# Paper: *RNN Language Models*


## Razwan Ahmed Tanvir


Date: 02/10/2022

Quote *"A LSTM is one of major recurrent neural net modules. It is designed for remembering the long-term memory, so that it should be able to consider relationships of distant words, such that a word at beginning of sentence and it at the end."*

Overview A language model is a model that generates sentences or documents. With this model we try to generate sentences that are meaningful or how close the sentences to the natural language. Language models estimate the probabilities of a word given it's previous words. Recurrent Neural Net Language Model (RNNLM) contains RNN in its network and it can work with the variable length input which is perfect for sentences as input. There are four basic steps to get the prediction of next word for a given sentence. These are, a) Getting the embedding vector, b) calculating the hidden layer values, c) calculating output layer and d) transforming the probability using softmax which coverts a real vector to a probability vector whose summation is overall one. Now, the evaluation of these models can be done using several methods but currently Perplexity is the common metric to evaluate such models. Perplexity measures how accurate the probabilistic model to generate target sentence. Perplexity ensures that the entropy is high. The smaller the perplexity, the better the model. LSTM is one of the major RNN models which remembers long term memory and can also incorporate context by establishing relationships with distant words. In this tutorial, the model is trained with the Penn Tree Bank Long Word Sequence Dataset.

This model generates sentences as output with a word as it's input. As an xample given in this tutorial, if we input the word 'apple' into the model, it generates- "apple a new u.s. economist with ¡unk¿ ¡unk¿ fixed more than to N the company said who is looking back to.". The generated sentence is not that meaningful.

Intellectual Merit *In this tutorial, the authors implemented a Recurrent Neural Net Language Model with Long Short Term Memory (LSTM). The author used linear transformation and to prevent the model from over-fitting, they used Dropout as regularization technique. Dropout method ensures that the model is not governed by some of the neurons. These methods of improving the model shows that the implementation is well organized and managed. Moreover, they used the GradientClipping to prevent gradient explosion. This tutorial uses the Penn Tree Bank Long Word Sequence Dataset to train their model.*

Broader Impact This tutorial helps to understand the RNNLM with LSTM to generate natural sentences. Also, we can see different techniques to improve the model output like using regularization, optimizer and eval_hook along with the effective evaluation metric named Perplexity. The authors published their implementation in a github repository and it is public. The Penn Tree Bank Long Word Sequence Dataset is also public and available for use. Anyone can replicate the implementation to better understand how this model works.

Keywords Natural Language Processing, Recurrent Neural Net Language Model, Perplexity

Discussion Questions

- The generated sentence after the training of the model does not have any meaning which is unexpected, authors should have evaluated the output sentence and the reason behind this output.
- I wonder how the different datasets could impact the accuracy of the model. Like, is it possible to establish some relationship with certain type of data and the trained model with that data.

Table 1: Grade deductions by section

| Overview | Intellectual M. | B. Impact | Keywords | Questions | Is Online? |
|---|---|---|---|---|---|
| | | | | | |