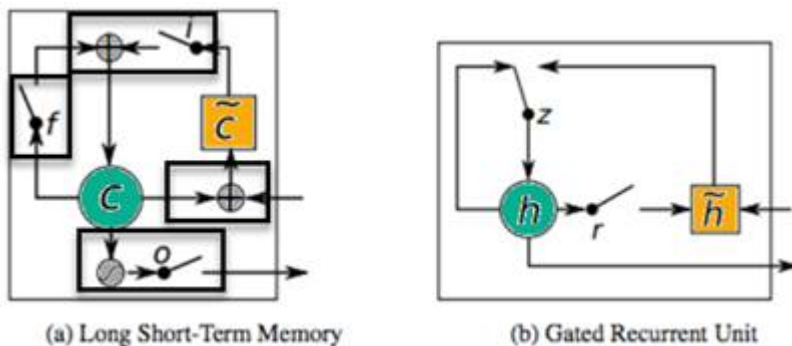# Differences between LSTM and GRU

There are a few subtle differences between a LSTM and a GRU, although to be perfectly honest, there are more similarities than differences! For starters, a GRU has one less gate than an LSTM. As you can see in the following diagram, an LSTM has an input gate, a forget gate, and an output gate. A GRU, on the other hand, has only two gates, a reset gate and an update gate. The reset gate determines how to combine new inputs with the previous memory, and the update gate defines how much of the previous memory remains:

(a) Long Short-Term Memory   (b) Gated Recurrent Unit

LSTM vs GRU

Another interesting fact is that if we set the reset gate to all 1s and the update gate to all 0s, do you know what we have? If you guessed a plain old recurrent neural network, you'd be right!

Here are the key differences between a LSTM and a GRU:

> A GRU has two gates, a LSTM has three.

> GRUs do not have an internal memory cell that is different from the exposed hidden state. This is because the output gate that the LSTM has does.

> The input and forget gates are coupled by an update gate that weighs the old and new content.

> The reset gate is applied directly to the previous hidden state.

> ❯ We do not apply a second non-linearity when computing the GRU output.

> ❯ There is no output gate, the weighted sum is what becomes the output.

Using a GRU versus a LSTM

Since GRUs are relatively new, the question usually arises as to which network type to use and when. To be honest, there really is no clear winner here, as GRUs have yet to mature, and LSTM variants seem to pop up every month. GRUs do have fewer parameters, and theoretically may train a bit faster than a LSTM. It may also theoretically need less data than a LSTM. On the other hand, if you have a lot of data, the extra power of the LSTM may work better for you.