Generative AI 3

Eckel, TJHSST AI2, Spring 2024

Background & Explanation

There's one last thing I want to include in this unit and I haven't been able to take the time to figure out how it would work and if it is any help. Maybe you can!

Another Twist on an RNN

So far, we've looked at RNNs where we only consider the output at the final step. But actually, since we're predicting what comes next, we could feed in the first value… and see how well it predicts the second. Then, keeping the network state, feed in the second… and see how well the first two predict the third. Etc. In this way, we could take a 101-character string and make from it 100 different steps, each of which takes a single new input and, combined with the carry over from the previous step, predicts a single new output. 100 different predictions.

There is an offhand comment in the textbook I read that we can actually use *all 100 predictions* to generate error values, and back propagate all 100 back through our network at every training step.

I tried to do this once and it exploded; absolutely useless results. But that was before I learned about softmax / CCE and the better initialization radius, so who knows what went wrong. Maybe it was even totally fine and I just picked a bad learning rate. No idea.

Your Task

So your task is simply this – get this to work on your text prediction machine, and see if you can get it to train faster than the version where we only propagate back error from one prediction at each step. In order to receive credit, it has to *work* – like, you still need to show me that it is improving in some way as the epochs go by – but it doesn't necessarily need to work *better than your Generative AI 1/2 network*. I'm quite curious if we can get it to be better, but I accept failed experiments, as long as they at least function acceptably.