

Name: Anantharam(Ananth) Subramanian

NLP HW #8

Evolution of Neural Networks in NLP

(2014 – Till Date)

Introduction: This document gives glimpse of how neural networks algorithms used in NLP has evolved since 2014.

Neural Networks in NLP:

2014: The late 2013 and early 2014 saw the starting of usage of Neural Networks within Natural Language Processing. The following algorithms started gaining popularity.

1. Recurrent Neural Networks (RNN)
2. Convolution Neural Networks (CNN)
3. Recursive Neural Networks

RNNs were thought to be difficult train and was quickly replaced by LSTM (Long Short-Term Memory).

2014 also found the usage of CNN in Language Processing and it operated in 2 dimensions. It was also found out that CNN can be combined with LSTM and it expected to speed up LSTM.

The year also saw emergence of Sequence to Sequence Learning. This is based on encoder-decoder model.

When we move from RNN to LSTM, we are introducing more & more controls, that controls the flow and mixing of Inputs as per trained Weights. This brings in more flexibility in controlling the outputs.

So, LSTM gives us more flexibility and hence more operating costs in terms of resources usage.

2015: The year 2015 saw “Attention” which is one of the best innovations among NLP algorithms. Attention makes up for some of the shortcomings of sequence to sequence algorithms. Attention is applicable where decisions are needed to be made based on certain parts of the input. One aspect of attention is that it provides a glimpse of the input that are relevant for a output based on the attention weights. Attention is seen as a fuzzy memory that consists of past hidden states and the algorithm decides what to choose from the memory. Some of the memory-based models include Neural Turing Machines, Memory Networks, End-to-End memory networks, Dynamic Memory Networks, Neural Differentiable Computer and Recurrent Entity Network.

The equation of attention comes at a cost. We need to calculate an attention value for each combination of input and output word. If the input sequence has 50 words and it generates a 50-word output sequence that would be 2500 attention values. But when we consider character level attention with sequences consisting of hundreds of tokens the above attention mechanisms can become prohibitively expensive.

2016 and Beyond:

Pre-trained models were developed during this era. Few examples include GloVe (Global Vectors for Word Representation), ELMO (Embeddings from Language Models), Universal Language Model Fine Tuning (ULMFiT), Universal Sentence Embedding and OpenAI Transformer.

GloVe, ELMO and Universal Sentence Embedder are word vector representations and can be used within a Neural Network Models.

They are helpful in Syntactic, Semantic analysis like POS (Part of Speech) tagging and Sentiment analysis.

One of the major inventions during this period is Transformer. Transformer concept plays a major role in identifying the relationship and sequences of words in a sentence.

The state-of-the-art NLP algorithm that was introduced in the late 2018 is BERT (Bidirectional Encoder Representations from Transformers). BERT's key differentiator is the application of applying bidirectional training of Transformer. Transformer includes two separate mechanisms — an encoder that reads the text input and a decoder that produces a prediction for the task. As BERT's generates a language model, only the encoder mechanism is necessary.

Every model discussed above varies in computational cost based on the output.

Many new things are getting introduced day by day and watching out the Data Science Journals will help us keep updated.

References:

1. <https://towardsdatascience.com/bert-explained-state-of-the-art-language-model-for-nlp-f8b21a9b6270>
2. <https://www.forbes.com/sites/forbestechcouncil/2018/11/06/the-evolution-of-natural-language-processing-and-its-impact-on-ai/#7071d2391119>
3. <https://runder.io/a-review-of-the-recent-history-of-nlp>
4. <http://www.wildml.com/2016/01/attention-and-memory-in-deep-learning-and-nlp/>
5. <https://medium.com/@saurabh.rathor092/simple-rnn-vs-gru-vs-lstm-difference-lies-in-more-flexible-control-5f33e07b1e57>