Guest Lecture 2: CS224n

Multitask Learning (Richard Socher)

author: luis0@stanford.edu
SUNet ID: 05794739

Richard Socher's presentation focused on the possible benefits of working on multi-task learning -- the idea of having a single model which can perform well in multiple NLP tasks. He believes this is the next natural step for AI -- deep architecture learning for multiple tasks.

According to Socher, single-task learning can now be considered "easy" -- generally speaking, given enough compute power and examples, a model is expected to perform well on a given supervised task. However, there's no knowledge sharing. Models are re-trained (starting from random), especially in NLP (isn't BERT and other embeddings a counter-example?). The proposal to address this problem is to two-pronged: (1) develop an architecture which is generic (this also requires expressing the problem correctly) and (2) define a multi-task metric that allows us to measure NLP performance across many tasks. Socher's solution to (1) is to make use of the question-and-answer model (similar to SQuAD) by casting non-QA problems, such as sentiment analysis being cast as a question ("Is this sentence positive or negative?"). This framing also somewhat leads to what a generalized model might look like. The proposed model consists of input Questions and Context, which are run through multiple alignment layers, co-attention layers, and (surprisingly!), bi-directional RNNs (multiple skip-connections, see the presentation for more details on the architecture). Socher made a comment about some difficulty arising from trying to use pure-attention-based architecture, though it wasn't very clear to me why this was the case. The model also decides between three types of outputs (which really seems a bit like cheating), consisting of (1) pointing to the context, (2) pointing to the question, or (3) a softmax over an external vocabulary. As for the second part, defining a metric, the proposed solution seem sort of hack. Essentially, ten currently-existing datasets with varying metrics (nF1, EM, etc.) were chosen to be part of the datasets. Their respective scores are scaled to 1-100 and summed, with no weighing.

It was interesting to learn about the different training strategies. For example, Socher found that anti-curriculum learning (hard tasks first, followed by easier tasks) actually helped the most with the training of the large model (even though it still did not achieve the theoretical maximum). They also had to weak it so that the translation tasks (which is the most distinct from the others) is trained on every batch of training (continuously). This allowed the larged model to improve significantly. It is quite interesting to see how the training order can affect the final performance of the model.

The most promising aspect of this model is its ability to zero-shot learn new tasks.