IX. Deep Learning

- Elements of Deep Learning
- Convolutional Neural Networks
- Autoencoder Networks
- Recurrent Neural Networks
- Long-Term Dependencies
- RNNs for Machine Translation
- Attention Mechanism
- Self Attention and Transformers
- Transformer Language Models
Long-Term Dependencies

Notation II (computational graph)  [notation: color, graph, language]

\[
\begin{align*}
    y^h(0) & = W^h y^h(1) = W^h y^h(2) \\
    x_1 & \rightarrow \cdots \rightarrow \sum_j \rightarrow y^h(0) \\
    x_p & \rightarrow \cdots \rightarrow \sum_j \rightarrow y^h(1) \\
    x(1) & \rightarrow \cdots \rightarrow \sum_j \rightarrow y^h(2) \\
    x(T-1) & \rightarrow \cdots \rightarrow \sum_j \rightarrow y^h(T) \\
    x(T) & \rightarrow \cdots \rightarrow \sum_j \rightarrow y^h(T-1)
\end{align*}
\]

\[
\begin{align*}
    y^h(t) & \text{ hidden} \\
    y^e(t) & \text{ hidden encoder} \\
    y^d(t) & \text{ hidden decoder} \\
    x(t) & \text{ input} \\
    y(t) & \text{ output} \\
    c(t) & \text{ target} \\
    y^a(t) & \text{ attention} \\
    y^o & \text{ output} \\
    c_1 & \text{ target (training)} \\
\end{align*}
\]

\[
\begin{align*}
    y^h(t) & = W^h y^h(t-1) \\
    y^h(0) & = W^h y^h(1) = W^h y^h(2) \\
    x_1 & \rightarrow \cdots \rightarrow \sum_j \rightarrow y^h(0) \\
    x_p & \rightarrow \cdots \rightarrow \sum_j \rightarrow y^h(1) \\
    x(1) & \rightarrow \cdots \rightarrow \sum_j \rightarrow y^h(2) \\
    x(T-1) & \rightarrow \cdots \rightarrow \sum_j \rightarrow y^h(T) \\
    x(T) & \rightarrow \cdots \rightarrow \sum_j \rightarrow y^h(T-1)
\end{align*}
\]
Long-Term Dependencies

Notation II (computational graph)

[notation: color, graph, language]
Long-Term Dependencies

Notation II (computational graph)

[notation: color, graph, language]
Long-Term Dependencies

Notation II (computational graph) [notation: color, graph, language]
Remarks (computational graph):

- The computational graph notation shown here follows Goodfellow/Bengio/Courville 2016:
  1. Each node in the graph indicates a variable. A variable may be a scalar, vector, matrix, tensor, or be of another type.
  2. An operation is a function of one or more variables. An operation returns a single output variable, which does not lose generality because the output variable can have multiple entries, such as a vector.

If a variable $b$ is computed by applying an operation to a variable $a$, a directed edge is drawn from $a$ to $b$. 
Long-Term Dependencies

Vanishing Gradient Problem

[TODO]
Long-Term Dependencies
RNN with Long Short-Term Memory (LSTM)

[TODO]
LSTM is a recurrent neural network architecture that is very efficient at remembering long term dependencies and that is less vulnerable to the vanishing gradient problem.
Long-Term Dependencies
RNN with Gated Recurrent Units (GRU)

[TODO]