Gated Recurrent Units (GRU)

The complexity, both conceptual and computational, for LSTM cells has motivated a number of researchers to attempt to simplify the LSTM equations while retaining the performance gains and modeling capabilities of the original equations.

There are a number of contenders for LSTM replacement, but one of the frontrunners is the gated recurrent unit (GRU), shown in Figure 7-5. The GRU removes one of the subcomponents of the LSTM but empirically seems to achieve performance similar to that of the LSTM. The GRU might be a suitable replacement for LSTM cells on sequence modeling projects.

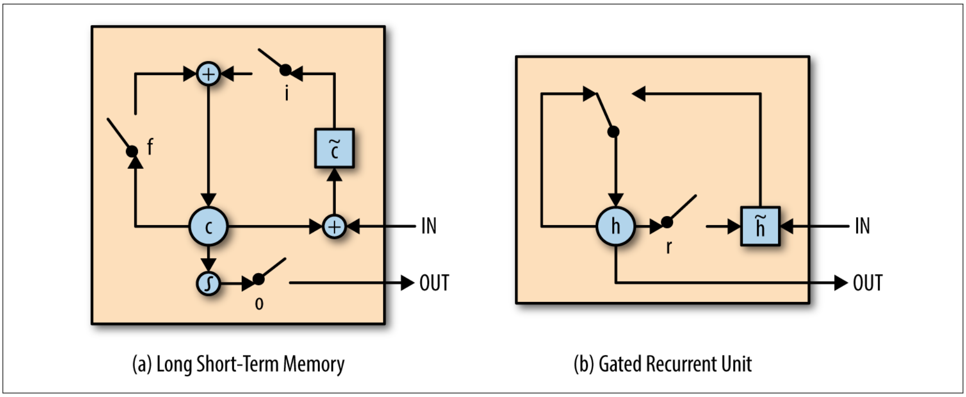

Figure 7-5. A gated recurrent unit (GRU) cell. GRUs preserve many of the benefits of LSTMs at lower computational cost.

Applications of Recurrent Models

While recurrent neural networks are useful tools for modeling time-series datasets, there are a host of other applications of recurrent networks. These include applications such as natural language modeling, machine translation, chemical retrosynthesis, and arbitrary computation with Neural Turing machines. In this section, we provide a brief tour of some of these exciting applications.

Sampling from Recurrent Networks

So far, we've taught you how recurrent networks can learn to model the time evolution of sequences of data. It stands to reason that if you understand the evolution rule for a set of sequences, you ought to be able to sample new sequences from the distribution of training sequences. And indeed, it turns out that that good sequences can be sampled from trained models. The most useful application thus far is in language modeling. Being able to generate realistic sentences is a very useful tool that underpins systems such as autocomplete and chatbots.

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.

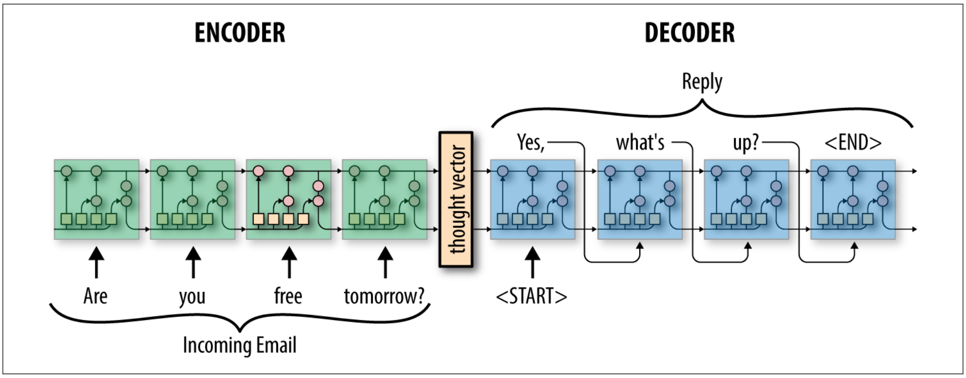

Figure 7-6. Sequence-to-sequence models are powerful tools that can learn sequence transformations. They have been applied to machine translation (for example, transforming a sequence of English words to Mandarin) and chemical retrosynthesis (transforming a sequence of chemical products into a sequence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