Why Don't We Use GANs for Sequences?

In Chapter 6, we discussed the problem of generating new images. We discussed models such as variational autoencoders that produced only blurry images and introduced the technology of generative adversarial networks that proves capable of producing sharp images. The question remains, though: if we need GANs for good image samples, why don't we use them for good sentences?

It turns out that today's generative adversarial models are mediocre at sampling sequences. It's not clear why this is the case. Theoretical understanding of GANs remains very weak (even by the standards of deep learning theory), but something about the game theoretic equilibrium discovery seems to perform worse for sequences than for images.

Seq2seq Models

Sequence-to-sequence (seq2seq) models are powerful tools that enable models to transform one sequence into another. The core idea of a sequence-to-sequence model is to use an encoding recurrent network that embeds input sequences into vector spaces alongside a decoding network that enables sampling of output sequences as described in previous sentences. Figure 7-6 illustrates a seq2seq model.

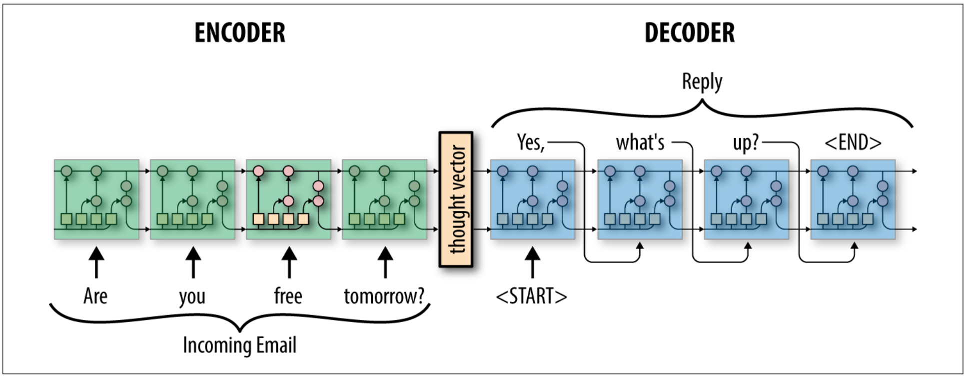

Figure 7-6. Sequence-to-sequence models are powerful tools that can learn sequence transformations. They have been applied to machine translation (for example, transforming a sequence of English words to Mandarin) and chemical retrosynthesis (transforming a sequence of chemical products into a sequence of reactants).

Things get interesting since encoder and decoder layers can themselves be deep. (RNN layers can be stacked in a natural fashion.) The Google neural machine translation (GNMT) system has many stacked encoding and decoding layers. As a result of this powerful representational capacity, it is capable of performing state-of-the-art translations far beyond the capabilities of its nearest nondeep competitors. Figure 7-7 illustrates the GNMT architecture.

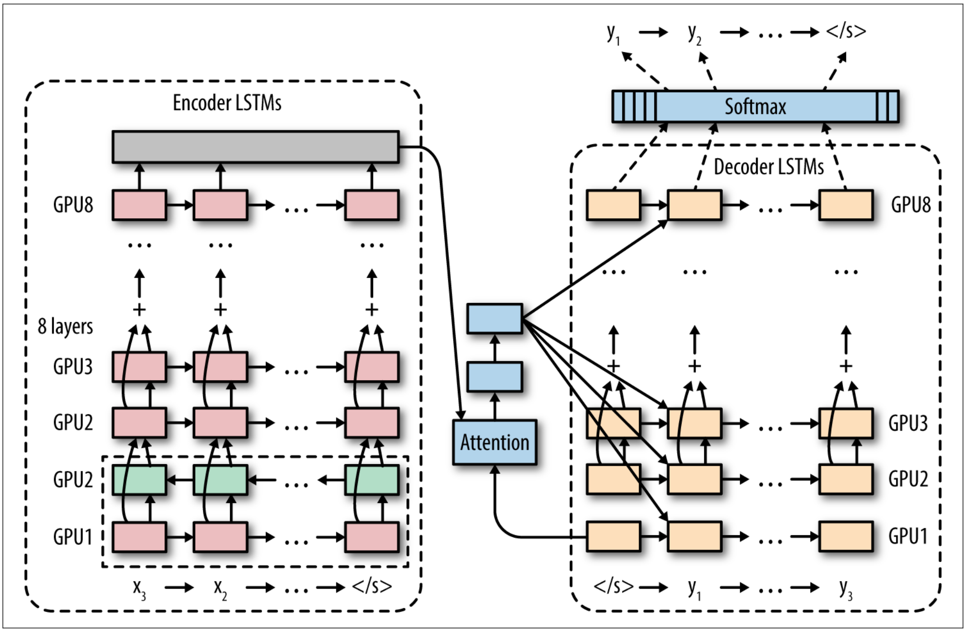

Figure 7-7. The Google neural machine translation (GNMT) architecture is a deep seq2seq model that learns to perform machine translation.

While so far we've mainly discussed applications to natural language processing, the seq2seq architecture has myriad applications in other domains. One of the authors has used seq2seq architectures to perform chemical retrosynthesis, the act of deconstructing molecules into simpler constituents. Figure 7-8 illustrates.

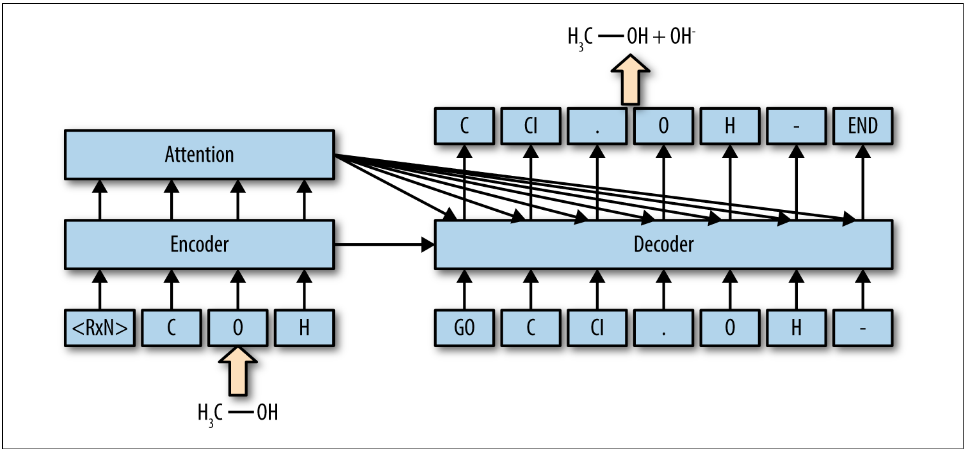

Figure 7-8. A seq2seq model for chemical retrosynthesis transforms a sequence of chemical products into a sequence of chemical reactants.

Neural Turing Machines

The dream of machine learning has been to move further up the abstraction stack: moving from learning short pattern-matching engines to learning to perform arbitrary computations. The Neural Turing machine is a powerful step in this evolution.

The Turing machine was a seminal contribution to the mathematical theory of computation. It was the first mathematical model of a machine capable of performing any computation. The Turing machine maintains a "tape" that provides a memory of the performed computation. The second part of the machine is a "head" that performs transformations on single tape cells. The insight of the Turing machine was that the "head" didn't need to be very complicated in order to perform arbitrarily complicated calculations.

The Neural Turing machine (NTM) is a very clever attempt to transmute a Turing machine itself into a neural network. The trick in this transmutation is to turn discrete actions into soft continuous functions (this is a trick that pops up in deep learning repeatedly, so take note!)

The Turing machine head is quite similar to the RNN cell! As a result, the NTM can be trained end-to-end to learn to perform arbitrary computations, in principle at least (Figure 7-9). In practice, there are severe limitations to the set of computations that the NTM can perform. Gradient flow instabilities (as always) limit what can be learned. More research and experimentation will be needed to devise successors to NTMs capable of learning more useful functions.