



Figure 10.3: The computational graph to compute the training loss of a recurrent network that maps an input sequence of \mathbf{x} values to a corresponding sequence of output \mathbf{o} values. A loss L measures how far each \mathbf{o} is from the corresponding training target \mathbf{y} . When using softmax outputs, we assume \mathbf{o} is the unnormalized log probabilities. The loss L internally computes $\hat{\mathbf{y}} = \text{softmax}(\mathbf{o})$ and compares this to the target \mathbf{y} . The RNN has input to hidden connections parametrized by a weight matrix \mathbf{U} , hidden-to-hidden recurrent connections parametrized by a weight matrix \mathbf{W} , and hidden-to-output connections parametrized by a weight matrix \mathbf{V} . Equation 10.8 defines forward propagation in this model. (Left) The RNN and its loss drawn with recurrent connections. (Right) The same seen as an time-unfolded computational graph, where each node is now associated with one particular time instance.