In [1]:

```
from __future__ import print_function
from keras.models import Sequential, Model
from keras.layers import Dense, Activation, Dropout
from keras.layers import LSTM, Input, Flatten, Bidirectional
from keras.layers.normalization import BatchNormalization
from keras.optimizers import Adam
from keras.callbacks import EarlyStopping, ModelCheckpoint
from keras.metrics import categorical_accuracy
import numpy as np
import random
import sys
import os
import time
import codecs
import collections
from six.moves import cPickle
import en_core_web_sm


# In[12]:


#import spacy, and french model
import spacy
nlp = en_core_web_sm.load()


# In[17]:

save_dir = 'models' # directory to store models
seq_length = 30 # sequence length
sequences_step = 1 #step to create sequences


# In[21]:

try:
    vocab_file = os.path.join(save_dir, "words_vocab.pkl")
except:
    print("Vocab file does not exist")
    pass
# In[22]:


def create_wordlist(doc):
    wl = []
    for word in doc:
        if word.text not in ("\n","\n\n",'\u2009','\xa0'):
            wl.append(word.text.lower())
    return wl


# In[23]:

input_file = 'episodes\\HP1.txt'
wordlist = []
#read data
with codecs.open(input_file, "r") as f:
  data = f.read()
```

```
#create sentences
doc = nlp(data)
wl = create_wordlist(doc)
wordlist = wordlist + wl


# In[24]:


# count the number of words
word_counts = collections.Counter(wordlist)

# Mapping from index to word : that's the vocabulary
vocabulary_inv = [x[0] for x in word_counts.most_common()]
vocabulary_inv = list(sorted(vocabulary_inv))

# Mapping from word to index
vocab = {x: i for i, x in enumerate(vocabulary_inv)}
words = [x[0] for x in word_counts.most_common()]

#size of the vocabulary
vocab_size = len(words)
print("vocab size: ", vocab_size)

#save the words and vocabulary
with open(os.path.join(vocab_file), 'wb') as f:
    cPickle.dump((words, vocab, vocabulary_inv), f)


# In[25]:


#create sequences
sequences = []
next_words = []
for i in range(0, len(wordlist) - seq_length, sequences_step):
    sequences.append(wordlist[i: i + seq_length])
    next_words.append(wordlist[i + seq_length])

print('nb sequences:', len(sequences))


# In[26]:


X = np.zeros((len(sequences), seq_length, vocab_size), dtype=np.bool)
y = np.zeros((len(sequences), vocab_size), dtype=np.bool)
for i, sentence in enumerate(sequences):
    for t, word in enumerate(sentence):
        X[i, t, vocab[word]] = 1
    y[i, vocab[next_words[i]]] = 1


# In[27]:


def bidirectional_lstm_model(seq_length, vocab_size):
    print('Build LSTM model.')
    model = Sequential()
    model.add(Bidirectional(LSTM(rnn_size, activation="relu"),input_shape=(seq_length, voca
    model.add(Dropout(0.6))
    model.add(Dense(vocab_size))
    model.add(Activation('softmax'))

    optimizer = Adam(lr=learning_rate)
    callbacks=[EarlyStopping(patience=2, monitor='val_loss')]
    model.compile(loss='categorical_crossentropy', optimizer=optimizer, metrics=[categorica
    print("model built!")
    return model


# In[57]:


rnn_size = 256 # size of RNN
batch_size = 32 # minibatch size
seq_length = 30 # sequence length
num_epochs = 50 # number of epochs
learning_rate = 0.001 #learning rate
sequences_step = 1 #step to create sequences


# In[58]:


md = bidirectional_lstm_model(seq_length, vocab_size)
md.summary()


# In[ ]:


#fit the model
callbacks=[EarlyStopping(patience=4, monitor='val_loss'),
           ModelCheckpoint(filepath=save_dir + "/" + 'my_model_gen_sentences_lstm.{epoch:02
                           monitor='val_loss', verbose=0, mode='auto', period=2)]
history = md.fit(X, y,
                 batch_size=batch_size,
                 shuffle=True,
                 epochs=num_epochs,
                 callbacks=callbacks,
                 validation_split=0.01)




#save the modelnn
md.save(save_dir + "/" + 'my_model_gen_sentences_lstm.final.hdf5')




#load vocabulary
print("loading vocabulary...")
try:
    vocab_file = os.path.join(save_dir, "words_vocab.pkl")
    with open(os.path.join(save_dir, 'words_vocab.pkl'), 'rb') as f:
        words, vocab, vocabulary_inv = cPickle.load(f)
except:
    pass




vocab_size = len(words)
```

```
from keras.models import load_model
# Load the model
print("loading model...")
model = load_model(save_dir + "/" + 'my_model_gen_sentences_lstm.final.hdf5')


def sample(preds, temperature=1.0):
    preds = np.asarray(preds).astype('float64')
    preds = np.log(preds) / temperature
    exp_preds = np.exp(preds)
    preds = exp_preds / np.sum(exp_preds)
    probas = np.random.multinomial(1, preds, 1)
    return np.argmax(probas)
```

```
Using TensorFlow backend.
C:\Users\mayan\AppData\Roaming\Python\Python37\site-packages\tensorflow\pyth
on\framework\dtypes.py:516: FutureWarning: Passing (type, 1) or '1type' as a
synonym of type is deprecated; in a future version of numpy, it will be unde
rstood as (type, (1,)) / '(1,)type'.
  _np_qint8 = np.dtype([("qint8", np.int8, 1)])
C:\Users\mayan\AppData\Roaming\Python\Python37\site-packages\tensorflow\pyth
on\framework\dtypes.py:517: FutureWarning: Passing (type, 1) or '1type' as a
synonym of type is deprecated; in a future version of numpy, it will be unde
rstood as (type, (1,)) / '(1,)type'.
  _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
C:\Users\mayan\AppData\Roaming\Python\Python37\site-packages\tensorflow\pyth
on\framework\dtypes.py:518: FutureWarning: Passing (type, 1) or '1type' as a
synonym of type is deprecated; in a future version of numpy, it will be unde
rstood as (type, (1,)) / '(1,)type'.
  _np_qint16 = np.dtype([("qint16", np.int16, 1)])
C:\Users\mayan\AppData\Roaming\Python\Python37\site-packages\tensorflow\pyth
on\framework\dtypes.py:519: FutureWarning: Passing (type, 1) or '1type' as a
synonym of type is deprecated; in a future version of numpy, it will be unde
rstood as (type, (1,)) / '(1,)type'.
  _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
C:\Users\mayan\AppData\Roaming\Python\Python37\site-packages\tensorflow\pyth
on\framework\dtypes.py:520: FutureWarning: Passing (type, 1) or '1type' as a
synonym of type is deprecated; in a future version of numpy, it will be unde
rstood as (type, (1,)) / '(1,)type'.
  _np_qint32 = np.dtype([("qint32", np.int32, 1)])
C:\Users\mayan\AppData\Roaming\Python\Python37\site-packages\tensorflow\pyth
on\framework\dtypes.py:525: FutureWarning: Passing (type, 1) or '1type' as a
synonym of type is deprecated; in a future version of numpy, it will be unde
rstood as (type, (1,)) / '(1,)type'.
  np_resource = np.dtype([("resource", np.ubyte, 1)])
C:\Users\mayan\AppData\Roaming\Python\Python37\site-packages\tensorboard\com
pat\tensorflow_stub\dtypes.py:541: FutureWarning: Passing (type, 1) or '1typ
e' as a synonym of type is deprecated; in a future version of numpy, it will
be understood as (type, (1,)) / '(1,)type'.
  _np_qint8 = np.dtype([("qint8", np.int8, 1)])
C:\Users\mayan\AppData\Roaming\Python\Python37\site-packages\tensorboard\com
pat\tensorflow_stub\dtypes.py:542: FutureWarning: Passing (type, 1) or '1typ
e' as a synonym of type is deprecated; in a future version of numpy, it will
```

```
be understood as (type, (1,)) / '(1,)type'.
  _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
C:\Users\mayan\AppData\Roaming\Python\Python37\site-packages\tensorboard\com
pat\tensorflow_stub\dtypes.py:543: FutureWarning: Passing (type, 1) or '1typ
e' as a synonym of type is deprecated; in a future version of numpy, it will
be understood as (type, (1,)) / '(1,)type'.
  _np_qint16 = np.dtype([("qint16", np.int16, 1)])
C:\Users\mayan\AppData\Roaming\Python\Python37\site-packages\tensorboard\com
pat\tensorflow_stub\dtypes.py:544: FutureWarning: Passing (type, 1) or '1typ
e' as a synonym of type is deprecated; in a future version of numpy, it will
be understood as (type, (1,)) / '(1,)type'.
  _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
C:\Users\mayan\AppData\Roaming\Python\Python37\site-packages\tensorboard\com
pat\tensorflow_stub\dtypes.py:545: FutureWarning: Passing (type, 1) or '1typ
e' as a synonym of type is deprecated; in a future version of numpy, it will
be understood as (type, (1,)) / '(1,)type'.
  _np_qint32 = np.dtype([("qint32", np.int32, 1)])
C:\Users\mayan\AppData\Roaming\Python\Python37\site-packages\tensorboard\com
pat\tensorflow_stub\dtypes.py:550: FutureWarning: Passing (type, 1) or '1typ
e' as a synonym of type is deprecated; in a future version of numpy, it will
be understood as (type, (1,)) / '(1,)type'.
  np_resource = np.dtype([("resource", np.ubyte, 1)])

vocab size:  3733
nb sequences: 40777
Build LSTM model.
WARNING:tensorflow:Large dropout rate: 0.6 (>0.5). In TensorFlow 2.x, dropou
t() uses dropout rate instead of keep_prob. Please ensure that this is inten
ded.
model built!
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
bidirectional_1 (Bidirection (None, 512)               8171520   
_________________________________________________________________
dropout_1 (Dropout)          (None, 512)               0         
_________________________________________________________________
dense_1 (Dense)              (None, 3733)              1915029   
_________________________________________________________________
activation_1 (Activation)    (None, 3733)              0         
=================================================================
Total params: 10,086,549
Trainable params: 10,086,549
Non-trainable params: 0
_________________________________________________________________
WARNING:tensorflow:From c:\users\mayan\appdata\local\programs\python\python3
7\lib\site-packages\keras\backend\tensorflow_backend.py:422: The name tf.glo
bal_variables is deprecated. Please use tf.compat.v1.global_variables instea
d.

Train on 40369 samples, validate on 408 samples
Epoch 1/50
40369/40369 [==============================] - 161s 4ms/step - loss: 6.3550
- categorical_accuracy: 0.0510 - val_loss: 6.0381 - val_categorical_accurac
y: 0.0833
Epoch 2/50
40369/40369 [==============================] - 155s 4ms/step - loss: 5.8264
- categorical_accuracy: 0.0875 - val_loss: 5.9484 - val_categorical_accurac
y: 0.0931
Epoch 3/50
```

```
40369/40369 [==============================] - 158s 4ms/step - loss: 5.6087
- categorical_accuracy: 0.1090 - val_loss: 5.8761 - val_categorical_accurac
y: 0.1275
Epoch 4/50
40369/40369 [==============================] - 157s 4ms/step - loss: 5.3876
- categorical_accuracy: 0.1364 - val_loss: 5.8572 - val_categorical_accurac
y: 0.1446
Epoch 5/50
40369/40369 [==============================] - 156s 4ms/step - loss: 5.1652
- categorical_accuracy: 0.1572 - val_loss: 5.7989 - val_categorical_accurac
y: 0.1618
Epoch 6/50
40369/40369 [==============================] - 157s 4ms/step - loss: 4.9518
- categorical_accuracy: 0.1766 - val_loss: 5.7743 - val_categorical_accurac
y: 0.1691
Epoch 7/50
40369/40369 [==============================] - 157s 4ms/step - loss: 4.7265
- categorical_accuracy: 0.1897 - val_loss: 5.8299 - val_categorical_accurac
y: 0.1814
Epoch 8/50
40369/40369 [==============================] - 157s 4ms/step - loss: 4.5517
- categorical_accuracy: 0.2035 - val_loss: 5.8274 - val_categorical_accurac
y: 0.1887
Epoch 9/50
40369/40369 [==============================] - 155s 4ms/step - loss: 4.2899
- categorical_accuracy: 0.2230 - val_loss: 5.8423 - val_categorical_accurac
y: 0.1863
Epoch 10/50
40369/40369 [==============================] - 155s 4ms/step - loss: 4.0527
- categorical_accuracy: 0.2412 - val_loss: 5.8758 - val_categorical_accurac
y: 0.1985
loading vocabulary...
loading model...
WARNING:tensorflow:Large dropout rate: 0.6 (>0.5). In TensorFlow 2.x, dropou
t() uses dropout rate instead of keep_prob. Please ensure that this is inten
ded.
Generating text with the following seed: "a a a a a a a a a a a a a a a a a
a a a a a a Invisibility Cloak on top of the tower"



---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
<ipython-input-1-0c7377959935> in <module>
    221     x = np.zeros((1, seq_length, vocab_size))
    222     for t, word in enumerate(sentence):
--> 223         x[0, t, vocab[word]] = 1.
    224     #print(x.shape)
    225 

KeyError: 'Invisibility'
```

In [9]:

```
import re
def untokenize(words):
    """
    Untokenizing a text undoes the tokenizing operation, restoring
    punctuation and spaces to the places that people expect them to be.
    Ideally, `untokenize(tokenize(text))` should be identical to `text`,
    except for line breaks.
    """
    text = ' '.join(words)
    step1 = text.replace("`` ", '"').replace(" ''", '"').replace('. . .',  '...')
    step2 = step1.replace(" ( ", " (").replace(" ) ", ") ")
    step3 = re.sub(r' ([.,:;?!%]+)([ \'"`])', r"\1\2", step2)
    step4 = re.sub(r' ([.,:;?!%]+)$', r"\1", step3)
    step5 = step4.replace(" '", "'").replace(" n't", "n't").replace(
         "can not", "cannot")
    step6 = step5.replace(" ` ", " '")
    return step6.strip()
```

In [19]:

```
#initiate sentences
seed_sentences = "They would have felt sorry for Hagrid when the time came for him to say g
generated = ''
sentence = []
for i in range (seq_length):
    sentence.append("a")
import nltk
from nltk import word_tokenize

seed = word_tokenize(seed_sentences)

for i in range(len(seed)):
    sentence[seq_length-i-1]=seed[len(seed)-i-1]




words_number = 500
generated =[]
#generate the text
for i in range(words_number):
    #create the vector
    x = np.zeros((1, seq_length, vocab_size))
    for t, word in enumerate(sentence):
        x[0, t, vocab[word.lower()]] = 1.
    #print(x.shape)

    #calculate next word
    preds = model.predict(x, verbose=0)[0]
    next_index = sample(preds, 0.1)
    next_word = vocabulary_inv[next_index]

    #add the next word to the text
    generated.append(next_word)
    # shift the sentence by one, and and the next word at its end
    sentence = sentence[1:] + [next_word]

print(" ".join(untokenize(generated).replace("\"","").split()))
```

```
. i'm you, said ron. i'm a a a a few a a a, and and a a little - a a of the
mirror, and the the mirror, and he'dn't be a the, and the mirror, and he cou
ldn't be a very a, and he had been a the mirror, and they had been a the mir
ror, and he wasn't be a the hat, and the troll, and the first years. he was
a in the mirror, and he'd been a a, and the first years. he wasn't be. i'm y
ou, said ron. he'd have been a nimbus two thousand. i'm you, said ron, i'm y
ou, said ron. i'm going to be a a a a the mirror, and he was going to be.
i'm you? said, said ron, harry, said ron, he said. i'm you, said ron. i'm wh
at you're going to be your. i'm you? i'm you, said ron. i'm you, said ron, h
arry's, but i'm i'm you, said ron. i'm you, said ron. i'm you, said ron, har
ry's got to the the mirror. i'm you, you're going to be a your, said ron.
i'm you've got to get a the mirror, but you're to be a the, said ron, and ro
n, and ron, and the the head of the hall, and the the hat, and the hall, and
the hall was the great hall, and the hall, and the hall, and the hat, the ha
t and the hat, and the hall was the hall, and the first years. they were a a
the hat, and the hat, and the hat, and the mirror, and the a the troll, and
a a a a, and the hat and the hat and the hat. harry had been the mirror, he
wasn't going to be. i'm you're not to be the hat, said ron. i'm you? i'm yo
u, said ron, i'm you, you're not to be you. i'm you, said ron. i'm you?
```

In [11]:

```
print(" ".join(untokenize(generated).replace("\"","").split()))
```

```
. i'm no, said ron, hagrid's your sorcerer's stone, he'dn't a the. hagrid, t
he yelled, but he's must be remembrall. he asked ron, but what's he's than,
said wood. i'm to the remembrall that, you've got all for the team. don't wa
nt to you, said hermione. i've got the see you, said ron. you're too not to
your i think i'm in the team, said ron. he said. oh, you're not me to you've
got to get. he is it. i'm i know you? well, said ron, harry's time to have t
o be of the back. i'm' - i've got the door, said ron, and harry had had to p
ercy on the gryffindor, but it's the too. harry's were the head, said ron, h
arry at once. the new. he's really've got to the the dragon's. you'll have t
o have been a a this, and snape wasn't have the the in a the few and that wa
s three of the gryffindor. i'm this, you'd be were no. ron, said ron. i'll n
ot him, said ron. i'm you get a it. i'm me - you're i've got to be i'm, said
ron. i'm your you, said ron. i'mn't come, said harry. i'm i'm' you, said ro
n, i'm you, you're going to be the house the first years in the mirror of an
d then i'll be the. i don't want to a know, said ron. i'm, said ron. how to
you, you've got to get a the away and year, that's the midnight your high, u
p? said ron, you've got the head, said ron, the gryffindor, said ron. i'm ab
out you, said ron, harry told ron. what are you? said, hagrid's us the you'r
e an''. said, said harry. ron, said ron, what i'm the other, said hagrid. yo
u've got to very going to tell what about it, you'll be you? i'll get me for
you, said harry, harry's be a his own. you've got to hear you will be you've
got about gryffindor, said ron, ron as they tree. then, but the harry, was,
but i've got to it, then, but they couldn't for? said, said harry. the house
on the the team, , but' harry's got to the and. so you could have have to be
charlie to be of the brooms. i'm trying to be going to be the high, harry an
d harry looked him. he couldn't a a you, you've got to a - of the head, and
ron had been this. he hadn't think they were, the the house. i'm he didn't s
ee you, said hermione, ron was, but he didn't be him. that was you, her from
the dog, but they were have been a was nimbus two thousand. ron, ron, i'll b
e stupid, said harry, but ron was in the the floor, and and the - quidditch
and a just and a, the eyes of the was broomstick, that was as they his long
as they were no with a hands, and they had a knocked. the went to a time to
have a the between. the whole of the troll, the floor. you've got to see yo
u, you're a'd seen, harry, said ron. he's got the the school, said harry. i
don't a a get it. you don't see you, said harry. the nimbus two thousand. po
tter, said ron. i'm, but you're a you around, i know you're. you're a, said
dumbledore. harry had about the found a which were they looked at up. harr
y's other. i don't see you, you're? said snape. i'm i've got to be seeker, i
think that was be can you? i don't you you? said hermione. i'm told you, sai
d harry. i've got to see
```

In [ ]:

```

```