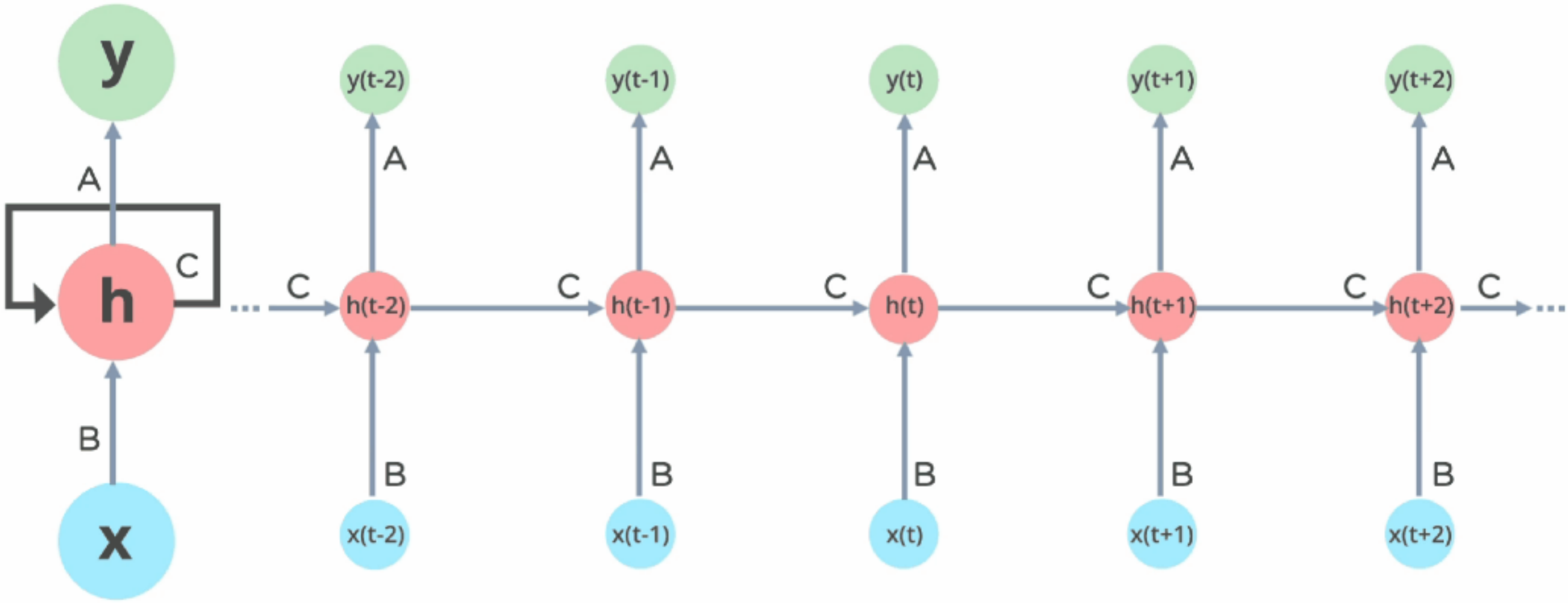
Recurrent connections

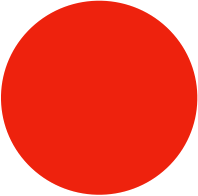


y
A
C
h
B
x
y(t-2)
y(t-1)
y(t)
y(t+1)
y(t+2)
h(t-2)
h(t-1)
h(t)
h(t+1)
h(t+2)
x(t-2)
x(t-1)
x(t)
x(t+1)
x(t+2)

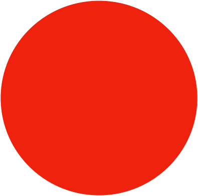
- Recurrent neural networks are based on the idea of an *infinite impulse response filter* (IRR), whereby the feature representation at the t^{th} sequence position, $\mathbf{h}^{(t)}$, is a function of both $\mathbf{x}^{(t)}$ and $\mathbf{h}^{(t-1)}$. This *hidden state* can then be used in a variety of of ways, for example in language modeling a word/token is predicted at each sequence step, whereas for a text classification task, only the feature layer at the last step, $\mathbf{h}^{(T)}$, is used to predict the output.

- In theory recurrent connections enable us to maximally capture context. In practice, training these networks becomes increasingly difficult for long sequences due to a phenomenon called vanishing gradients.

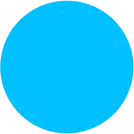
- The popular Long-Short Term Memory (LSTM) cell block is based on this idea!



$$h^{(t-1)}$$



$$h^{(t)}$$



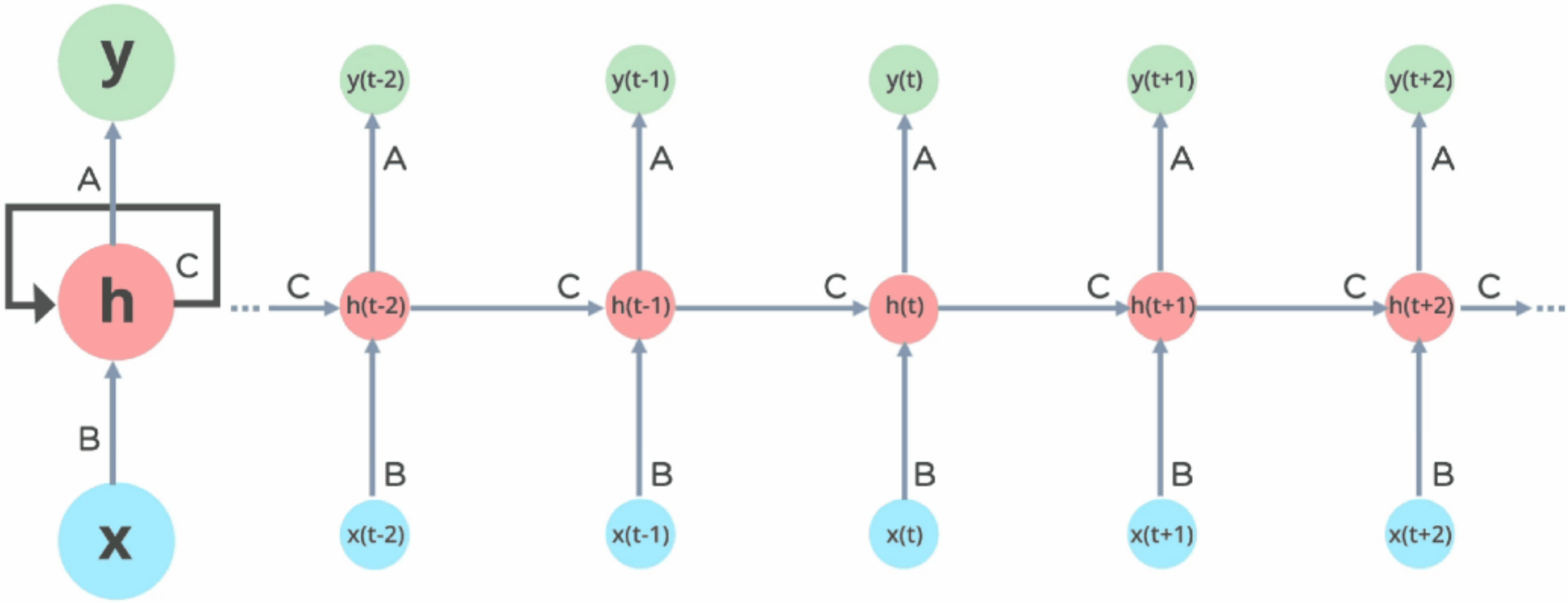
$$x^{(t)}$$

[redacted]

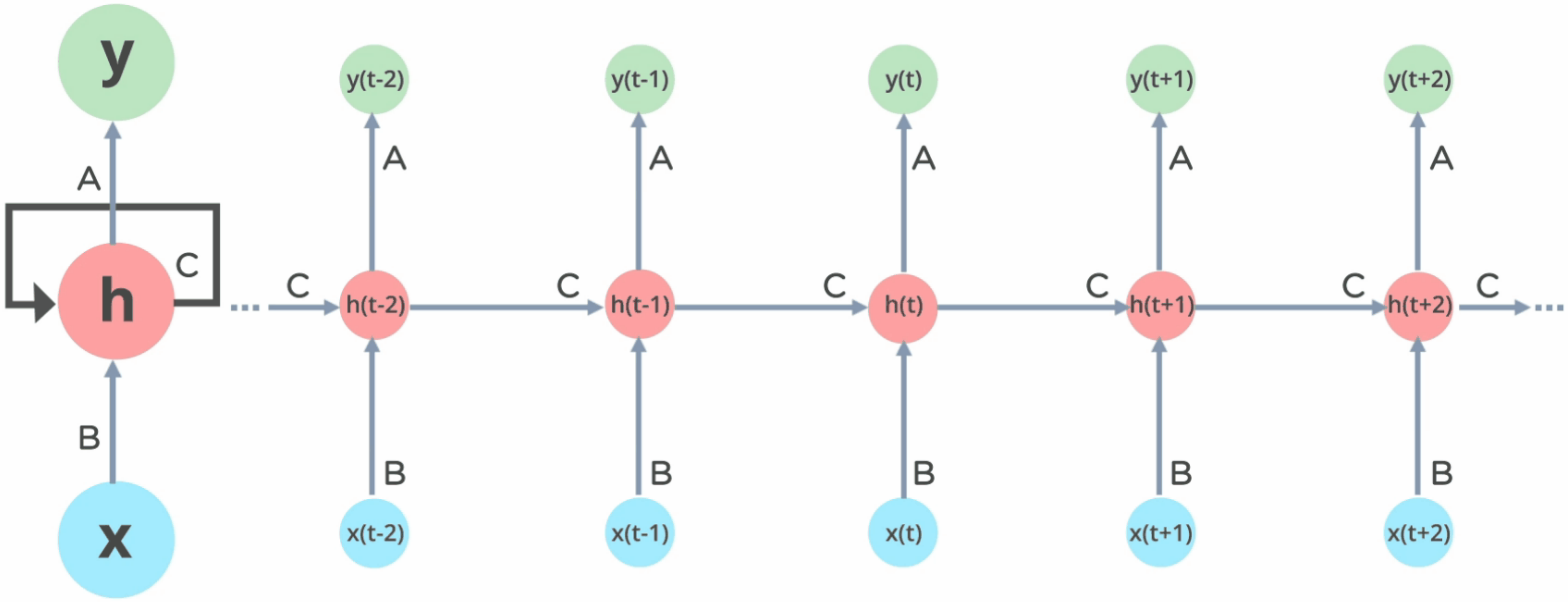
[redacted]

NN cell

Hidden state transition



y
A
C
h
B
x
y(t-2)
y(t-1)
y(t)
y(t+1)
y(t+2)
A
A
A
A
A
C
C
C
C
C
C
h(t-2)
h(t-1)
h(t)
h(t+1)
h(t+2)
B
B
B
B
B
x(t-2)
x(t-1)
x(t)
x(t+1)
x(t+2)

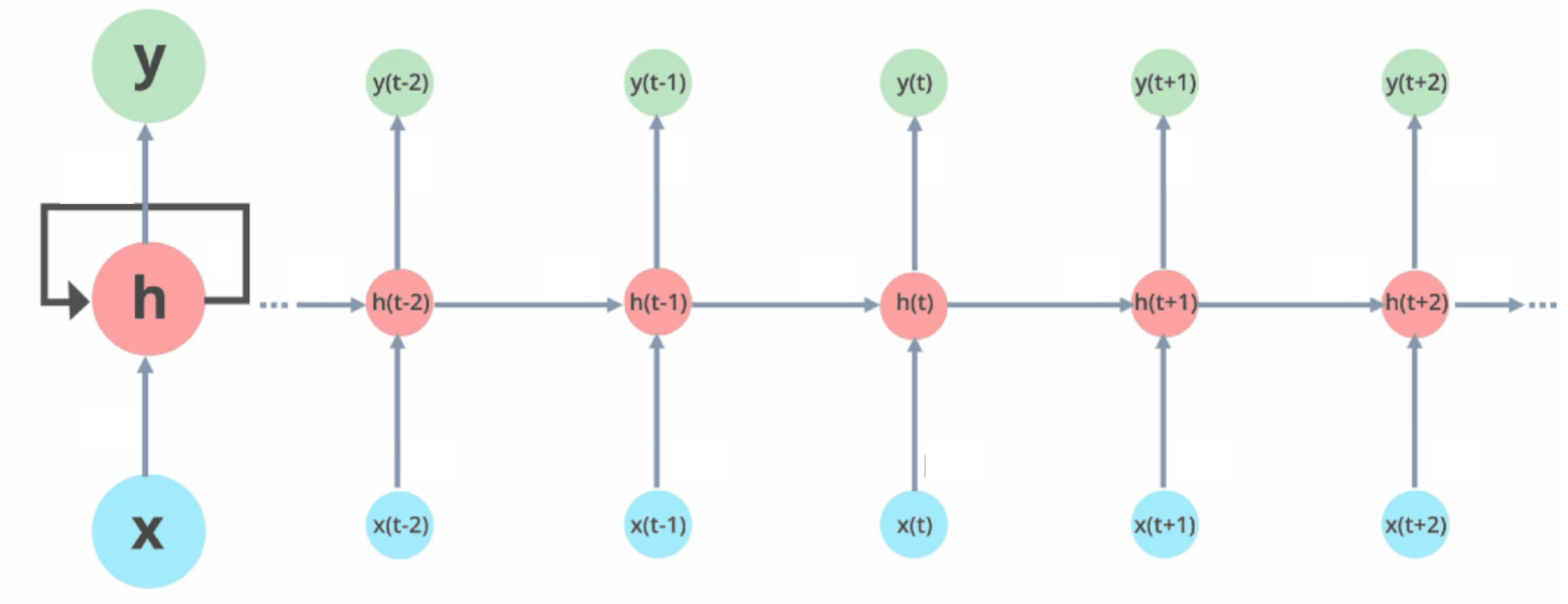

y
A
h
C
B
x
y(t-2)
y(t-1)
y(t)
y(t+1)
y(t+2)
A
A
A
A
A
C
C
C
C
C
C
h(t-2)
h(t-1)
h(t)
h(t+1)
h(t+2)
B
B
B
B
B
x(t-2)
x(t-1)
x(t)
x(t+1)
x(t+2)

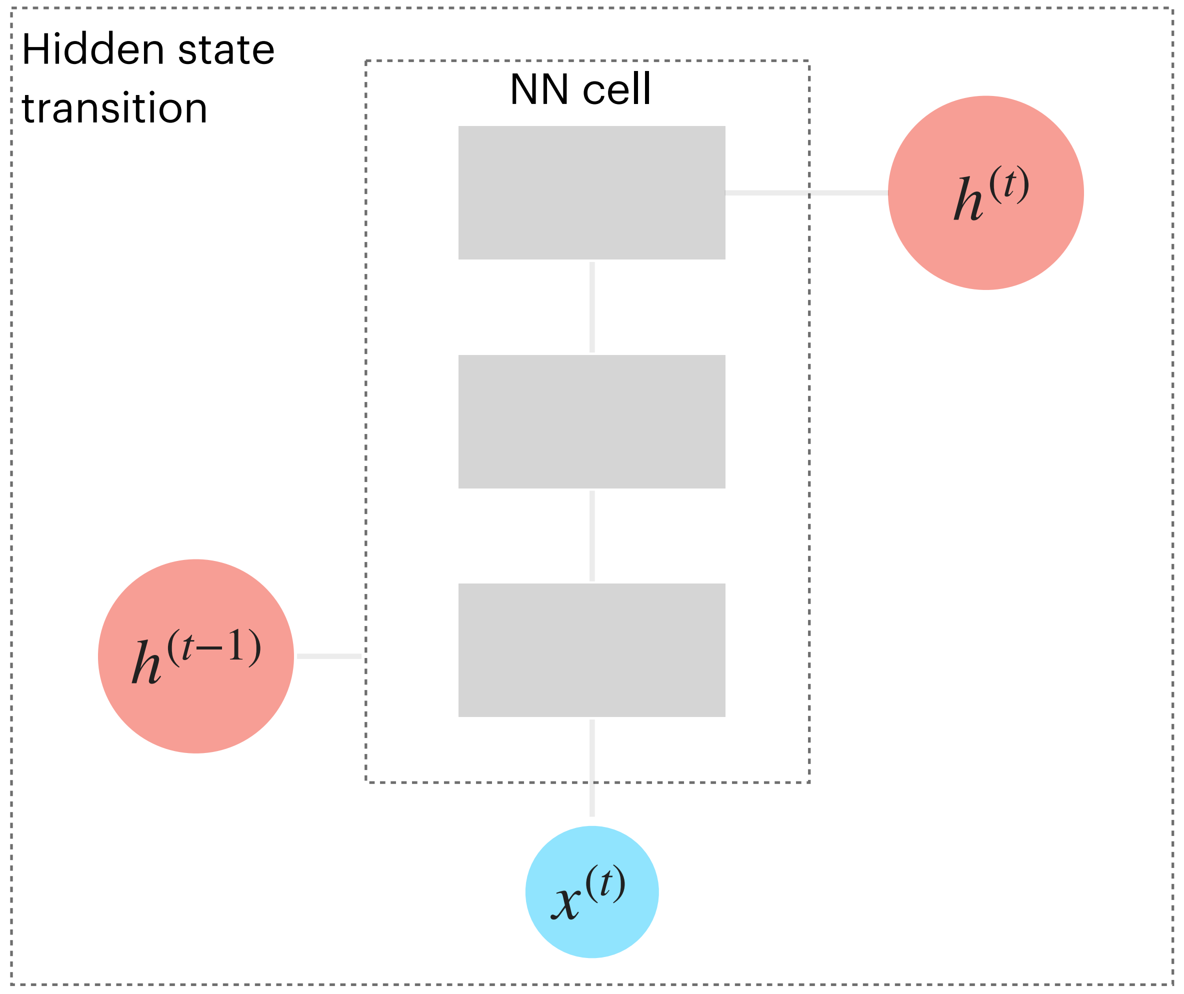
Recurrent connections

- Recurrent neural networks are based on the idea of an *infinite impulse response filter* (IRR), whereby the feature representation at the t^{th} sequence position, $\mathbf{h}^{(t)}$, is a function of both $\mathbf{x}^{(t)}$ and $\mathbf{h}^{(t-1)}$. This *hidden state* can then be used in a variety of of ways, for example in language modeling a word/token is predicted at each sequence step, whereas for a text classification task, only the feature layer at the last step, $\mathbf{h}^{(T)}$, is used to predict the output.

- In theory recurrent connections enable us to maximally capture context. In practice, training these networks becomes increasingly difficult for long sequences due to a phenomenon called vanishing gradients.

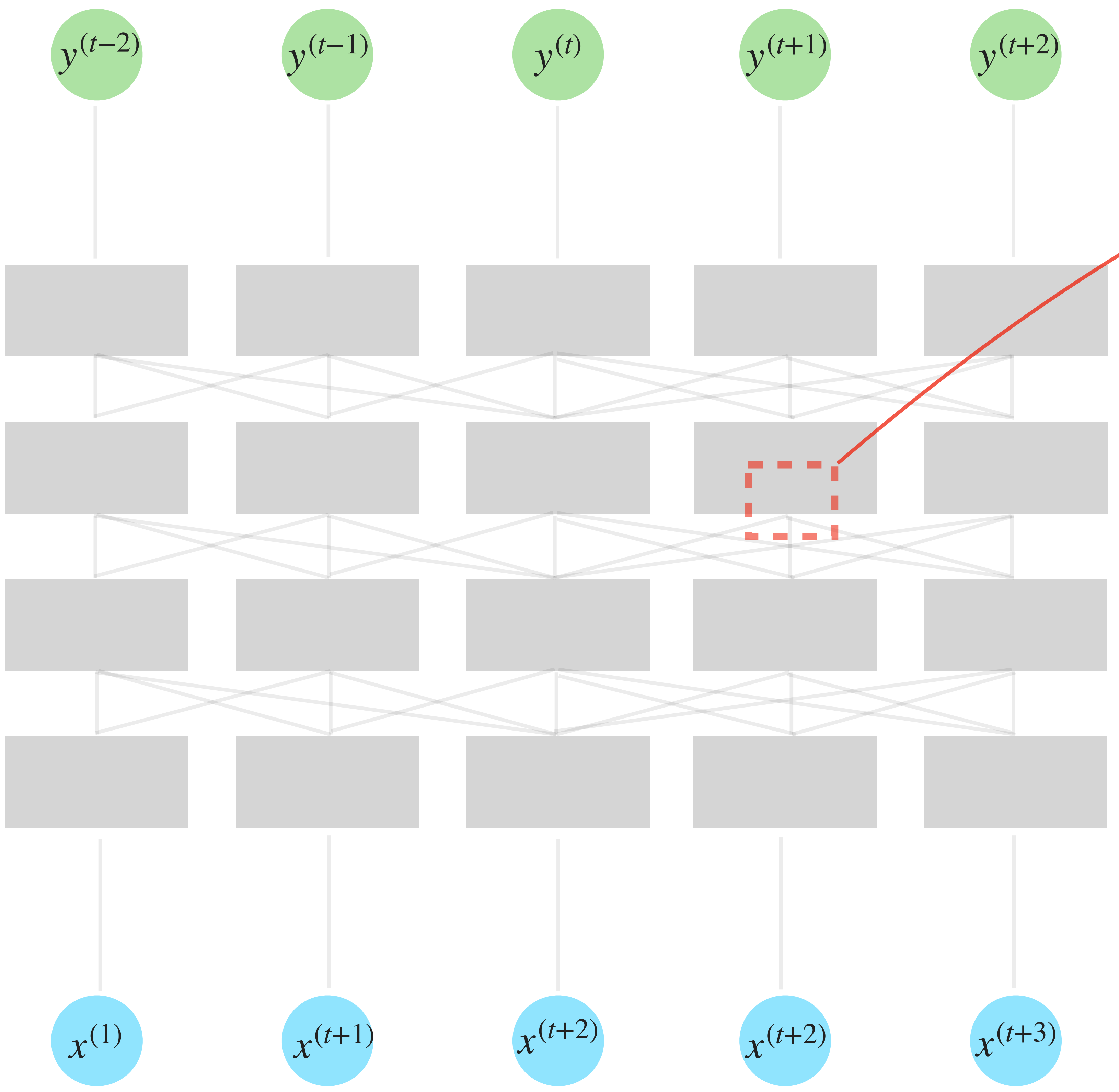
- The popular Long-Short Term Memory (LSTM) cell block is based on this idea!



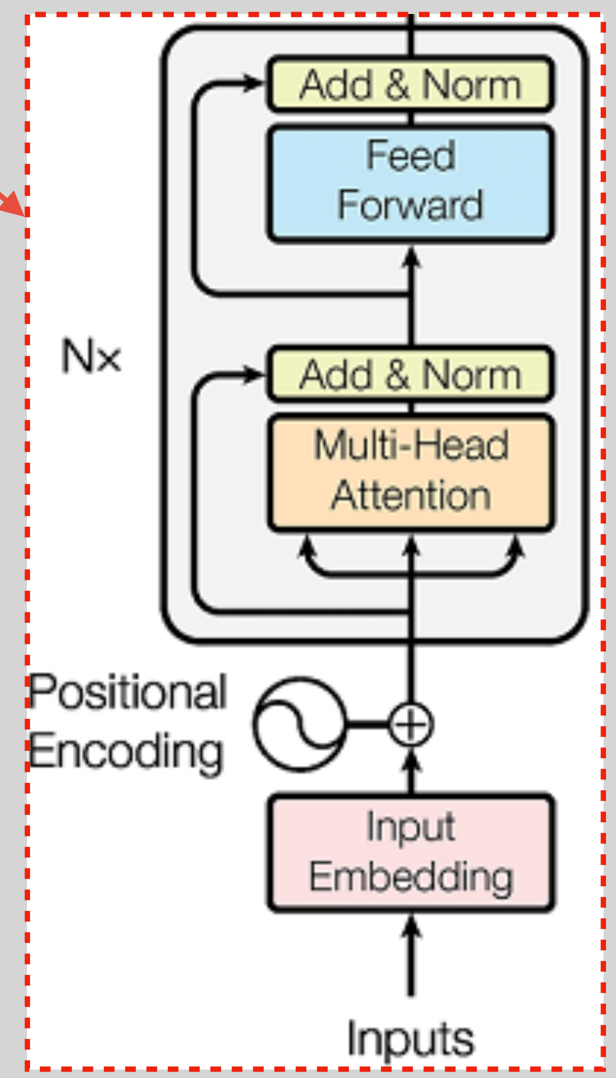


Transformer networks

- Transformer networks describe text sequences of a fully connected graph
  - The embedding at each sequence position, at each layer, is a function of the embeddings at all sequence positions from the previous layer.
  - The computation that produces the embedding at each sequence position at each layer is referred to as multi-head self attention (more on this in lecture 09).
  - This architecture is advantageous from both in terms of computational and optimization.



Multi-head self attention as presented in the 2017 NIPS paper: *"Attention is all you need!"* [1]



[1] Vaswani et al., 2017