

```
In [ ]: from __future__ import print_function

from keras.preprocessing import sequence
from keras.models import Sequential
from keras.layers import Dense, Embedding
from keras.layers import LSTM
from keras.datasets import imdb

max_features = 20000

# обрезание текстов после данного количества слов (среди top max_features на
# иболее используемые слова)
maxlen = 80
batch_size = 128 # увеличьте значение для ускорения обучения

print('Загрузка данных...')
(x_train, y_train), (x_test, y_test) = imdb.load_data(num_words=max_features)
print(len(x_train), 'тренировочные последовательности')
print(len(x_test), 'тестовые последовательности')

print('Pad последовательности (примеров в x единицу времени)')
x_train = sequence.pad_sequences(x_train, maxlen=maxlen)
x_test = sequence.pad_sequences(x_test, maxlen=maxlen)
print('x_train shape:', x_train.shape)
print('x_test shape:', x_test.shape)

print('Построение модели...')
model = Sequential()
model.add(Embedding(max_features, 128))
model.add(LSTM(128, dropout=0.2, recurrent_dropout=0.2))
model.add(Dense(64, activation='sigmoid')) # прогон 2

model.add(Dense(1, activation='sigmoid'))

# стоит попробовать использовать другие оптимайзер и другие конфигурации оптимайзеров
model.compile(loss='binary_crossentropy',
              optimizer='adam',
              metrics=['accuracy'])

print('Процесс обучения...')
model.fit(x_train, y_train,
          batch_size=batch_size,
          epochs=50, # увеличьте при необходимости
          validation_data=(x_test, y_test))
score, acc = model.evaluate(x_test, y_test,
                             batch_size=batch_size)
print('Результат при тестировании:', score)
print('Тестовая точность:', acc)
```

```
Загрузка данных...
25000 тренировочные последовательности
25000 тестовые последовательности
Pad последовательности (примеров в x единицу времени)
x_train shape: (25000, 80)
x_test shape: (25000, 80)
Построение модели...
Процесс обучения...

/home/roman/anaconda3/lib/python3.7/site-packages/tensorflow_core/python/framework/indexed_slices.py:433: UserWarning: Converting sparse IndexedSlices to a dense Tensor of unknown shape. This may consume a large amount of memory.
  "Converting sparse IndexedSlices to a dense Tensor of unknown shape. "
```

```
Train on 25000 samples, validate on 25000 samples
Epoch 1/50
25000/25000 [=====] - 93s 4ms/step - loss: 0.4919 -
accuracy: 0.7558 - val_loss: 0.3880 - val_accuracy: 0.8283
Epoch 2/50
25000/25000 [=====] - 95s 4ms/step - loss: 0.3074 -
accuracy: 0.8750 - val_loss: 0.3739 - val_accuracy: 0.8363
Epoch 3/50
25000/25000 [=====] - 91s 4ms/step - loss: 0.2334 -
accuracy: 0.9121 - val_loss: 0.3961 - val_accuracy: 0.8312
Epoch 4/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.1902 -
accuracy: 0.9301 - val_loss: 0.4431 - val_accuracy: 0.8250
Epoch 5/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.1618 -
accuracy: 0.9417 - val_loss: 0.4899 - val_accuracy: 0.8246
Epoch 6/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.1211 -
accuracy: 0.9579 - val_loss: 0.5957 - val_accuracy: 0.8128
Epoch 7/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.1023 -
accuracy: 0.9643 - val_loss: 0.5750 - val_accuracy: 0.8118
Epoch 8/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.0901 -
accuracy: 0.9690 - val_loss: 0.5570 - val_accuracy: 0.8056
Epoch 9/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.0697 -
accuracy: 0.9771 - val_loss: 0.6790 - val_accuracy: 0.8119
Epoch 10/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.0592 -
accuracy: 0.9814 - val_loss: 0.7489 - val_accuracy: 0.8119
Epoch 11/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.0442 -
accuracy: 0.9866 - val_loss: 0.7955 - val_accuracy: 0.8070
Epoch 12/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.0428 -
accuracy: 0.9860 - val_loss: 0.7940 - val_accuracy: 0.8012
Epoch 13/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.0365 -
accuracy: 0.9884 - val_loss: 0.8295 - val_accuracy: 0.8068
Epoch 14/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.0348 -
accuracy: 0.9885 - val_loss: 0.8926 - val_accuracy: 0.8080
Epoch 15/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.0328 -
accuracy: 0.9898 - val_loss: 0.9253 - val_accuracy: 0.8083
Epoch 16/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.0324 -
accuracy: 0.9898 - val_loss: 0.8646 - val_accuracy: 0.7989
Epoch 17/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.0312 -
accuracy: 0.9901 - val_loss: 0.9131 - val_accuracy: 0.8043
Epoch 18/50
25000/25000 [=====] - 89s 4ms/step - loss: 0.0216 -
accuracy: 0.9935 - val_loss: 0.9557 - val_accuracy: 0.8063
Epoch 19/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.0189 -
accuracy: 0.9947 - val_loss: 1.0105 - val_accuracy: 0.8031
Epoch 20/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.0227 -
accuracy: 0.9928 - val_loss: 1.0400 - val_accuracy: 0.8067
Epoch 21/50
25000/25000 [=====] - 88s 4ms/step - loss: 0.0236 -
accuracy: 0.9926 - val_loss: 1.0119 - val_accuracy: 0.8021
Epoch 22/50
25000/25000 [=====] - 90s 4ms/step - loss: 0.0140 -
accuracy: 0.9956 - val_loss: 1.0492 - val_accuracy: 0.8054
Epoch 23/50
```

In []: