Enhanced LSTM for Natural Language Inference

The paper "Enhanced LSTM for Natural Language Inference" by Qian Chen, Xiaodan Zhu, Zhenhua Ling, Si Wei, Hui Jiang, and Diana Inkpen, published in *Association for Computational Linguistics* in September 2016, present a new state-of-the-art result, achieving the accuracy of 88.6% on the Stanford Natural Language Inference Dataset. They first demonstrate that carefully designing sequential inference models based on chain LSTMs can outperform all previous models. Based on this, they further show that by explicitly considering recursive architectures in both local inference modeling and inference composition, they achieve additional improvement. Particularly, incorporating syntactic parsing information contributes to our best result — it further improves the performance even when added to the already very strong model.

Initially, the input sentences are tokenized and embedded into dense vector representations, typically using techniques like word embeddings or contextual embeddings. The embedded representations of the premise and hypothesis sentences are then fed into bidirectional LSTM (BiLSTM) layers so the model can learn to represent a word and its context. Then, a soft alignment layer is used to compute the attention weights to obtain the local relevance between a premise and hypothesis. For the hidden state of a word in a premise, the relevant semantics in the hypothesis is identified and composed using the attention weight. Then, to further sharpen local inference information between elements and capture inference relationships, the difference and the element-wise product for the tuples of BiLSTM result and weighted summation for relevant semantics are computed and concatenated with the original vectors. At last a composition layer is applied to determine the overall inference relationship between a premise and hypothesis. In this layer, BiLSTM is used to capture local inference information and context for local inference results. Then, both average and max polling are used and concatenated to form the final fixed length vector. The vector is then put into a final multuplayer perceptron classifer, which has a hidden layer with tanh activation and softmax output layer. The model is trained using multi-class cross entropy loss.

Through experiment, their ESIM model achieves an accuracy of 88.0%, which has already outperformed all the previous models, including those using much more complicated network architectures. They ensemble their ESIM model with syntactic tree-LSTMs based on syntactic parse trees and achieve significant improvement over our best sequential encoding model ESIM, attaining an accuracy of 88.6%. This shows that syntactic tree-LSTMs complement well with ESIM.