e-05 -1.897e-05 -3.020e-05 -4.864e-02  2.587e-01  3.587e-03
   3.878e-03  3.123e-02  6.444e-03  1.324e-01  1.159e-03  4.329e-02
   6.805e-01  4.855e-01  4.069e-01  6.280e-03 -8.709e-03  3.054e-02
   8.595e-02 -3.119e-03 -3.927e-02  1.200e-02  9.913e-04]
 [ 1.629e-02  1.944e-01  7.055e-02  5.717e-01 -1.117e-01  1.666e-05
   5.109e-05 -4.651e-05 -3.676e-05  8.344e-03  7.898e-03  9.333e-03
  -5.460e-03 -7.562e-05  9.036e-05  2.920e-05  4.959e-05  8.541e-03
  -8.978e-06 -1.339e-05  8.622e-05 -1.412e-01  1.108e-01  1.381e-01
   3.493e-03  1.182e-01  2.086e-02  1.173e-01  2.262e-02 -1.031e-02
   5.130e-01  6.527e-02  4.626e-02  7.687e-02  2.811e-01  9.272e-02
   1.021e-02  5.286e-02  4.109e-01  4.948e-02  8.363e-04]
 [ 9.695e-03  1.194e-01  3.516e-02  4.700e-01  1.043e-01 -2.824e-05
  -1.511e-05  1.508e-05  1.995e-03  9.839e-05  4.325e-03 -7.799e-05
  -5.662e-05 -5.346e-05  1.070e-02  3.892e-05  1.033e-05 -8.095e-05
   4.569e-06 -2.365e-05 -6.221e-07 -3.648e-02  1.079e-02  5.264e-04
   6.172e-02  2.502e-04  7.350e-02  1.181e-01 -8.860e-04  1.807e-01
   6.278e-01  3.183e-01  3.506e-01  1.354e-02  2.242e-01  3.514e-02
   3.821e-03  1.857e-03 -7.719e-03  1.585e-01  8.987e-04]
 [ 2.003e-02  2.707e-01  4.914e-02 -8.741e-02  7.294e-02  9.376e-06
   1.103e-04 -1.090e-06 -5.183e-04 -4.080e-05 -6.073e-04 -1.674e-04
   3.798e-03  9.041e-05 -1.175e-05  3.792e-05  2.809e-05  8.696e-05
   8.937e-06  2.919e-05  6.677e-03 -4.635e-02  1.139e-01  1.227e-03
   3.500e-03  2.516e-03  5.708e-03  1.244e-01  1.163e-01 -1.450e-02
   6.429e-01  4.936e-01 -1.336e-02  2.874e-03  2.815e-01  1.931e-04
  -6.319e-03  1.347e-01  2.755e-01  1.899e-01 -2.663e-04]]
```

In [34]:
```
y_train = np.array(Y)
y_test = np.array(C)
```

In [35]:
```
# reshape input to be [samples, time steps, features]
X_train = np.reshape(trainX, (trainX.shape[0], 1, trainX.shape[1]))
X_test = np.reshape(testT, (testT.shape[0], 1, testT.shape[1]))
```

In [36]:
```
batch_size = 32
```

In [37]:
```
# 1. define the network
model = Sequential()
model.add(LSTM(4,input_dim=41))  # try using a GRU instead, for fun
model.add(Dropout(0.1))
model.add(Dense(1))
model.add(Activation('sigmoid'))
print(model.get_config())
```

```
{'name': 'sequential_1', 'layers': [{'class_name': 'InputLayer', 'config': {'ba
tch_input_shape': (None, None, 41), 'dtype': 'float32', 'sparse': False, 'ragge
d': False, 'name': 'lstm_1_input'}}, {'class_name': 'LSTM', 'config': {'name':
'lstm_1', 'trainable': True, 'batch_input_shape': (None, None, 41), 'dtype': 'f
loat32', 'return_sequences': False, 'return_state': False, 'go_backwards': Fals
e, 'stateful': False, 'unroll': False, 'time_major': False, 'units': 4, 'activa
tion': 'tanh', 'recurrent_activation': 'sigmoid', 'use_bias': True, 'kernel_ini
tializer': {'class_name': 'GlorotUniform', 'config': {'seed': None}}, 'recurren
t_initializer': {'class_name': 'Orthogonal', 'config': {'gain': 1.0, 'seed': No
ne}}, 'bias_initializer': {'class_name': 'Zeros', 'config': {}}, 'unit_forget_b
ias': True, 'kernel_regularizer': None, 'recurrent_regularizer': None, 'bias_re
gularizer': None, 'activity_regularizer': None, 'kernel_constraint': None, 'rec
urrent_constraint': None, 'bias_constraint': None, 'dropout': 0.0, 'recurrent_d
ropout': 0.0, 'implementation': 2}}, {'class_name': 'Dropout', 'config': {'nam
e': 'dropout_1', 'trainable': True, 'dtype': 'float32', 'rate': 0.1, 'noise_sha
pe': None, 'seed': None}}, {'class_name': 'Dense', 'config': {'name': 'dense_
1', 'trainable': True, 'dtype': 'float32', 'units': 1, 'activation': 'linear',
'use_bias': True, 'kernel_initializer': {'class_name': 'GlorotUniform', 'confi
g': {'seed': None}}, 'bias_initializer': {'class_name': 'Zeros', 'config': {}},
'kernel_regularizer': None, 'bias_regularizer': None, 'activity_regularizer': N
one, 'kernel_constraint': None, 'bias_constraint': None}}, {'class_name': 'Acti
vation', 'config': {'name': 'activation_1', 'trainable': True, 'dtype': 'float3
2', 'activation': 'sigmoid'}}]}
```

In [38]:
```
# try using different optimizers and different optimizer configs
model.compile(loss='binary_crossentropy',optimizer='adam',metrics=['accuracy'])
checkpointer = callbacks.ModelCheckpoint(filepath="results/checkpoint-{epoch:02d}
csv_logger = CSVLogger('results/training_set_iranalysis.csv',separator=',', apper
model.fit(X_train, y_train, batch_size=batch_size, epochs=10, validation_data=(X_
model.save("results/lstm1layer_model.hdf5")

loss, accuracy = model.evaluate(X_test, y_test)
print("\nLoss: %.2f, Accuracy: %.2f%%" % (loss, accuracy*4000))
y_pred = model.predict(X_test)
classes_x=np.argmax(y_pred, axis=1)
np.savetxt('lstm1predicted.txt', y_pred, fmt='%01d')
```

```
Epoch 1/10
63/63 [==============================] - 9s 36ms/step - loss: 0.5476 - accurac
y: 0.0000e+00 - val_loss: -0.1486 - val_accuracy: 0.0000e+00

Epoch 00001: val_accuracy improved from -inf to 0.00000, saving model to result
s\checkpoint-01.hdf5
Epoch 2/10
63/63 [==============================] - 1s 11ms/step - loss: 0.2695 - accurac
y: 0.0000e+00 - val_loss: -1.3877 - val_accuracy: 0.0000e+00

Epoch 00002: val_accuracy did not improve from 0.00000
Epoch 3/10
63/63 [==============================] - 1s 11ms/step - loss: -0.0180 - accurac
y: 0.0000e+00 - val_loss: -2.7819 - val_accuracy: 0.0000e+00

Epoch 00003: val_accuracy did not improve from 0.00000
Epoch 4/10
63/63 [==============================] - 1s 11ms/step - loss: -0.3845 - accurac
y: 0.0000e+00 - val_loss: -4.4075 - val_accuracy: 0.0000e+00

Epoch 00004: val_accuracy did not improve from 0.00000
Epoch 5/10
63/63 [==============================] - 1s 11ms/step - loss: -0.7871 - accurac
y: 0.0000e+00 - val_loss: -6.6218 - val_accuracy: 0.0000e+00

Epoch 00005: val_accuracy did not improve from 0.00000
Epoch 6/10
63/63 [==============================] - 1s 12ms/step - loss: -1.2860 - accurac
y: 0.0000e+00 - val_loss: -8.6142 - val_accuracy: 0.0000e+00

Epoch 00006: val_accuracy did not improve from 0.00000
Epoch 7/10
63/63 [==============================] - 1s 12ms/step - loss: -1.7663 - accurac
y: 0.0000e+00 - val_loss: -10.9196 - val_accuracy: 0.0000e+00

Epoch 00007: val_accuracy did not improve from 0.00000
Epoch 8/10
63/63 [==============================] - 1s 15ms/step - loss: -2.3025 - accurac
y: 0.0000e+00 - val_loss: -13.1058 - val_accuracy: 0.0000e+00

Epoch 00008: val_accuracy did not improve from 0.00000
Epoch 9/10
63/63 [==============================] - 1s 12ms/step - loss: -2.6582 - accurac
```

```
y: 0.0000e+00 - val_loss: -15.1117 - val_accuracy: 0.0000e+00

Epoch 00009: val_accuracy did not improve from 0.00000
Epoch 10/10
63/63 [==============================] - 1s 9ms/step - loss: -3.1830 - accurac
y: 0.0000e+00 - val_loss: -17.1921 - val_accuracy: 0.0000e+00

Epoch 00010: val_accuracy did not improve from 0.00000
16/16 [==============================] - 0s 3ms/step - loss: -17.1921 - accurac
y: 0.0000e+00

Loss: -17.19, Accuracy: 0.00%
```

In [ ]:
```
# The LSTM model is deceived with 0.00% accuracy score
```