Recurrent Neural Networks - Introduction

ENGG 192, Dartmouth College
Spencer Bertsch
Winter 2019

This notebook discusses the basics of recurrent neural networks. Next week I will build on this foundation and uses the PyTorch library to construct a simple, one layer RNN and more complex networks which use long short term memory units. Before constructing more complex RNNs using APIs such as Keras, it's important to understand the intracacies of the nodes and understand the math behind each function.

Sources:

While learning about RNNs, I read several helpful online articles and watches several videos from professors and professionals in the field. Sources are listed below.

Medium.com - Building RNNs is Fun with PyTorch and Google Colab

Illustrated Guide to LSTM's and GRU's: A step by step explanation

Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM)

A friendly introduction to Recurrent Neural Networks

A Simple Neural Network from Scratch with PyTorch and Google Colab

Time Sequence Prediction - Sine Wave

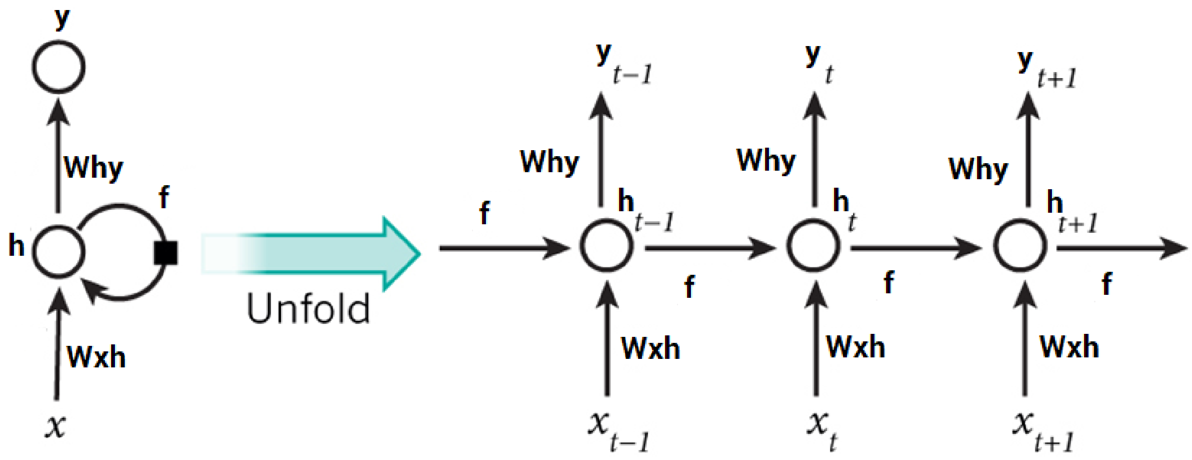
Fundamentals of Deep Learning – Introduction to Recurrent Neural Networks

[3]

```python
#Imports
import IPython as IP
```

[6]

```python
IP.display.Image('https://s3-ap-south-1.amazonaws.com/av-blog-media/wp-co
```



The diagram above is sourced from Analytics Vidhya and shows that recurrent neural networks by introducing the aspect of learning from time into each node. This allows networks to learn not only from the variety of input features in the data, but also by 'remembering' the trends on the data through time. This is crucial when the patterns that we're trying to map aren't simply a function of the input features, but rather a time sequence which repeats or evolves through time.

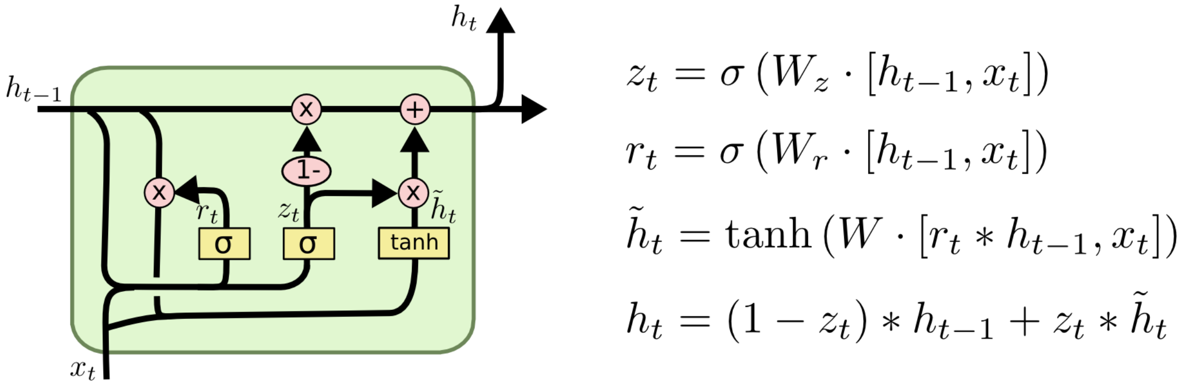
Here we see a single node h and a function $f(t)$ used to calculate the current state of h.

Our neuron doesnt simply take the input feature $x(t)$, it also takes $h(t-1)$ as an input. This gives us $h(t) = f(h(t-1), x(t))$ where $h(t-1)$ is the state of the neuron at the previous time step.

Long Short Term Memory (LSTM)

Watch video again ans explain

```
[25]    IP.display.Image('http://colah.github.io/posts/2015-08-Understanding-LSTM
```


h_t
h_{t-1}
x
+
1-
x
x
r_t
z_t
$\tilde{h}_t$
σ
σ
tanh
x_t
$z_t = \sigma\left(W_z \cdot [h_{t-1}, x_t]\right)$
$r_t = \sigma\left(W_r \cdot [h_{t-1}, x_t]\right)$
$\tilde{h}_t = \tanh\left(W \cdot [r_t * h_{t-1}, x_t]\right)$
$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t$

In order to take full advantage of the ability to learn from previous 'guesses' or previous outputs $h(t-1)$ the LSTM or long short term memory unit was conceived. This unit essentially acts in place of a single node in the network. It makes use of the tanh function and the sigmoid fucntion to learn the propper weights for the gates used to both remember and forget the information from previous time steps.

This is helpful becasue it means that as the network trains, it will learn what it needs to remember, and what it needs to forget in order to make the correct output predictions. However, becasue the previous network predictions are added to the inputs, a selection gate is added to ensure that only the predictions and not simply the memories are allowed through to the end (as true predictions).

Long short term memory is a complicated subject, but I will be exploring it in much more detail in the upcoming week.

Creating a Sine Wave

This will be left as one of the exercises for this upcoming week. Using a sine wave as the input for a recurrent neural network, especially an LSTM network, is a great use case becasue the prediction directly relies on the time series, not on any categorical or spacial data.

Attention Mechanisms

In addition to recurrent neural networks, attention mechanisms are used in sequence to sequence applicatin such a translation. Articles such as this one from Data Camp discuss how the use of attention mechanisms may overtake the capabilities of recurrent neural networks which make use of LSTM units.

I will also go into more depth next week when discussing attention mechanisms, and when to use them in conjunction or in place of recurrent neural networks.