In [2]:

```
import numpy
import sys
import nltk
nltk.download('stopwords')
from nltk.tokenize import RegexpTokenizer
from nltk.corpus import stopwords
from keras.models import Sequential
from keras.layers import Dense, Dropout, LSTM
from keras.utils import np_utils
from keras.callbacks import ModelCheckpoint
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
Using TensorFlow backend.
```

In [3]:

```
file = open("frankestein.2.txt").read()
```

In [4]:

```
def tokenize_words(input):
    input = input.lower()
    tokenizer = RegexpTokenizer(r'\w+')
    tokens = tokenizer.tokenize(input)
    filtered = filter(lambda token: token not in stopwords.words('english'), tokens)
    return " ".join(filtered)

processed_inputs = tokenize_words(file)
```

In [5]:

```
chars = sorted(list(set(processed_inputs)))
char_to_num = dict((c, i) for i, c in enumerate(chars))
```

In [6]:

```
input_len = len(processed_inputs)
vocab_len = len(chars)
print ("Total number of characters:", input_len)
print ("Total vocab:", vocab_len)
```

```
Total number of characters: 269995
Total vocab: 43
```

In [7]:

```
seq_length = 100
x_data = []
y_data = []
```

In [8]:

```
for i in range(0, input_len - seq_length, 1):
    in_seq = processed_inputs[i:i + seq_length]
    out_seq = processed_inputs[i + seq_length]
    x_data.append([char_to_num[char] for char in in_seq])
    y_data.append(char_to_num[out_seq])

n_patterns = len(x_data)
print ("Total Patterns:", n_patterns)
```

```
Total Patterns: 269895
```

In [9]:

```
X = numpy.reshape(x_data, (n_patterns, seq_length, 1))
X = X/float(vocab_len)
```

In [10]:

```
y = np_utils.to_categorical(y_data)
```

In [11]:

```
model = Sequential()
model.add(LSTM(256, input_shape=(X.shape[1], X.shape[2]), return_sequences=True))
model.add(Dropout(0.2))
model.add(LSTM(256, return_sequences=True))
model.add(Dropout(0.2))
model.add(LSTM(128))
model.add(Dropout(0.2))
model.add(Dense(y.shape[1], activation='softmax'))
```

```
WARNING: Logging before flag parsing goes to stderr.
W0523 05:47:10.594311 140415618766656 deprecation_wrapper.py:119] From /mnt/disks/user/anacon
da3/lib/python3.7/site-packages/keras/backend/tensorflow_backend.py:74: The name tf.get_defau
lt_graph is deprecated. Please use tf.compat.v1.get_default_graph instead.

W0523 05:47:10.733228 140415618766656 deprecation_wrapper.py:119] From /mnt/disks/user/anacon
da3/lib/python3.7/site-packages/keras/backend/tensorflow_backend.py:517: The name tf.placehol
der is deprecated. Please use tf.compat.v1.placeholder instead.

W0523 05:47:10.748069 140415618766656 deprecation_wrapper.py:119] From /mnt/disks/user/anacon
da3/lib/python3.7/site-packages/keras/backend/tensorflow_backend.py:4138: The name tf.random_
uniform is deprecated. Please use tf.random.uniform instead.

W0523 05:47:11.553309 140415618766656 deprecation_wrapper.py:119] From /mnt/disks/user/anacon
da3/lib/python3.7/site-packages/keras/backend/tensorflow_backend.py:133: The name tf.placehol
der_with_default is deprecated. Please use tf.compat.v1.placeholder_with_default instead.

W0523 05:47:11.561165 140415618766656 deprecation.py:506] From /mnt/disks/user/anaconda3/lib/
python3.7/site-packages/keras/backend/tensorflow_backend.py:3445: calling dropout (from tenso
rflow.python.ops.nn_ops) with keep_prob is deprecated and will be removed in a future versio
n.
Instructions for updating:
Please use `rate` instead of `keep_prob`. Rate should be set to `rate = 1 - keep_prob`.
```

In [12]:

```
model.compile(loss='categorical_crossentropy', optimizer='adam')
```

```
W0523 05:47:17.728994 140415618766656 deprecation_wrapper.py:119] From /mnt/disks/user/anacon
da3/lib/python3.7/site-packages/keras/optimizers.py:790: The name tf.train.Optimizer is depre
cated. Please use tf.compat.v1.train.Optimizer instead.

W0523 05:47:17.755992 140415618766656 deprecation_wrapper.py:119] From /mnt/disks/user/anacon
da3/lib/python3.7/site-packages/keras/backend/tensorflow_backend.py:3295: The name tf.log is
deprecated. Please use tf.math.log instead.
```

In [13]:

```
filepath = "model_weights_saved.hdf5"
checkpoint = ModelCheckpoint(filepath, monitor='loss', verbose=1, save_best_only=True, mode=
'min')
desired_callbacks = [checkpoint]
```

In [14]:

```
model.fit(X, y, epochs=15, batch_size=256, callbacks=desired_callbacks)
```

```
W0523 05:48:30.910070 140415618766656 deprecation.py:323] From /mnt/disks/user/anaconda3/lib/
python3.7/site-packages/tensorflow/python/ops/math_grad.py:1250: add_dispatch_support.<locals
>.wrapper (from tensorflow.python.ops.array_ops) is deprecated and will be removed in a futur
e version.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
```

```
Epoch 1/15
269895/269895 [==============================] - 701s 3ms/step - loss: 2.8426

Epoch 00001: loss improved from inf to 2.84260, saving model to model_weights_saved.hdf5
Epoch 2/15
269895/269895 [==============================] - 655s 2ms/step - loss: 2.5503

Epoch 00002: loss improved from 2.84260 to 2.55027, saving model to model_weights_saved.hdf5
Epoch 3/15
269895/269895 [==============================] - 617s 2ms/step - loss: 2.3698

Epoch 00003: loss improved from 2.55027 to 2.36980, saving model to model_weights_saved.hdf5
Epoch 4/15
269895/269895 [==============================] - 619s 2ms/step - loss: 2.2303

Epoch 00004: loss improved from 2.36980 to 2.23032, saving model to model_weights_saved.hdf5
Epoch 5/15
269895/269895 [==============================] - 618s 2ms/step - loss: 2.1281

Epoch 00005: loss improved from 2.23032 to 2.12810, saving model to model_weights_saved.hdf5
Epoch 6/15
269895/269895 [==============================] - 617s 2ms/step - loss: 2.0528

Epoch 00006: loss improved from 2.12810 to 2.05278, saving model to model_weights_saved.hdf5
Epoch 7/15
269895/269895 [==============================] - 617s 2ms/step - loss: 1.9901

Epoch 00007: loss improved from 2.05278 to 1.99011, saving model to model_weights_saved.hdf5
Epoch 8/15
269895/269895 [==============================] - 617s 2ms/step - loss: 2.5756

Epoch 00008: loss did not improve from 1.99011
Epoch 9/15
269895/269895 [==============================] - 612s 2ms/step - loss: 2.4471

Epoch 00009: loss did not improve from 1.99011
Epoch 10/15
269895/269895 [==============================] - 617s 2ms/step - loss: 2.3330

Epoch 00010: loss did not improve from 1.99011
Epoch 11/15
269895/269895 [==============================] - 616s 2ms/step - loss: 2.2406

Epoch 00011: loss did not improve from 1.99011
Epoch 12/15
269895/269895 [==============================] - 617s 2ms/step - loss: 2.1643

Epoch 00012: loss did not improve from 1.99011
Epoch 13/15
269895/269895 [==============================] - 616s 2ms/step - loss: 2.1030

Epoch 00013: loss did not improve from 1.99011
Epoch 14/15
269895/269895 [==============================] - 618s 2ms/step - loss: 2.0502

Epoch 00014: loss did not improve from 1.99011
Epoch 15/15
269895/269895 [==============================] - 618s 2ms/step - loss: 2.0057

Epoch 00015: loss did not improve from 1.99011
```

Out[14]:

```
<keras.callbacks.History at 0x7fb480eb7e80>
```

In [15]:

```
filename = "model_weights_saved.hdf5"
model.load_weights(filename)
model.compile(loss='categorical_crossentropy', optimizer='adam')
```

In [16]:

```
num_to_char = dict((i, c) for i, c in enumerate(chars))
```

In [17]:

```
start = numpy.random.randint(0, len(x_data) - 1)
pattern = x_data[start]
print("Random Seed:")
print("\"", ''.join([num_to_char[value] for value in pattern]), "\"")
```

```
Random Seed:
" nt persuasive words even power heart trust soul hellish form full treachery fiend like mali
ce hear c "
```

In [18]:

```
for i in range(1000):
    x = numpy.reshape(pattern, (1, len(pattern), 1))
    x = x / float(vocab_len)
    prediction = model.predict(x, verbose=0)
    index = numpy.argmax(prediction)
    result = num_to_char[index]
    seq_in = [num_to_char[value] for value in pattern]

    sys.stdout.write(result)

    pattern.append(index)
    pattern = pattern[1:len(pattern)]
```

```
onpanion seered shall seered shall seered shall seered shall seered shall seered shall seered
shall seered shall seered shall seered shall seered shall seered shall seered shall seered sh
all seered shall seered shall seered shall seered shall seered shall seered shall seered shal
l seered shall seered shall seered shall seered shall seered shall seered shall seered shall
seered shall seered shall seered shall seered shall seered shall seered shall seered shall se
ered shall seered shall seered shall seered shall seered shall seered shall seered shall seer
ed shall seered shall seered shall seered shall seered shall seered shall seered shall seered
shall seered shall seered shall seered shall seered shall seered shall seered shall seered sh
```