LETS GROW MORE INTERNSHIP

AUTHOR - AYUSH CHHOKER

Next Word Prediction

Importing Datasets:

```
import tensorflow as tf
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.layers import Embedding, LSTM, Dense
from tensorflow.keras.models import Sequential
from tensorflow.keras.utils import to_categorical
from tensorflow.keras.optimizers import Adam
import pickle
import numpy as np
import os
```

```
file = open("metamorphosis_clean.txt", "r", encoding = "utf8")
lines = []

for i in file:
    lines.append(i)

print("The First Line: ", lines[0])
print("The Last Line: ", lines[-1])
```

```
The First Line:  One morning, when Gregor Samsa woke from troubled dreams, he found

The Last Line:  first to get up and stretch out her young body.
```

```
data = ""

for i in lines:
    data = ' '. join(lines)

data = data.replace('\n', '').replace('\r', '').replace('\ufeff', '')
data[:360]
```

```
import string

translator = str.maketrans(string.punctuation, ' '*len(string.punctuation)) #map punctuati
new_data = data.translate(translator)

new_data[:500]
```

```
z = []

for i in data.split():
    if i not in z:
        z.append(i)

data = ' '.join(z)
data[:500]
```

tokenizer

```
tokenizer = Tokenizer()
tokenizer.fit_on_texts([data])

# saving the tokenizer for predict function.
pickle.dump(tokenizer, open('tokenizer1.pkl', 'wb'))

sequence_data = tokenizer.texts_to_sequences([data])[0]
sequence_data[:10]


vocab_size = len(tokenizer.word_index) + 1
print(vocab_size)


sequences = []

for i in range(1, len(sequence_data)):
    words = sequence_data[i-1:i+1]
    sequences.append(words)

print("The Length of sequences are: ", len(sequences))
sequences = np.array(sequences)
sequences[:10]


X = []
y = []

for i in sequences:
    X.append(i[0])
    y.append(i[1])

X = np.array(X)
y = np.array(y)


print("The Data is: ", X[:5])
print("The responses are: ", y[:5])


y = to_categorical(y, num_classes=vocab_size)
y[:5]
```

Creating the Model:

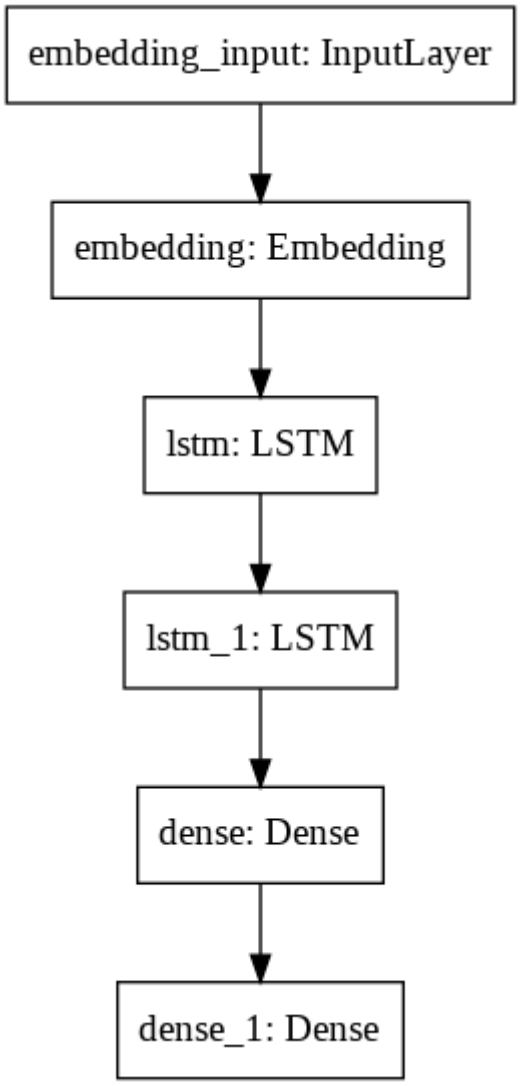
```
model = Sequential()
model.add(Embedding(vocab_size, 10, input_length=1))
model.add(LSTM(1000, return_sequences=True))
model.add(LSTM(1000))
model.add(Dense(1000, activation="relu"))
model.add(Dense(vocab_size, activation="softmax"))


model.summary()
```

Plot The Model:

```
from tensorflow import keras
from keras.utils.vis_utils import plot_model

keras.utils.plot_model(model, to_file='model.png', show_layer_names=True)
```



Callbacks:

```
from tensorflow.keras.callbacks import ModelCheckpoint
from tensorflow.keras.callbacks import ReduceLROnPlateau
from tensorflow.keras.callbacks import TensorBoard

checkpoint = ModelCheckpoint("nextword1.h5", monitor='loss', verbose=1,
    save_best_only=True, mode='auto')

reduce = ReduceLROnPlateau(monitor='loss', factor=0.2, patience=3, min_lr=0.0001, verbose

logdir='logsnextword1'
tensorboard_Visualization = TensorBoard(log_dir=logdir)
```

Compile The Model:

```
model.compile(loss="categorical_crossentropy", optimizer=Adam(lr=0.001))
```

```
/usr/local/lib/python3.7/dist-packages/keras/optimizer_v2/optimizer_v2.py:356: UserW
  "The `lr` argument is deprecated, use `learning_rate` instead.")
```

FIT THE MODEL

```
model.fit(X, y, epochs=150, batch_size=64, callbacks=[checkpoint, reduce, tensorboard_Visu
```

```
61/61 [==============================] - 14s 229ms/step - loss: 0.6182

Epoch 00135: loss did not improve from 0.61690
Epoch 136/150
61/61 [==============================] - 14s 227ms/step - loss: 0.6178

Epoch 00136: loss did not improve from 0.61690
Epoch 137/150
61/61 [==============================] - 14s 222ms/step - loss: 0.6191

Epoch 00137: loss did not improve from 0.61690
Epoch 138/150
61/61 [==============================] - 14s 226ms/step - loss: 0.6179

Epoch 00138: loss did not improve from 0.61690
Epoch 139/150
61/61 [==============================] - 14s 226ms/step - loss: 0.6183

Epoch 00139: loss did not improve from 0.61690
Epoch 140/150
61/61 [==============================] - 14s 225ms/step - loss: 0.6172

Epoch 00140: loss did not improve from 0.61690
Epoch 141/150
61/61 [==============================] - 14s 224ms/step - loss: 0.6182

Epoch 00141: loss did not improve from 0.61690
```

```
Epoch 142/150
61/61 [==============================] - 14s 225ms/step - loss: 0.6177

Epoch 00142: loss did not improve from 0.61690
Epoch 143/150
61/61 [==============================] - 14s 227ms/step - loss: 0.6174


Epoch 00143: loss did not improve from 0.61690
Epoch 144/150
61/61 [==============================] - 14s 226ms/step - loss: 0.6165

Epoch 00144: loss improved from 0.61690 to 0.61654, saving model to nextword1.h5
Epoch 145/150
61/61 [==============================] - 14s 232ms/step - loss: 0.6170

Epoch 00145: loss did not improve from 0.61654
Epoch 146/150
61/61 [==============================] - 14s 223ms/step - loss: 0.6169

Epoch 00146: loss did not improve from 0.61654
Epoch 147/150
61/61 [==============================] - 14s 226ms/step - loss: 0.6173

Epoch 00147: loss did not improve from 0.61654
Epoch 148/150
61/61 [==============================] - 14s 231ms/step - loss: 0.6171

Epoch 00148: loss did not improve from 0.61654
Epoch 149/150
61/61 [==============================] - 14s 223ms/step - loss: 0.6171

Epoch 00149: loss did not improve from 0.61654
Epoch 150/150
```