

1 Introduction of Shakespeare language Generation Task

I have to make an LSTM model understand the shakespeare language for generation purposes. I will write some words and the model will predict the next possible word

2 Methodology

2.1 Pre-Processing

1. Defined a function read all text from .txt file
2. Defined a function to split words by spaces and remove stop words in between them
3. Defined a tokenizer function to assign unique token to each word
4. Defined a function to create sequence of 5 words and their target to be 6th word
5. Defined a function to predict the model inference
6. Implemented a grid search function to find the best hyper parameter

2.2 Model Architecture

1. 512 LSTM units stacked
2. Dense layer with 128 neurons
3. Dense layer with 256 neurons
4. Dense layer with 512 neurons
5. Final dense layer with number of neurons equal to number of classes and softmax activation function.

2.3 Results

1. On 20000 samples I achieved these results on LSTM and simple RNN model
2. LSTM accuracy: 84.943. RNN accuracy: 34.58

On sentence: "Was this an answer to" LSTM result: ['unusual', 'raught', 'miseries', 'employ'] RNN result: ['enter', 'sergeant', 'enter', 'shrewsbury']

2.4 Discussion

1. LSTM was able to capture more relationship than RNN
2. LSTM showed diversity in generated samples
3. Model starts to converge rapidly after 25 epochs