CHAPTER 36

Recurrent Neural Networks (RNNs)

Recurrent neural networks (RNNs) are another specialized scheme of neural network architectures. RNNs are developed to solve learning problems where information about the past (i.e., past instants/events) is directly linked to making future predictions. Such sequential examples play up frequently in many real-world tasks such as language modeling where the previous words in the sentence are used to determine what the next word will be. Also in stock market prediction, the last hour/day/week stock prices define the future stock movement. RNNs are particularly tuned for time series or sequential tasks.

In a sequential problem, there is a looping or feedback framework that connects the output of one sequence to the input of the next sequence. RNNs are ideal for processing 1-D sequential data, unlike the grid-like 2-D image data in convolutional neural networks.

This feedback framework enables the network to incorporate information from past sequences or from time-dependent datasets when making a prediction.
In this section, we will cover the broad conceptual overview of recurrent neural networks and in particular the Long Short-Term Memory RNN variant (LSTM) which is the state-of-the-art technique for various sequential problems such as image captioning, stock market prediction, machine translation, and text classification.

The Recurrent Neuron

The first building block of the RNN is the recurrent neuron (see Figure 36-1). The neurons of the recurrent network are entirely different from those of other neural network architectures. The key difference here is that the recurrent neuron maintains a memory or a state from past computations. It does this by taking as input the output of the previous instant y_{t-1} in addition to its current input at a particular instant x_t.

E. Bisong, *Building Machine Learning and Deep Learning Models on Google Cloud Platform*,
https://doi.org/10.1007/978-1-4842-4470-8_36

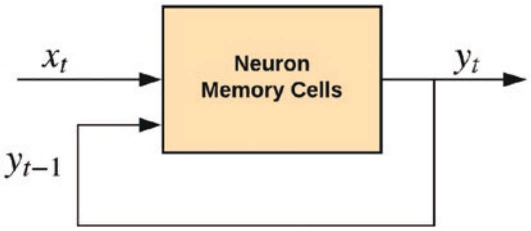

Figure 36-1. *A recurrent neuron*

In Figure 36-1, the recurrent neuron stands in contrast with neurons of the MLP and CNN architectures because instead of transferring a hierarchy of information across the network from one neuron to the other, data is looped back into the same neuron at every new time instant. A time instant can also mean a new sequence.

Hence, the recurrent neuron has two input weights, W_{x_t} and $W_{y_{t-1}}$, for the input at time x_t and for the input at time instant y_{t-1}. See Figure 36-2.

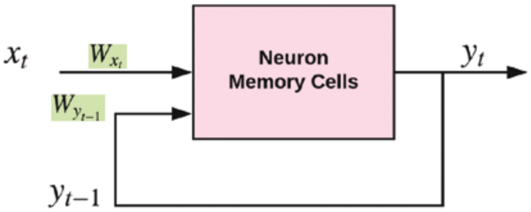

Figure 36-2. *Recurrent neuron with input weights*

Similar to other neurons, the recurrent neuron also injects non-linearity into the network by passing its weighted sums or affine transformations through a non-linear activation function.

Unfolding the Recurrent Computational Graph

A recurrent neural network is formalized as an unfolded computational graph. An unfolded computational graph shows the flow of information through the recurrent layer at every time instant in the sequence. Suppose we have a sequence of five time steps, we will unfold the recurrent neuron five times across the number of instants.
The number of sequences constitutes the layers of the recurrent neural network architecture. See Figure 36-3.