

# Sentiment Analysis using LSTM

## Import the required libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from keras.datasets import imdb
from keras.utils import pad_sequences
from keras.models import Sequential
from keras.layers import LSTM, Embedding, Dense, Dropout
```

## Set the Hyper parameter

```
MAX_FEATURES = 5000 # number of words to keep from the dataset
MAXLEN = 250 # maximum length of a sequence
BATCH_SIZE = 32
EPOCHS = 3
```

## Load the IMDB dataset

```
(X_train, y_train), (X_test, y_test) = imdb.load_data(num_words=MAX_FEATURES)
```

## Pad sequences to the same length

```
X_train = pad_sequences(X_train, maxlen=MAXLEN)
X_test = pad_sequences(X_test, maxlen=MAXLEN)
```

## Define the model

```
model = Sequential()
model.add(Embedding(input_dim=MAX_FEATURES, output_dim=128, input_length=MAXLEN))
model.add(LSTM(units=64, dropout=0.2, recurrent_dropout=0.2))
model.add(Dense(units=1, activation='sigmoid'))
```

WARNING:tensorflow:Layer lstm\_1 will not use cuDNN kernels since it doesn't meet the criteria. It will use a generic GPU kernel as

## Compile the model

```
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
```

## Train the model

[+ Code](#)
[+ Text](#)

```
history = model.fit(X_train, y_train, batch_size=BATCH_SIZE, epochs=EPOCHS, validation_data=(X_test, y_test))
```

```
Epoch 1/3
782/782 [=====] - 987s 1s/step - loss: 0.4168 - accuracy: 0.8081 - val_loss: 0.3421 - val_accuracy: 0.8582
Epoch 2/3
782/782 [=====] - 961s 1s/step - loss: 0.3028 - accuracy: 0.8750 - val_loss: 0.3418 - val_accuracy: 0.8609
Epoch 3/3
782/782 [=====] - 973s 1s/step - loss: 0.2529 - accuracy: 0.9013 - val_loss: 0.3792 - val_accuracy: 0.8501
```

## Evaluate the model on test data

```
score, acc = model.evaluate(X_test, y_test, batch_size=BATCH_SIZE)
print('Test score:', score)
print('Test accuracy:', acc)
```

```
782/782 [=====] - 62s 79ms/step - loss: 0.3792 - accuracy: 0.8501
Test score: 0.3792017102241516
Test accuracy: 0.8500800132751465
```