LSTM to Transformer to Star-Transformer

Logan Lebanoff, Chhaya Chouhan, Abhiditya Jha

Text classification is the prediction of which category a given text is. The creation of a perfect text classification model is at least NP-hard. Rather than attempt to create a perfect text classification model, we usually turn to creating approximate models, often using neural networks. In this work, we compare various models developed in this area and discuss their implications with regards to computational complexity. First, we will introduce the Long short-term Memory (LSTM) network, which processes inputs in an iterative manner, applying some operations on a persistent "hidden state" at every iteration. It runs in O(n) time, but on a GPU, it is really slow because of its iterative nature: it has to wait for the previous words to be processed before processing the current word. This does not take advantage of the parallelism of a GPU. Next, we will introduce the Transformer. It does away with the persistent hidden state, and therefore does not need to process inputs in an iterative manner. It makes full use of the GPU's parallelism. However, it runs in O(n^2), which is slower and takes much more memory, but the full utilization of the GPU makes it run faster than LSTM. Finally, we will introduce the Star-Transformer, which modifies the Transformer in such a way that the complexity reduces from quadratic to linear, making it more efficient while still making full use of the GPU. Through this study, we aim to understand how these different models work and finally, reflect upon the computational challenges and advances in neural networks.