RNNs, LSTMs, and Reddit Comments

Ivan Aguilar, Brandon McKinzie
University of California, Berkeley

Recurrent Neural Networks

A recurrent neural network(RNN) is the natural architecture to work with sequential data which depend on previous inputs. With a standard network, there is no way for the network to remember previous inputs, but in a recurrent network information is passed from one time step to the next through the hidden layer. This is visually shown in Figure 1 below.

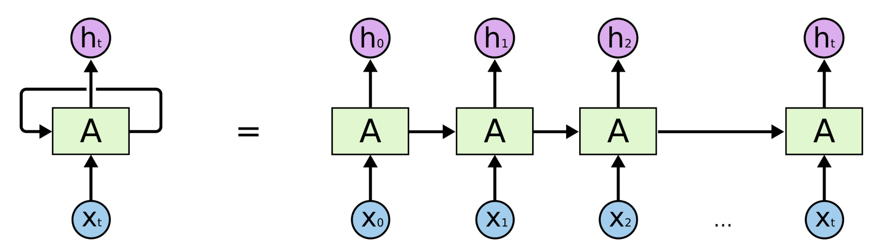

Figure 1: A visualization of an RNN which has been rolled out to explicitly show how information propagates from previous states. Image source: ref. [1].

In a Vanilla RNN, we have three weight matrices.U, V, and W. U maps our input to the hidden space, V maps the hidden state to the output state, and W is what makes our network recurrent, connecting the hidden state to itself. The forward propagation of the network is defined by two non-linear functions, f and g.

$$s(t) = f(Ux(t) + Ws(t-1)) \quad (1)$$
$$o(t) = g(Vs(t)) \quad (2)$$

$s(t)$ is the hidden state at time t, where a unit of time corresponds to a input to the network. $o(t)$ is the output of the network at time t. We would like to note that $s(t)$, depends on $s(t-1)$.

Training an RNN

When dealing with a recurrent network the back-prop algorithm does not immediately make sense due to the feedback connections within the hidden layer. By looking at the network in the rolled out state of Figure 1, we can update back propagation to also go back in *time*.

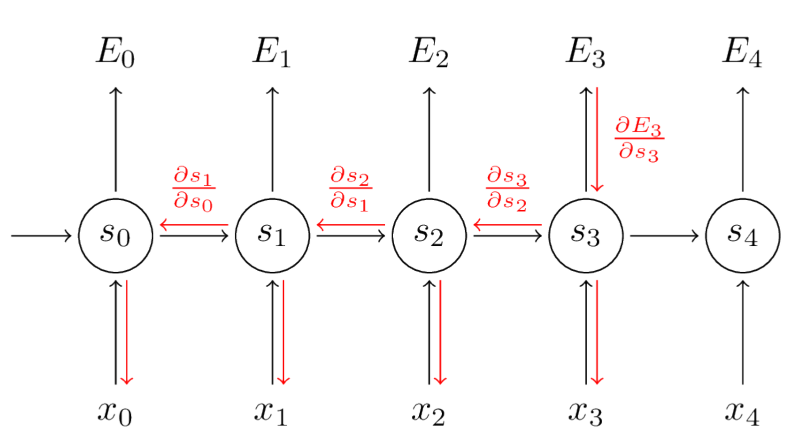

Figure 2: Propagation of error through layers and time. E_t depends on the hidden state s_t, and input x_t. s_t depends on the previous state and previous input to that state. Image source: ref. [2].

Long Short Term Memory

Vanilla RNN's have two troubling issues, vanishing/exploding gradients and the inability to preserve long term dependencies. The LSTM solves these two problems.

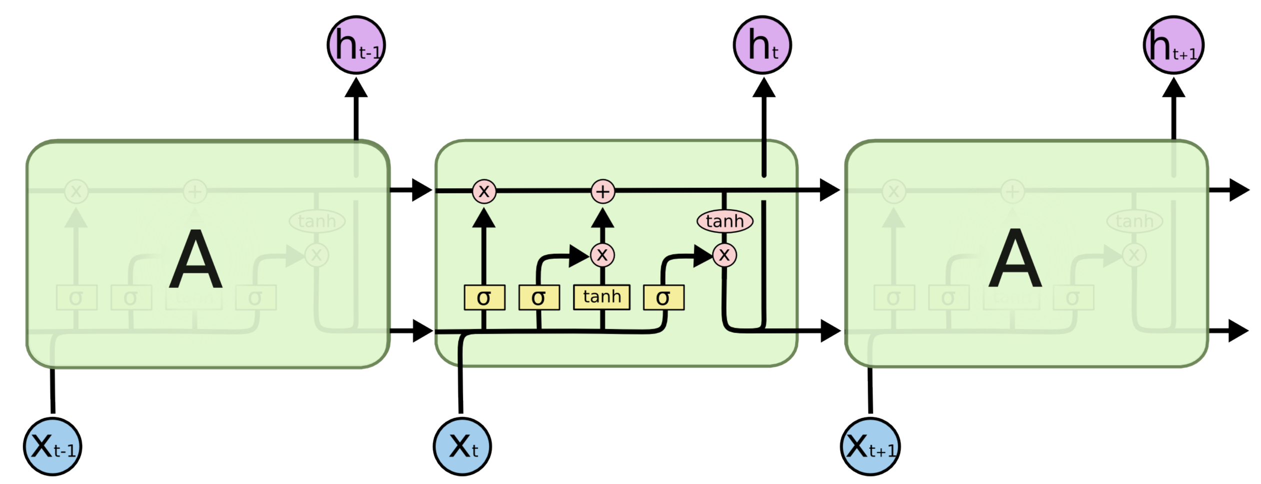

Figure 3: Three time steps of an LSTM with a cell highlighting all of the internal components. Image source: ref. [1].

The LSTM network is best described by gates which determine how information propagates through the network via a cell state.

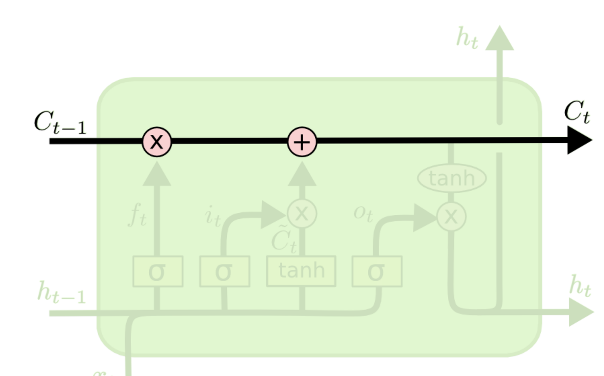

Figure 4: The cell state of the network and the gates labeled. Image source: ref. [1].

1. The "forget" gate, $f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$, determines what portion of our cell state to keep.
2. The "input" gate, $i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$ and $\widetilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_C)$ determines how to update the cell state given the input.
3. The "output" gate, $o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$ and $h_t = o_t \star \tanh(C_t)$ determines what the output of the network will be.

Training an LSTM

All of the gates described above have their own parameters which need to be optimized. The training procedure is still back propagation through time, like with the Vanilla RNN, but due to the complexity of an LSTM Cell training time is increased. To minimize the error, the chain rule takes us through previous time steps, but the process of getting there is muddled by the additional gates mentioned above.

Generated Reddit Comments

Below, we show comments generated from our RNN and LSTM, both written from scratch in NumPy. We also include comments generated from a simple implementation with TFLearn. These models were all trained for less than an hour on 20,000 Reddit comments.

RNN Comments

1. Many should still especially as me in nothing as the fight combat are favorite of your lot.
2. It should know arguing really pounds and much.
3. If you can both used as the fact [expletive] is better or kind and quickly with the [expletive] maybe i did already know.

LSTM Comments

1. People cash everything and christian the to dust or the glorious.
2. Education time is empty ps4.
3. Do riot vocal lol.

TFLearn Comments

1. Provide some flexibity and a good.
2. It sounds the sure and stread and of the fart to a some of the part.
3. In games journalism that do the press and the lan meaning.

Conclusions

We've shown how simple implementations of RNNs and LSTMs can produce reasonably structured sentences after surprisingly short training times. A further application of the LSTM model includes the attention based encoder-decoder which is used in image captioning, and machine translation. For example, one can decompile images of LaTeXto source code with a network consisting of convolution layers followed by recurrent layers (e.g. LSTM layers).

References

[1] Olah, C. (2015, August 27). Understanding LSTM Networks. Retrieved December 3, 2016, from http://colah.github.io/posts/2015-08-Understanding-LSTMs/

[2] Britz, D. (2015, October 8). Recurrent Neural Networks Tutorial, Part 3 - Backpropagation Through Time and Vanishing Gradients. Retrieved December 4, 2016, from http://www.wildml.com/2015/10/recurrent-neural-networks-tutorial-part-3-backpropagation-through-time-and-vanishing-gradients/