

word embedding with keras lib

June 5, 2018

1 Word Embedding With keras lib

Hossein Yousefi

```
In [13]: from keras.datasets import imdb
         from keras import preprocessing
         from keras.models import Sequential
         from keras.layers import Flatten
         from keras.layers import Dense , Embedding

In [16]: max_features = 10000
         maxlen = 20

         (x_train,y_train), (x_test, y_test) = imdb.load_data(num_words=max_features)

In [17]: #pre processing
         x_train = preprocessing.sequence.pad_sequences(x_train, maxlen=maxlen)
         x_test = preprocessing.sequence.pad_sequences(x_test,maxlen=maxlen)

In [18]: network = Sequential()
         network.add(Embedding(10000, 8, input_length=maxlen))

         network.add(Flatten())

         network.add(Dense(1, activation='sigmoid'))
         network.compile(optimizer='rmsprop', loss='binary_crossentropy', metrics=['acc'])
         network.summary()

         history = network.fit(x_train, y_train,
                               epochs=10,
                               batch_size=32,
                               validation_split=0.2)
```

Layer (type)	Output Shape	Param #
embedding_3 (Embedding)	(None, 20, 8)	80000

```

flatten_3 (Flatten)          (None, 160)          0
-----
dense_3 (Dense)              (None, 1)            161
=====
Total params: 80,161
Trainable params: 80,161
Non-trainable params: 0

-----
Train on 20000 samples, validate on 5000 samples
Epoch 1/10
20000/20000 [=====] - 2s 81us/step - loss: 0.6759 - acc: 0.6048 - val.
Epoch 2/10
20000/20000 [=====] - 2s 88us/step - loss: 0.5658 - acc: 0.7426 - val.
Epoch 3/10
20000/20000 [=====] - 2s 84us/step - loss: 0.4752 - acc: 0.7807 - val.
Epoch 4/10
20000/20000 [=====] - 2s 76us/step - loss: 0.4263 - acc: 0.8078 - val.
Epoch 5/10
20000/20000 [=====] - 1s 73us/step - loss: 0.3930 - acc: 0.8258 - val.
Epoch 6/10
20000/20000 [=====] - 1s 73us/step - loss: 0.3668 - acc: 0.8395 - val.
Epoch 7/10
20000/20000 [=====] - 2s 76us/step - loss: 0.3435 - acc: 0.8532 - val.
Epoch 8/10
20000/20000 [=====] - 2s 76us/step - loss: 0.3223 - acc: 0.8656 - val.
Epoch 9/10
20000/20000 [=====] - 1s 74us/step - loss: 0.3023 - acc: 0.8767 - val.
Epoch 10/10
20000/20000 [=====] - 1s 73us/step - loss: 0.2839 - acc: 0.8860 - val.

```