Project Proposal CSE702

horizontal line

Sequence to sequence learning with neural networks

**(Team: Payraw Salih, Parth Shah)**

**17th March 2019**

# Paper

The paper we are going to replicate is Sequence to Sequence Learning with Neural Networks, Sutskever et al. NIPS, 2014 (I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence to sequence learning with neural networks,” in Proceedings of Advances in Neural Information Processing Systems, 2014, pp. 3104–3112.)

The paper can be found at - <http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf>

# Why This Paper?

During the course of the seminar, we've have had multiple discussions regarding the translations provided by Machine Learning models

1. The paper tackles what must be one of the sternest tests of all when it comes to assessing how well the meaning of a sentence has been understood: machine translation.
2. The paper presents a general end-to-end approach to sequence learning that makes minimal assumptions on the sequence structure.
3. The overall process takes a sequence of (English word) tokens as input and produces another sequence of (French/German word) tokens as output. Thus the technique is called sequence-to-sequence learning.

We will try to replicate the same and produce a similar model which performs as good as the model described in the paper. Using bleu scoring system as a metric as described in this paper.

# 

# 

# Summary

1. Social Science Perspective

Human translation is slowly being aided by the use of machine learning alternatives. The current methods have a certain blind spot when it comes to the consideration of gender biases. We propose an LSTM RNN model in this project. LSTM has been known to be effective in taking a general problem of translating from one language to another, into an effective vector based approach. It creates such long-range dependencies such as the dependencies that are present between words in a sentence (gender form, subjects etc.). Thus we aim to provide with a model that can provide accurate translations and also overcomes the blind spots of modern translation models.

1. Computational Perspective

We plan on using a seq2seq model which consist of layers of LSTM cells stacked one over another working in a Recurrent Neural Network. An RNN takes the input of current data input as well as the previous state of the network into consideration when calculating outputs. One of the key technical contributions of the paper, as stated by the authors themselves, is when feeding the input sentence into the encoding LSTM, they discovered that the end-to-end process works much better if the sentence is input in reverse word order. Because of the introduction of many short term dependencies to the dataset.

# Details

## Data

The model was trained on the WMT ’14 English to French translation task and evaluated using the BLEU algorithm to produce a BLEU score. We plan on using the English to German translation task of WMT '16 for our model. The data was taken from the Europarl v7, Common Crawl, and News Commentary v11 corpora.

## Methods

We will create an LSTM Recurrent Neural Network Model divided in three parts Encoder, Attention and Decoder. All layers use BasicLSTMCell as LSTM Cell base available in Tesor Flow's RNN package.

## 

## Metrics

The paper uses BLEU Score as a metric. We will use the same as BLEU achieves a high correlation with human judgements of quality.

## Risks

Given the change in the dataset and the attempted upgrades as suggested, getting a similar BLEU score for our new model is the only risk. We will use SubWord unit which is developed for encoding rare words under the assumption that they're made up of smaller components. Useful in particular for German-English translation because of German's agglutinative word formation of noun lexemes. Subwords are encoded using Byte Pair Encoding and these subwords are used in the vocabulary.

## Limits

The authors built deep LSTMs with 4 layers of 1000 cells, and 1000-dimensional word embeddings, an input vocabulary of 160,000 words, and an output vocabulary of 80,000 words. For training, each layer resides on a separate GPU, with another 4 GPUs used to parallelize the softmax. This configuration achieved a speed of 6,300 words per second and training took about 10 days. Given the configuration available to us, replicating this on the same scale would take a lot more time and processing power. Thus replicating on a large scale is something we are still figuring out.

## Task Breakdown

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

## Stretch Goals

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.