Ex. No. 6	LSTM
Date of Exercise	**07/09/2025**

Aim:

To train an LSTM network for generating text based on Shakespeare's works.

Description:

LSTM networks are a type of RNN specialized for handling long-term dependencies. We use a character-level LSTM model to generate Shakespeare-style text by training on his works.

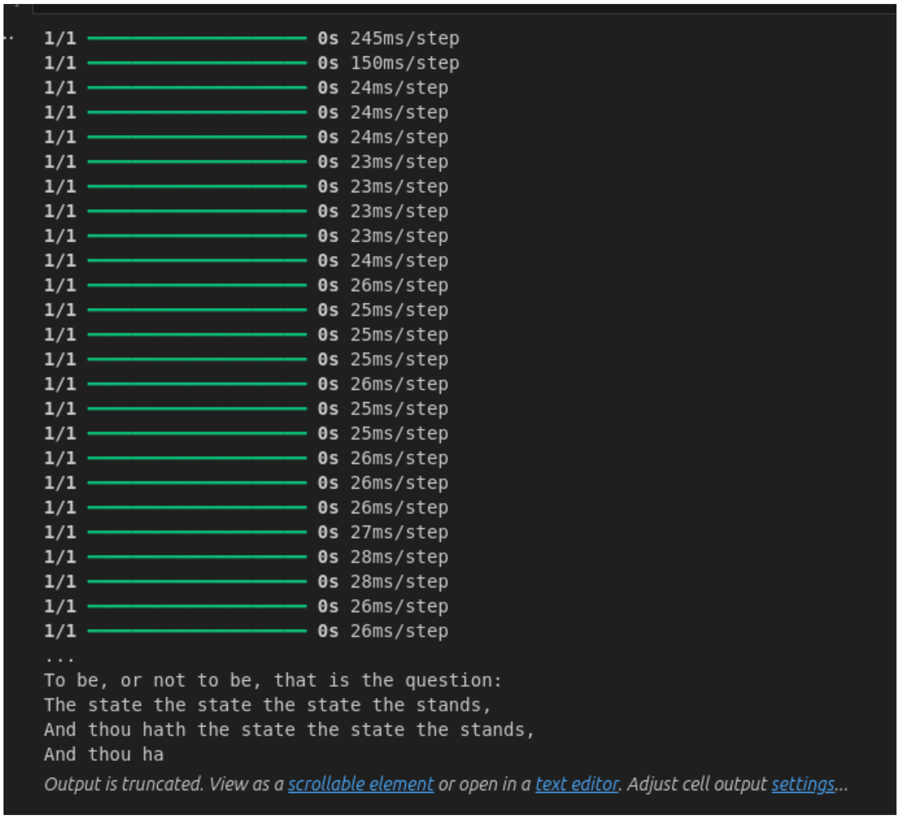
Code:

```
import tensorflow as tf
from tensorflow.keras.preprocessing.sequence import pad_sequences
from tensorflow.keras.utils import to_categorical
import numpy as np
# Load dataset

path_to_file = tf.keras.utils.get_file("shakespeare.txt",
"https://raw.githubusercontent.com/karpathy/char-
rnn/master/data/tinyshakespeare/input.txt")

text = open(path_to_file, 'rb').read().decode(encoding='utf-8')
# Preprocess data
```

```
chars = sorted(set(text))
char2idx = {c: i for i, c in enumerate(chars)}
idx2char = np.array(chars)
# Prepare training data
seq_length = 100
sequences = []
for i in range(len(text) - seq_length):
sequences.append([char2idx[c] for c in text[i:i+seq_length]])
X = np.array([seq[:-1] for seq in sequences])
y = np.array([seq[-1] for seq in sequences])
y = to_categorical(y, num_classes=len(chars))
# Build LSTM model
model = tf.keras.Sequential([
tf.keras.layers.Embedding(len(chars), 64, input_length=seq_length-1),
tf.keras.layers.LSTM(256, return_sequences=True),

tf.keras.layers.LSTM(256),
tf.keras.layers.Dense(len(chars), activation='softmax')
])
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
model.fit(X, y, epochs=10, batch_size=128)
# Generate text
def generate_text(seed_text, num_chars):
for _ in range(num_chars):
seed_seq = np.array([char2idx[c] for c in seed_text[-seq_length+1:]])
seed_seq = seed_seq.reshape(1, -1)
pred = model.predict(seed_seq)
next_char = idx2char[np.argmax(pred)]
seed_text += next_char
return seed_text
```

```
print(generate_text("To be, or not to be: that is the question.", 100))
```

Sample Output:



```
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 245ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 150ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 24ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 24ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 24ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 23ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 23ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 23ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 23ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 24ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 26ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 25ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 25ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 25ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 26ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 25ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 25ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 26ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 26ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 26ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 27ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 28ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 28ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 26ms/step
1/1 ━━━━━━━━━━━━━━━━━━━━ 0s 26ms/step
...
To be, or not to be, that is the question:
The state the state the state the stands,
And thou hath the state the state the stands,
And thou ha
```

Output is truncated. View as a scrollable element or open in a text editor. Adjust cell output settings...

Youtube Link

Long Short Term memory in a simple Way ;)

Result

The code for Long Short Term Memory is Done successfully and the output is been verified