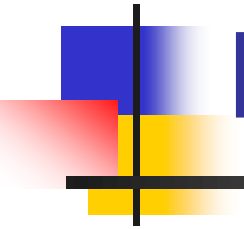
Deep Learning Using TensorFlow



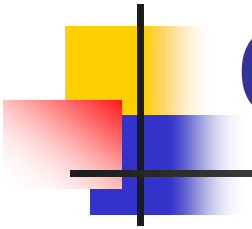
Dr. Ash Pahwa

Lesson 8:

Recurrent Neural Network + Reinforcement Learning

Lesson 8.1: Recurrent Neural Networks (RNN)

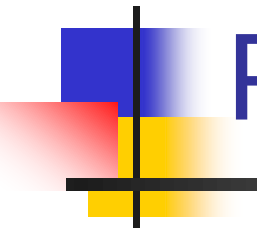
Copyright 2019 - Dr. Ash Pahwa



Outline

- Types of Recurrent Neural Networks (RNN)
- Definition of RNN
- Example: RNN
- Mathematics of RNN
  - Long Short-Term Memory (LSTM)
  - GRU – Gated Recurrent Unit
- RNN Implementation using TensorFlow
  - Next Lesson#8.2

Copyright 2019 - Dr. Ash Pahwa



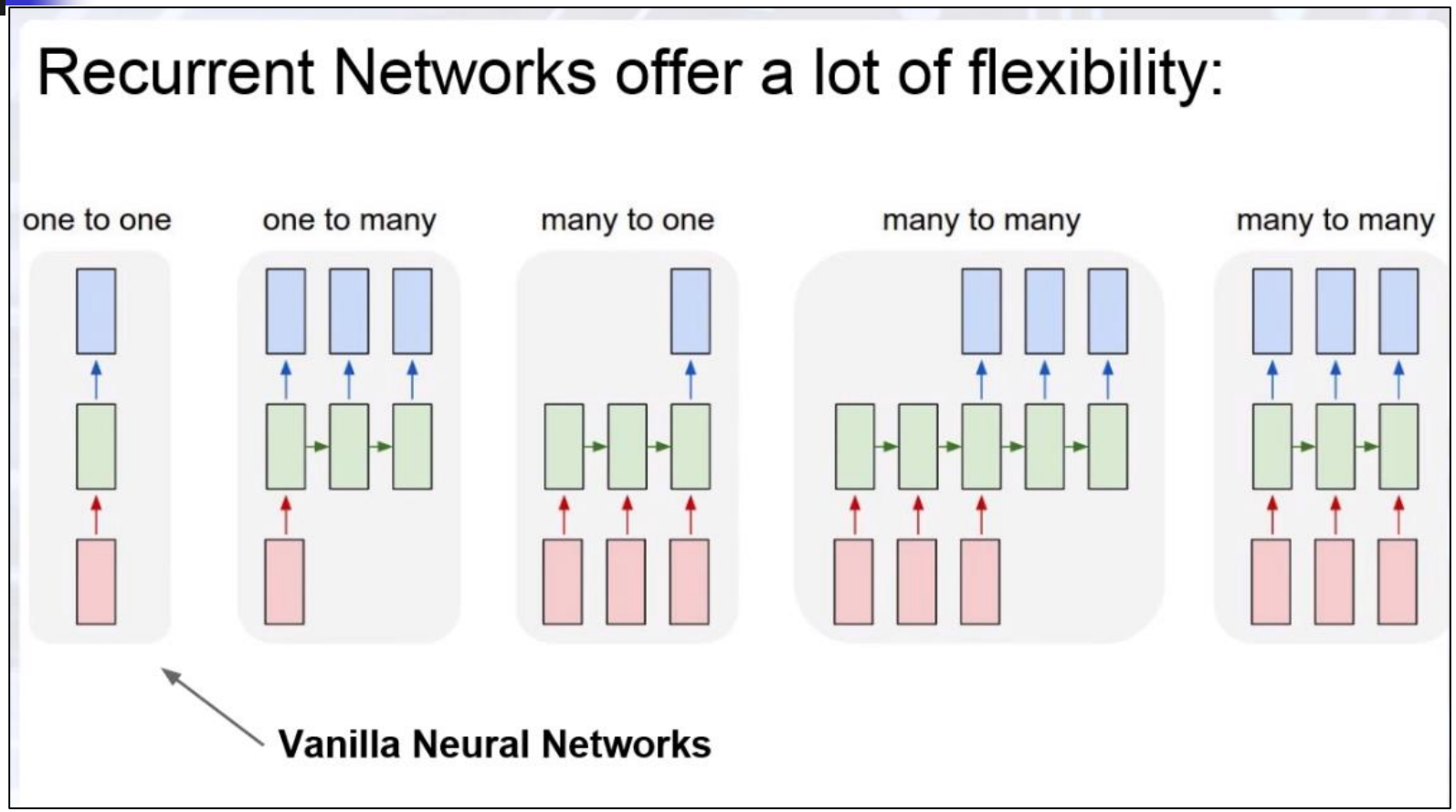
Types of Recurrent Neural Networks

Types of Recurrent Neural Networks

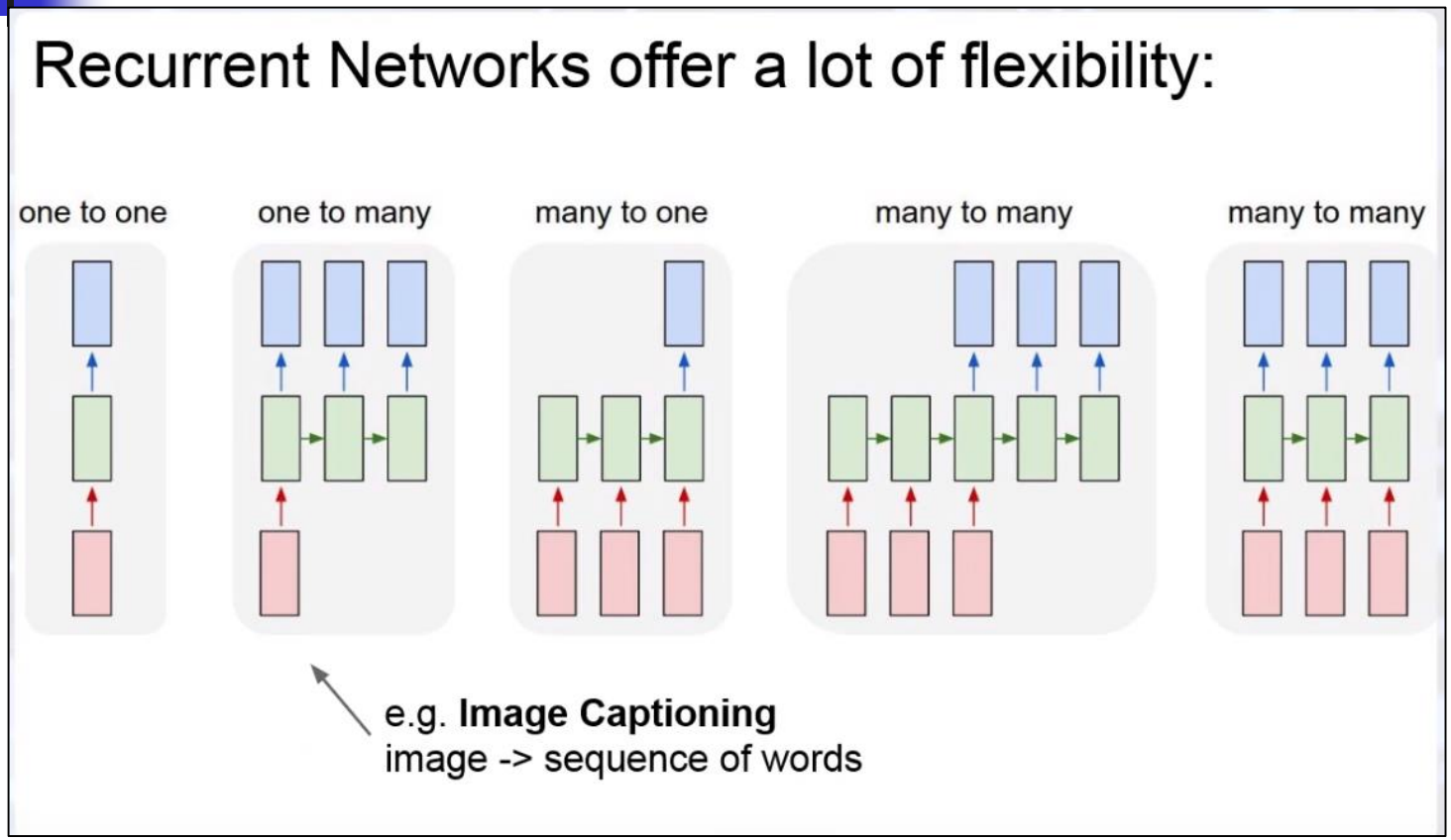
- The next 5 Slides contain information from the following talk
- Recurrent Network Language Models for Dense Image Captioning
  - Andrej Karpathy, Research Scientist,
  - OpenAI - RE•WORK Deep Learning Summit 2016
- YouTube Video
- https://www.youtube.com/watch?v=qPcCk1V1JO8

Copyright 2019 - Dr. Ash Pahwa

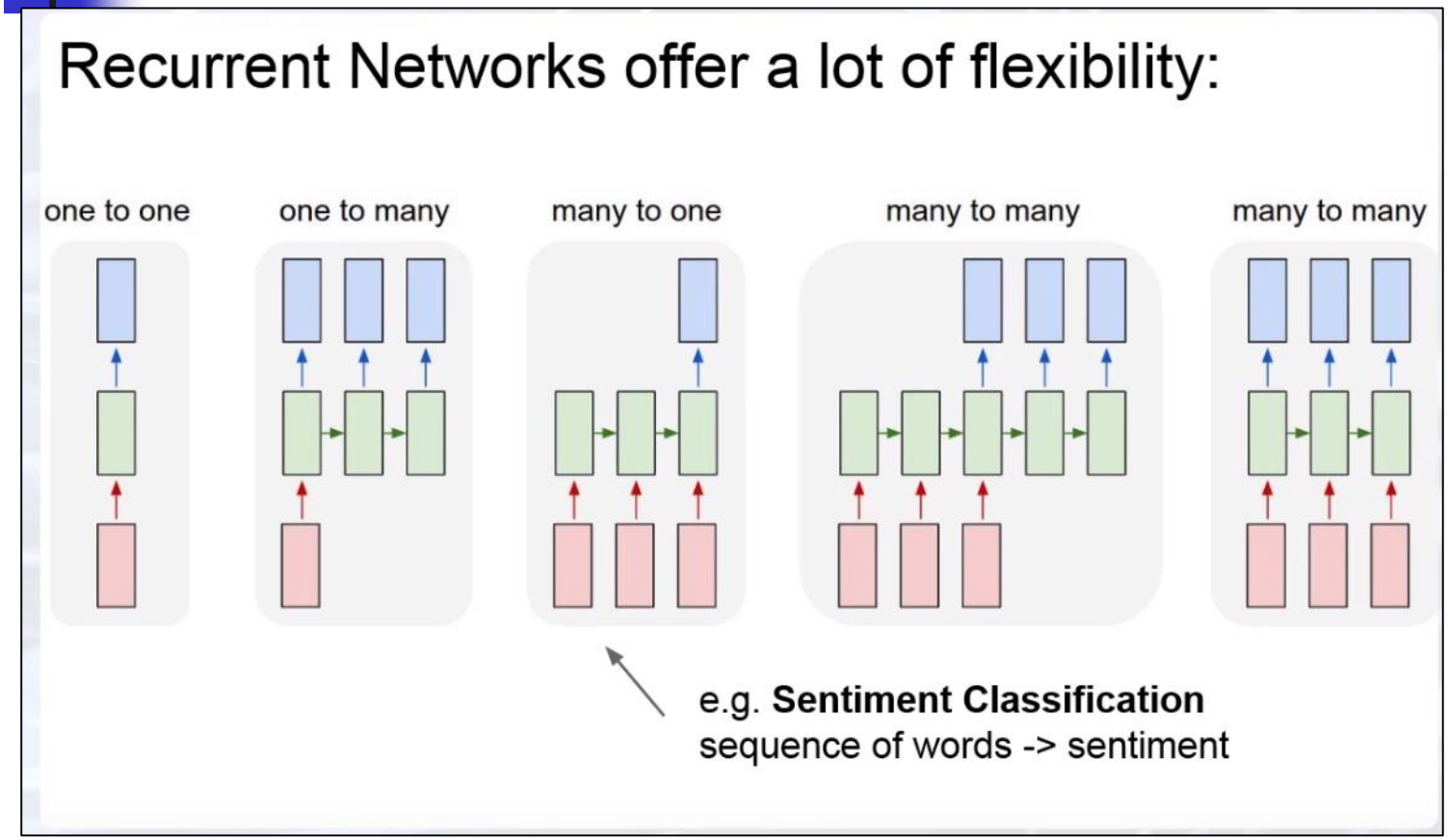
Image Classification



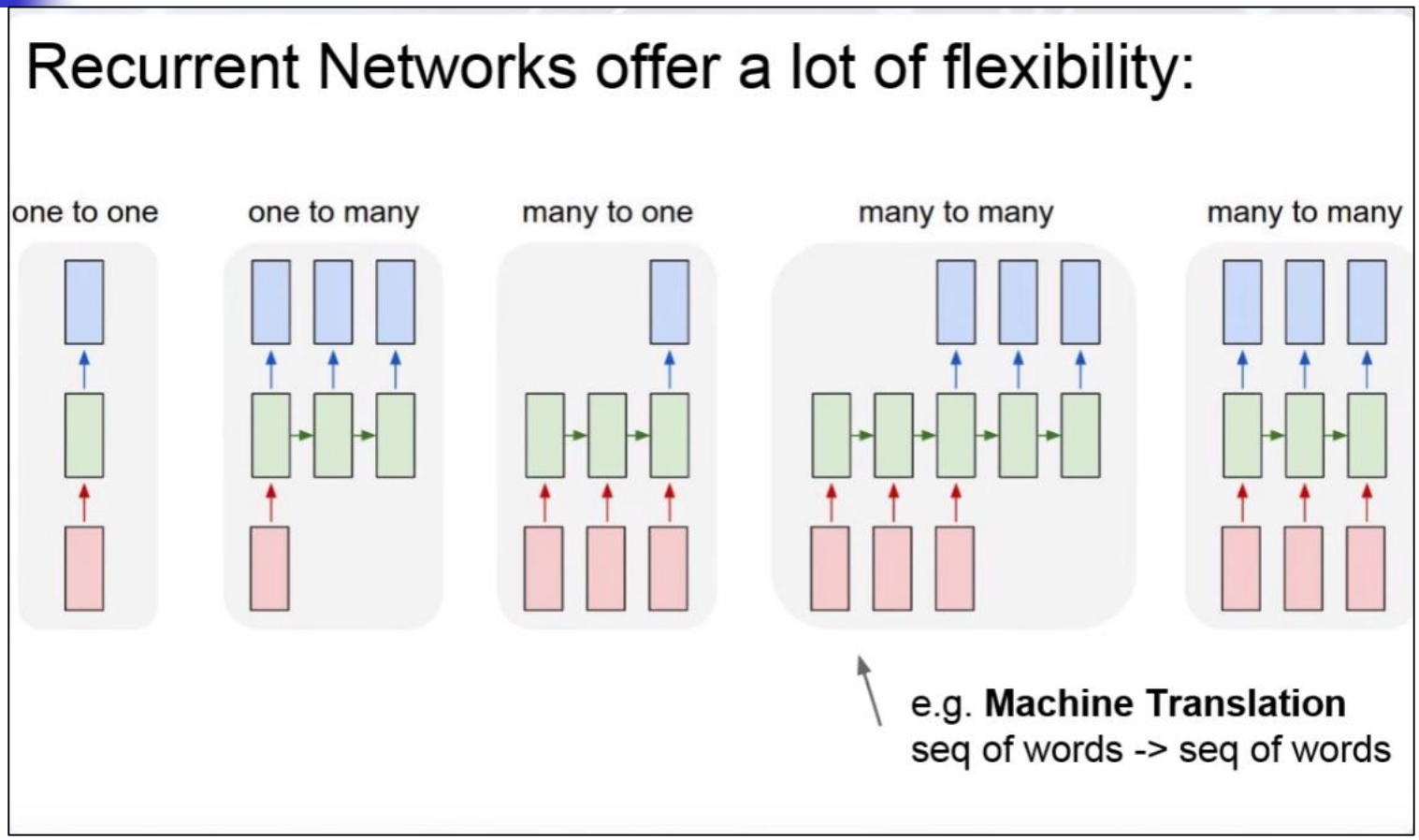
RNN Application
Image Captioning



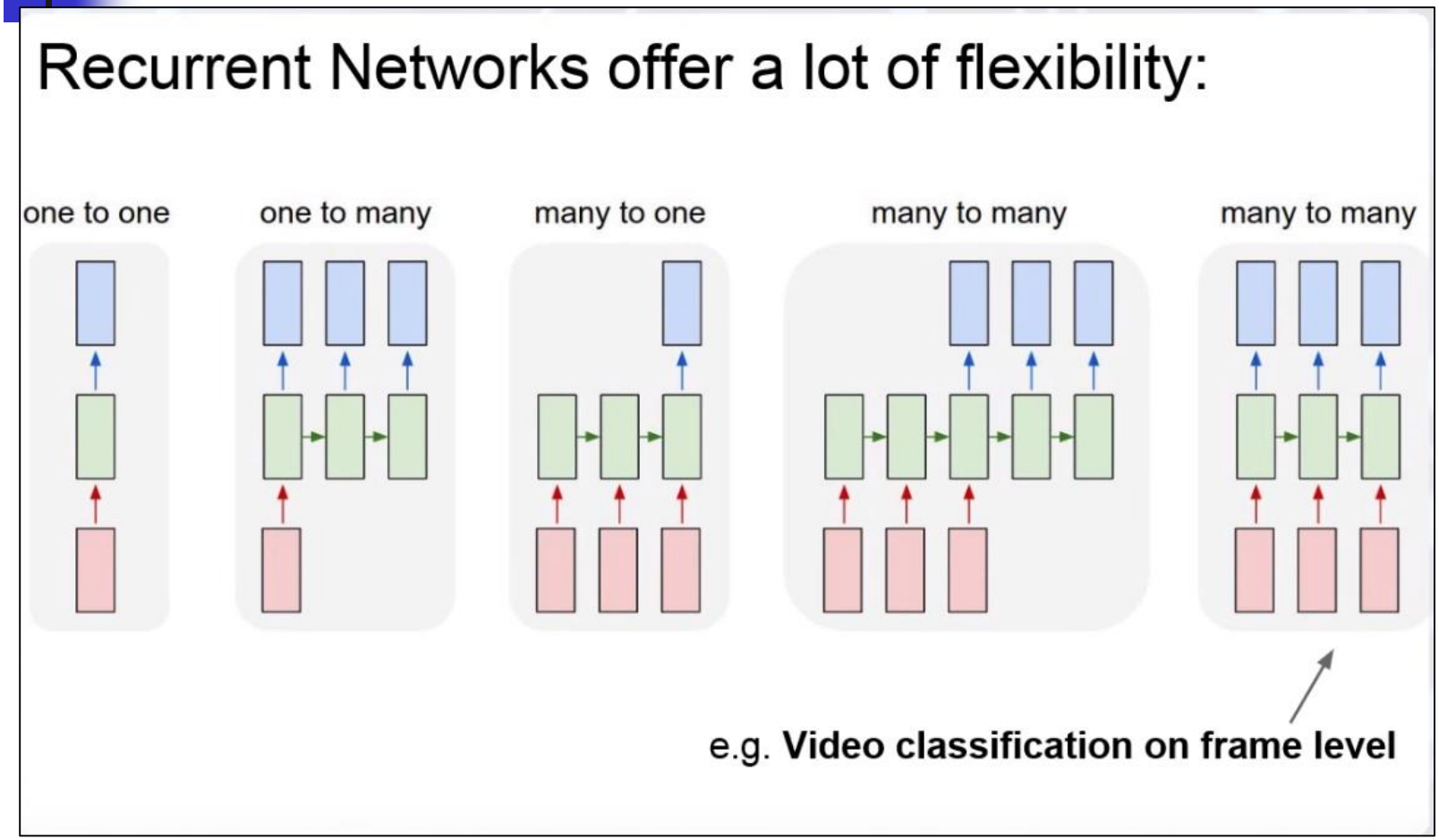
RNN Application
Sentiment Classification

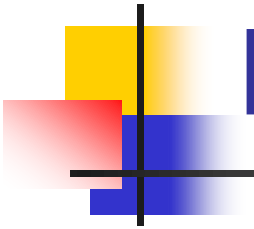


RNN Application
Machine Translation



RNN Application
Video Classification on Frame Level





RNN Applications

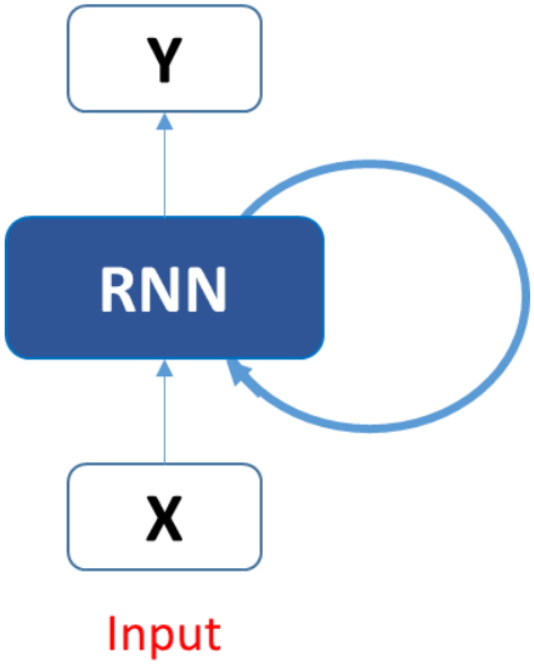
- Time Series Data
  - Value of a time-series at time ‘t’ is related to the time series data at time ‘t-1’

Copyright 2019 - Dr. Ash Pahwa

Definition of RNN

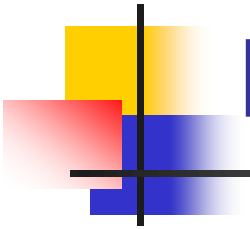
What is a Recurrent Neural Network (RNN)?

Neural Network with Feedback



Copyright 2019 - Dr. Ash Pahwa

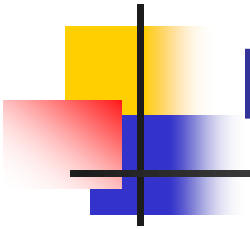
Feed Forward Neural Networks Drawbacks



- Input / Output
  - Fixed sized input and output
- No Memory
- Not designed for a sequence of data
  - Time series data

Copyright 2019 - Dr. Ash Pahwa

What are Recurrent Neural Networks?



- Input is a sequence of data
  - Time series data
  - Variable size input
- Output
  - Variable size output
- Feedback loop

Copyright 2019 - Dr. Ash Pahwa

Example: Recurrent Neural Network

Copyright 2019 - Dr. Ash Pahwa

Fully Forward Neural Network

- University Courses

	Courses
1	Math Review for Data Science
2	R Programming
3	Python Programming

Demand	Courses
Demand for Math Courses is high	Math Review for Data Science
Demand for R Prog is high	R Programming
Demand for Python Prog is high	Python Programming

Copyright 2019 - Dr. Ash Pahwa

Fully Forward Neural Network Encoding

- University Courses

	Demand	Code
1	Demand for Math Courses is high	$\begin{bmatrix}1\\0\\0\end{bmatrix}$
2	Demand for R Prog is high	$\begin{bmatrix}0\\1\\0\end{bmatrix}$
3	Demand for Python Prog is high	$\begin{bmatrix}0\\0\\1\end{bmatrix}$

	Courses	Code
1	Math Review for Data Science	$\begin{bmatrix}1\\0\\0\end{bmatrix}$
2	R Programming	$\begin{bmatrix}0\\1\\0\end{bmatrix}$
3	Python Programming	$\begin{bmatrix}0\\0\\1\end{bmatrix}$

Copyright 2019 - Dr. Ash Pahwa

Neural Network

$$\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\begin{bmatrix}1\\0\\0\end{bmatrix}=\begin{bmatrix}1\\0\\0\end{bmatrix}$$

$$\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\begin{bmatrix}0\\1\\0\end{bmatrix}=\begin{bmatrix}0\\1\\0\end{bmatrix}$$

$$\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\begin{bmatrix}0\\0\\1\end{bmatrix}=\begin{bmatrix}0\\0\\1\end{bmatrix}$$

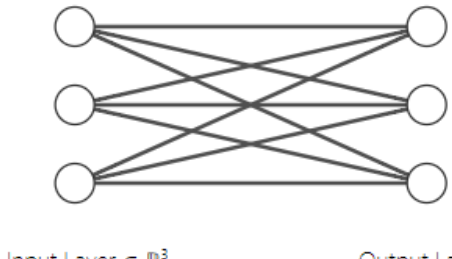
Input

Demand for Math Courses is high $\begin{bmatrix}1\\0\\0\end{bmatrix}$

Demand for R Prog Courses is high $\begin{bmatrix}0\\1\\0\end{bmatrix}$

Demand for Python Prog Courses is high $\begin{bmatrix}0\\0\\1\end{bmatrix}$

$$\begin{bmatrix}1&0&0\\0&1&0\\0&0&1\end{bmatrix}\begin{bmatrix}?\\?\\?\end{bmatrix}=$$



Output

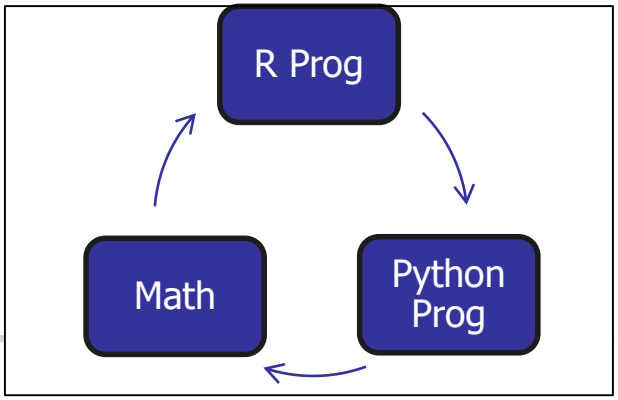
$\begin{bmatrix}1\\0\\0\end{bmatrix}$ Math Courses offered

$\begin{bmatrix}0\\1\\0\end{bmatrix}$ R Prog Courses offered

$\begin{bmatrix}0\\0\\1\end{bmatrix}$ Python Prog Courses offered

Copyright 2019 - Dr. Ash Pahwa

Example: RNN



- University Courses

Quarter	Months	Courses
Fall 2017	Oct – Dec	Math
Winter 2018	Jan – March	R Programming
Spring 2018	April – June	Python
Summer 2018	July - Sep	School Closed

Quarter	Months	Courses
Fall 2018	Oct – Dec	Math
Winter 2019	Jan – March	R Programming
Spring 2019	April – June	Python
Summer 2019	July - Sep	School Closed

Copyright 2019 - Dr. Ash Pahwa

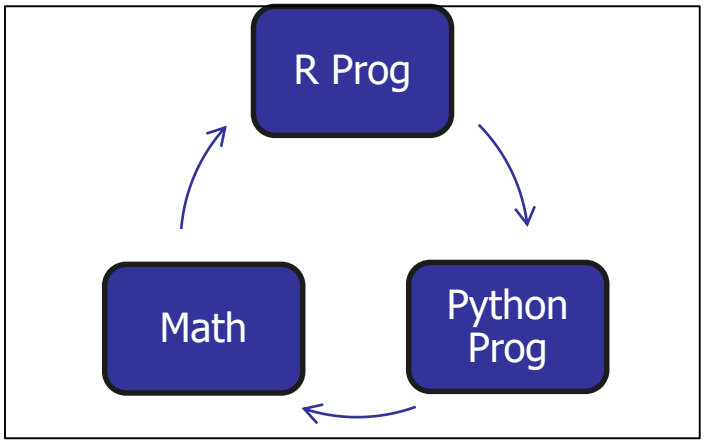
Example: RNN

Quarter	Months	Courses
Fall 2017	Oct – Dec	Math
Winter 2018	Jan – March	R Programming
Spring 2018	April – June	Python

$$\begin{bmatrix}1\\0\\0\end{bmatrix}$$ Math Course

$$\begin{bmatrix}0\\1\\0\end{bmatrix}$$ R Prog Course

$$\begin{bmatrix}0\\0\\1\end{bmatrix}$$ Python Prog Course



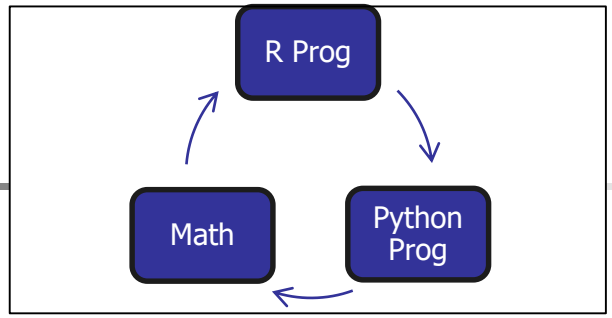
Copyright 2019 - Dr. Ash Pahwa

Example: RNN

$$\begin{bmatrix}0 & 0 & 1\\ 1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}\begin{bmatrix}1\\0\\0\end{bmatrix}=\begin{bmatrix}0\\1\\0\end{bmatrix}$$

$$\begin{bmatrix}0 & 0 & 1\\ 1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}\begin{bmatrix}0\\1\\0\end{bmatrix}=\begin{bmatrix}0\\0\\1\end{bmatrix}$$

$$\begin{bmatrix}0 & 0 & 1\\ 1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}\begin{bmatrix}0\\0\\1\end{bmatrix}=\begin{bmatrix}1\\0\\0\end{bmatrix}$$



Input

$\begin{bmatrix}1\\0\\0\end{bmatrix}$ Math Courses offered

$\begin{bmatrix}0\\1\\0\end{bmatrix}$ R Prog Courses offered

$\begin{bmatrix}0\\0\\1\end{bmatrix}$ Python Prog Courses offered

$$\begin{bmatrix}0 & 0 & 1\\ 1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}\begin{bmatrix}?\\?\\?\end{bmatrix}=$$

RNN

Output

$\begin{bmatrix}0\\1\\0\end{bmatrix}$ R Prog Courses offered

$\begin{bmatrix}0\\0\\1\end{bmatrix}$ Python Prog Courses offered

$\begin{bmatrix}1\\0\\0\end{bmatrix}$ Math Courses offered

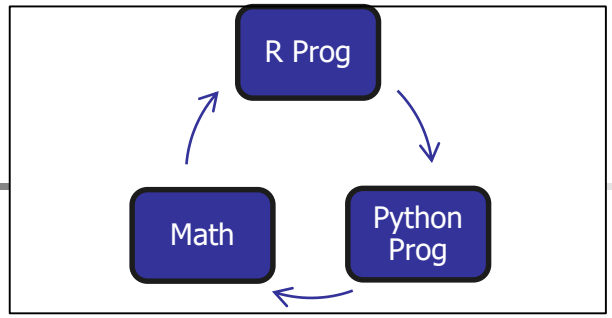
Copyright 2019 - Dr. Ash Pahwa

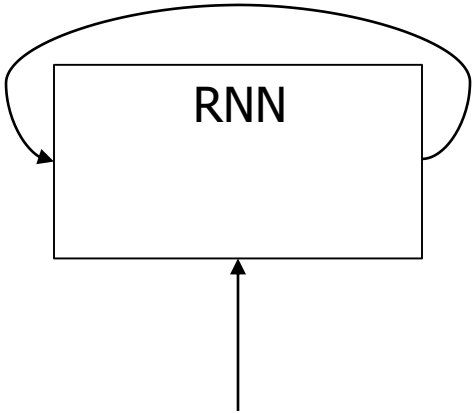
Example: RNN

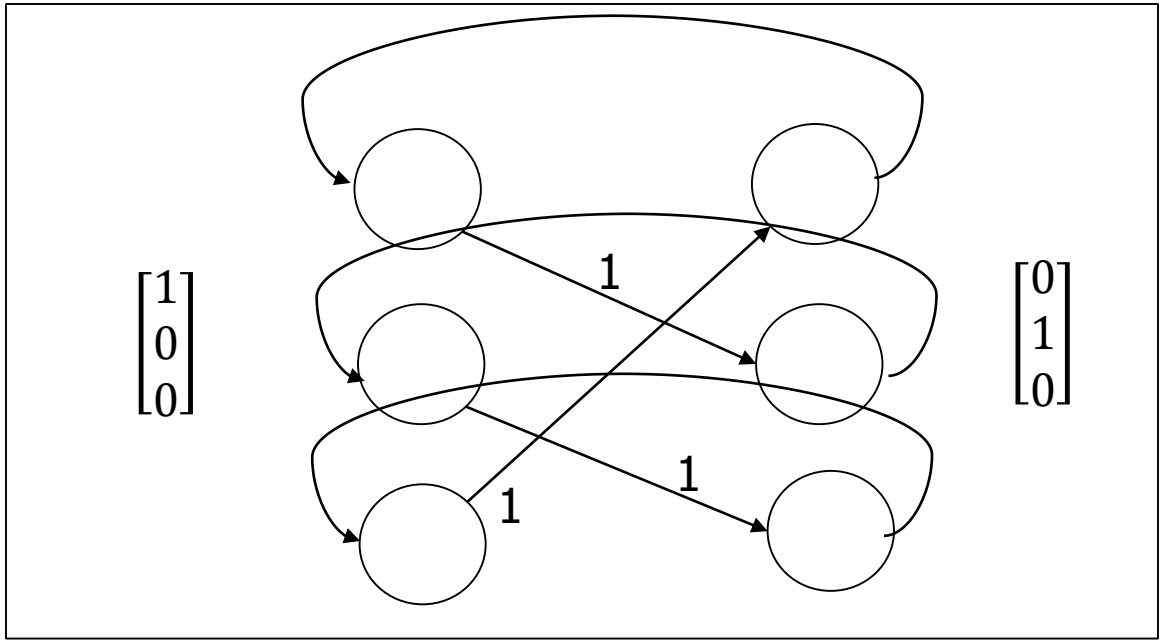
$$\begin{bmatrix}0 & 0 & 1\\ 1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}\begin{bmatrix}1\\0\\0\end{bmatrix}=\begin{bmatrix}0\\1\\0\end{bmatrix}$$

$$\begin{bmatrix}0 & 0 & 1\\ 1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}\begin{bmatrix}0\\1\\0\end{bmatrix}=\begin{bmatrix}0\\0\\1\end{bmatrix}$$

$$\begin{bmatrix}0 & 0 & 1\\ 1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}\begin{bmatrix}0\\0\\1\end{bmatrix}=\begin{bmatrix}1\\0\\0\end{bmatrix}$$





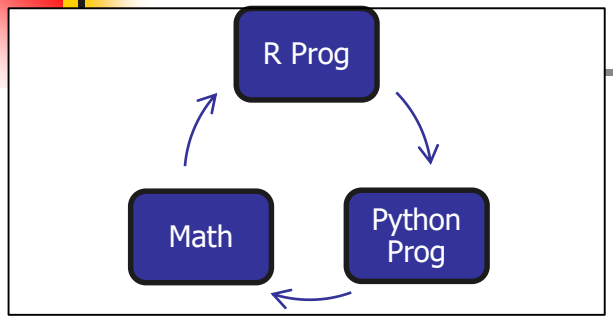


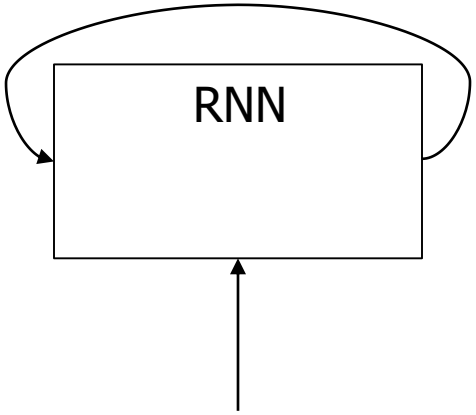
$$\begin{bmatrix}1\\0\\0\end{bmatrix}$$

$$\begin{bmatrix}0\\1\\0\end{bmatrix}$$

Copyright 2019 - Dr. Ash Pahwa

Example: RNN
Combination of Demand and Previous State

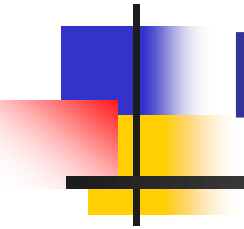




Demand for the course

#	Quarter	Year	Input Demand	Course Offered
1	Fall	1		Math
2	Winter	2	Demand Math	Math
3	Spring	2		R Prog
4	Fall	2	Demand R	R Prog
5	Winter	3		Python
6	Spring	3		Math
7	Fall	3		R Prog
8	Winter	4		Python

Copyright 2019 - Dr. Ash Pahwa



Mathematics of RNN

RNN

- Recursive definition of RNN
  - $S_t = F_w(S_{t-1}, X_t)$
  - The new state (S_t) of RNN is a function of
    - Old State at time 't-1' (S_{t-1})
    - Input at time 't' (X_t)
    - F_w is a $'\tanh()'$ function

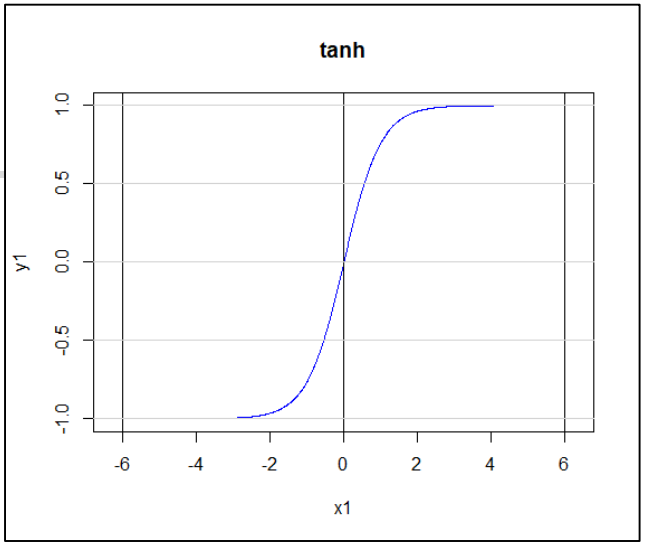
Copyright 2019 - Dr. Ash Pahwa

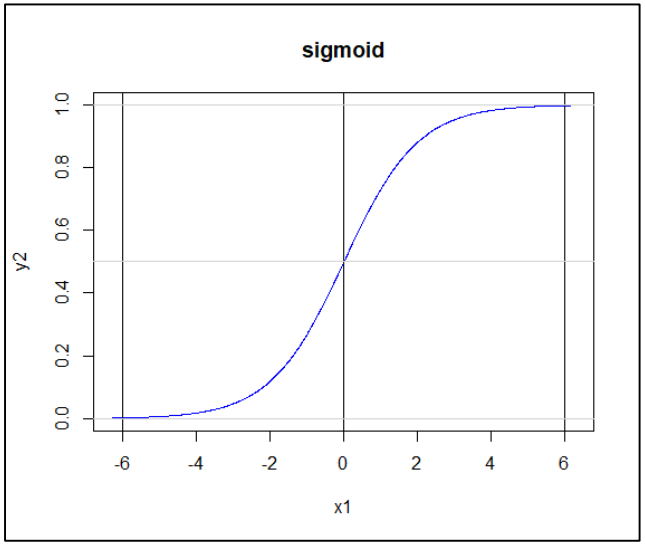
Hyperbolic 'tanh' function

- $\tanh(x) = \frac{\sinh(x)}{\cosh(x)} = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$
- Similar to Sigmoid Function

$$Sigmoid\ Function = f(x) = \frac{1}{1+e^{-x}}$$

```
> ##############################################
> # Hyperbolic tan function
> # tanh function + Sigmoid
> ##############################################
> # tanh Function
> x1 = seq(-2*pi, 2*pi, 0.01)
> y1 = tanh(x1)
> plot(x1,y1,type='l',col="blue",main="tanh")
> abline(v=seq(-6,6,6),col="black",lty=1)
> abline(h=seq(-1,1,0.5),col="lightgrey",lty=1)
> ##############################################
> # Sigmoid Function
> y2 <- 1/(1+exp(-x1))
> plot(x1,y2,type='l',col="blue",main="sigmoid")
> abline(v=seq(-6,6,6),col="black",lty=1)
> abline(h=seq(0,1,0.5),col="lightgrey",lty=1)
> ##############################################
```

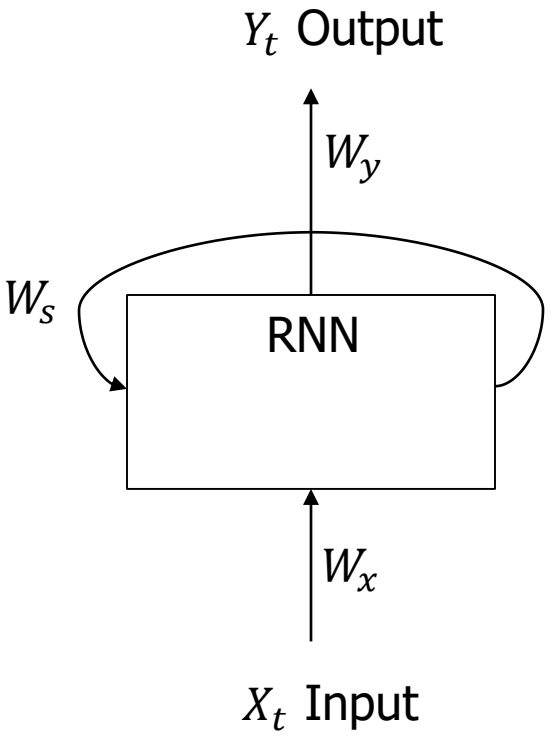




Copyright 2019 - Dr. Ash Pahwa

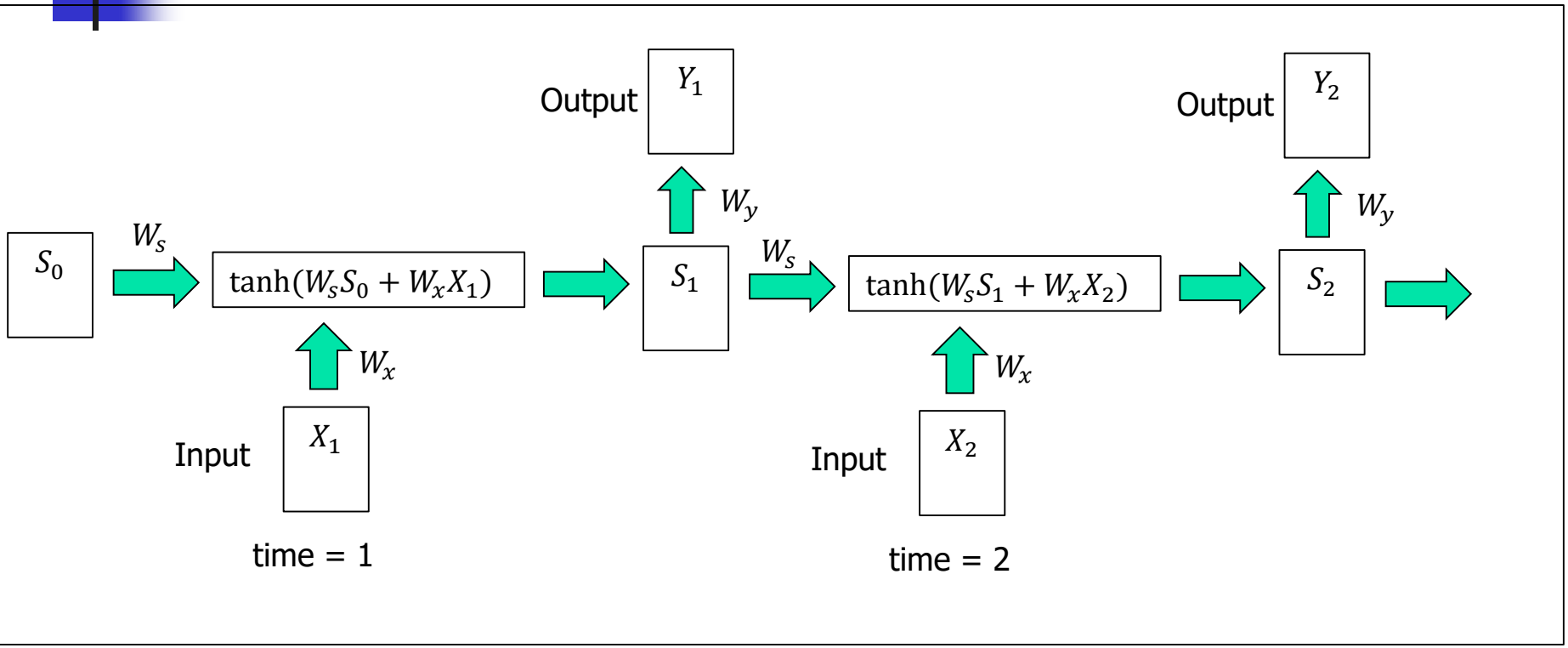
RNN

- $S_t = F_w(S_{t-1}, X_t)$
- $S_t = \tanh(W_s S_{t-1} + W_x X_t)$
- $Y_t = W_y S_t$



First Layer RNN

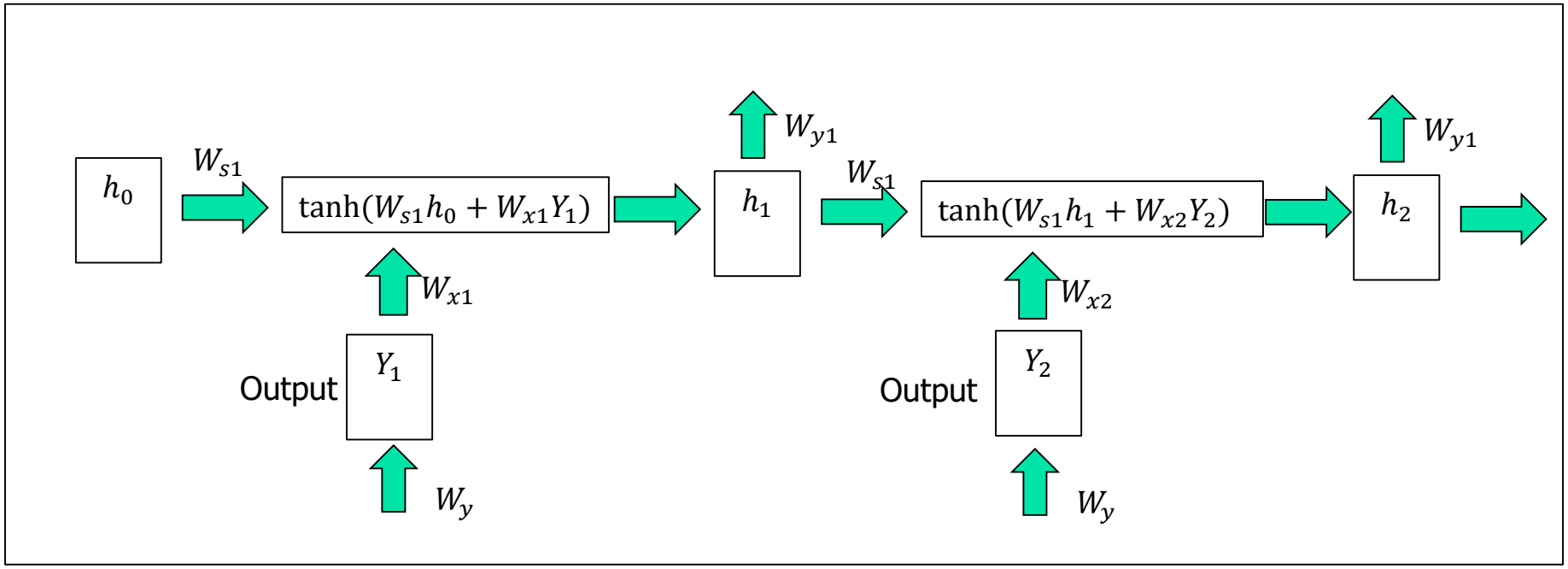
- $S_t = \tanh(W_s S_{t-1} + W_x X_t)$
- $Y_t = W_y S_t$



Copyright 2019 - Dr. Ash Pahwa

Second Layer RNN

- $S_t = \tanh(W_s S_{t-1} + W_x X_t)$
- $Y_t = W_y S_t$



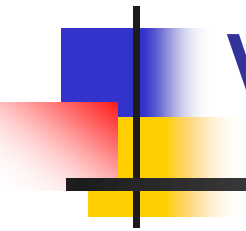
Accuracy improves when more layers are added to RNN

Copyright 2019 - Dr. Ash Pahwa

How to Compute the Weights?

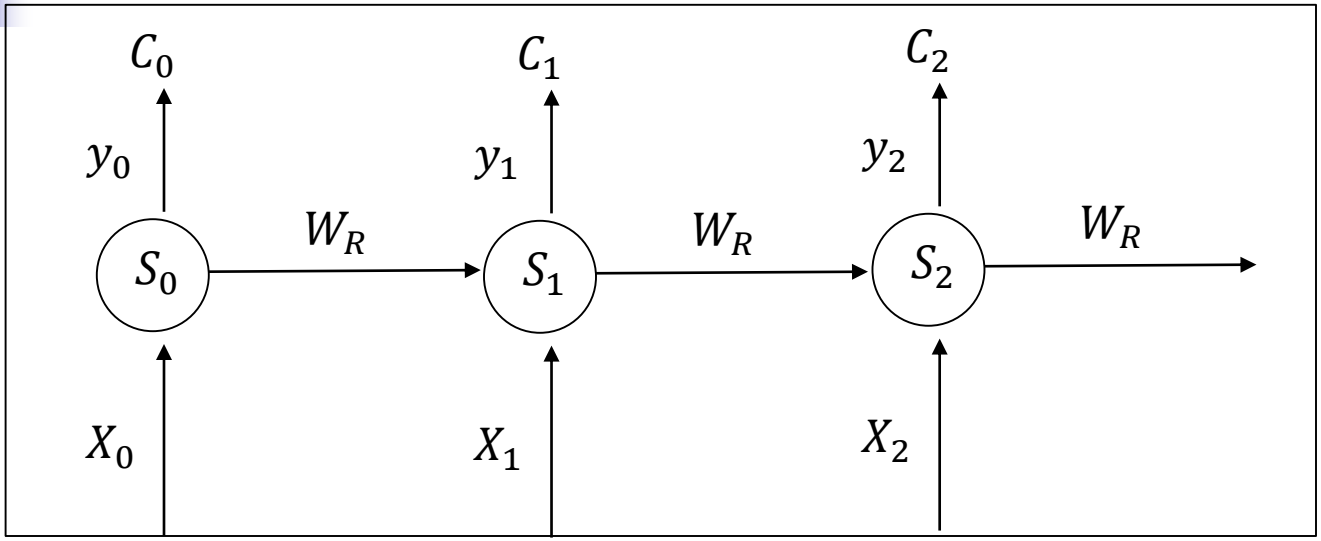
- Feed Forward Neural Network
  - Back Propagation method
- Recurrent Neural Network
  - Back Propagation through time

Copyright 2019 - Dr. Ash Pahwa



Vanishing/Exploding Gradient

Compute the Gradient



- $\frac{\partial C}{\partial W_R} = \sum_t \frac{\partial C_t}{W_R}$
  - $\frac{\partial C_2}{\partial W_R} = \frac{\partial C_2}{\partial y_2}\frac{\partial y_2}{\partial S_2}\frac{\partial S_2}{\partial g}\frac{\partial g}{\partial a}\frac{\partial a}{\partial W_R}$
  - $a = (W_1 X_2 + W_R S_1)$

If gradient < 1
After 100 steps – gradient will vanish

If gradient > 1
After 100 steps – gradient will explode

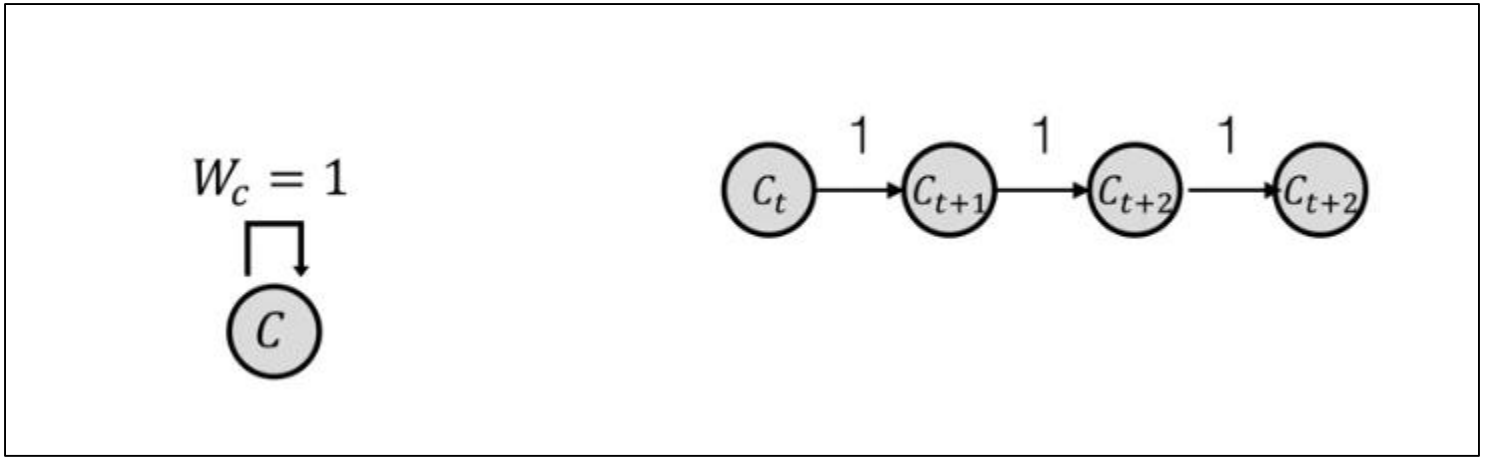
Copyright 2019 - Dr. Ash Pahwa

How to Solve
Vanishing/Exploding Gradient

- Exploding gradient
  - Truncated BPTT (Back Propagation Through Time)
  - Clip gradient at threshold
  - RMSProp to adjust Learning Rate
- Vanishing Gradient
  - Harder to detect
  - Weight initialization
  - Use ReLu activation because it's gradient is 1
  - RMSprop to adjust Learning Rate
  - New Neural Network Architecture
    - LSTM (Long Short-Term Memory) and
    - GRU (Gated Recurrent Units)

Copyright 2019 - Dr. Ash Pahwa

Use Memory to store any Information



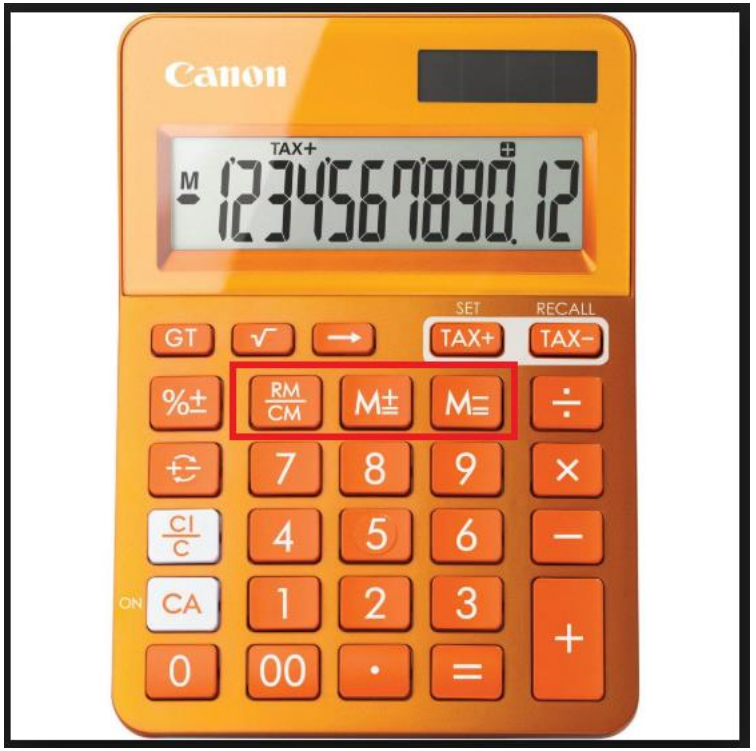
Memory will be used to store the gradient

Therefore, the vanishing gradient problem can be solved

Copyright 2019 - Dr. Ash Pahwa

Long Short-Term Memory

- All Calculators have 3 functions
  - Clear memory (CM)
  - Add to the memory (M+)
  - Recall memory (RM)

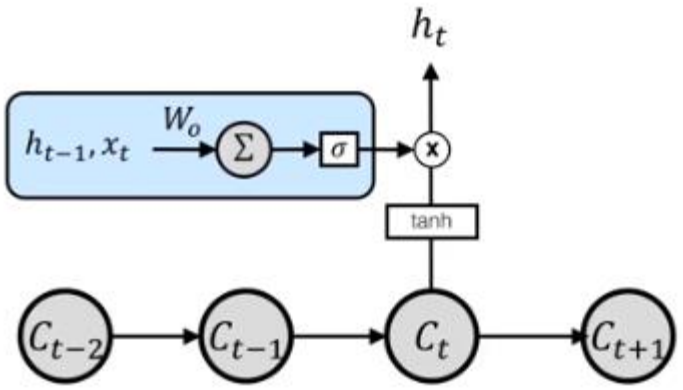


- Gates built in the RNN
  - Forget Gate: Flush the memory
  - Input Gate: Add to the memory
  - Output Gate: Get from the memory

Copyright 2019 - Dr. Ash Pahwa

Output Gate
Recall Memory

Output gate

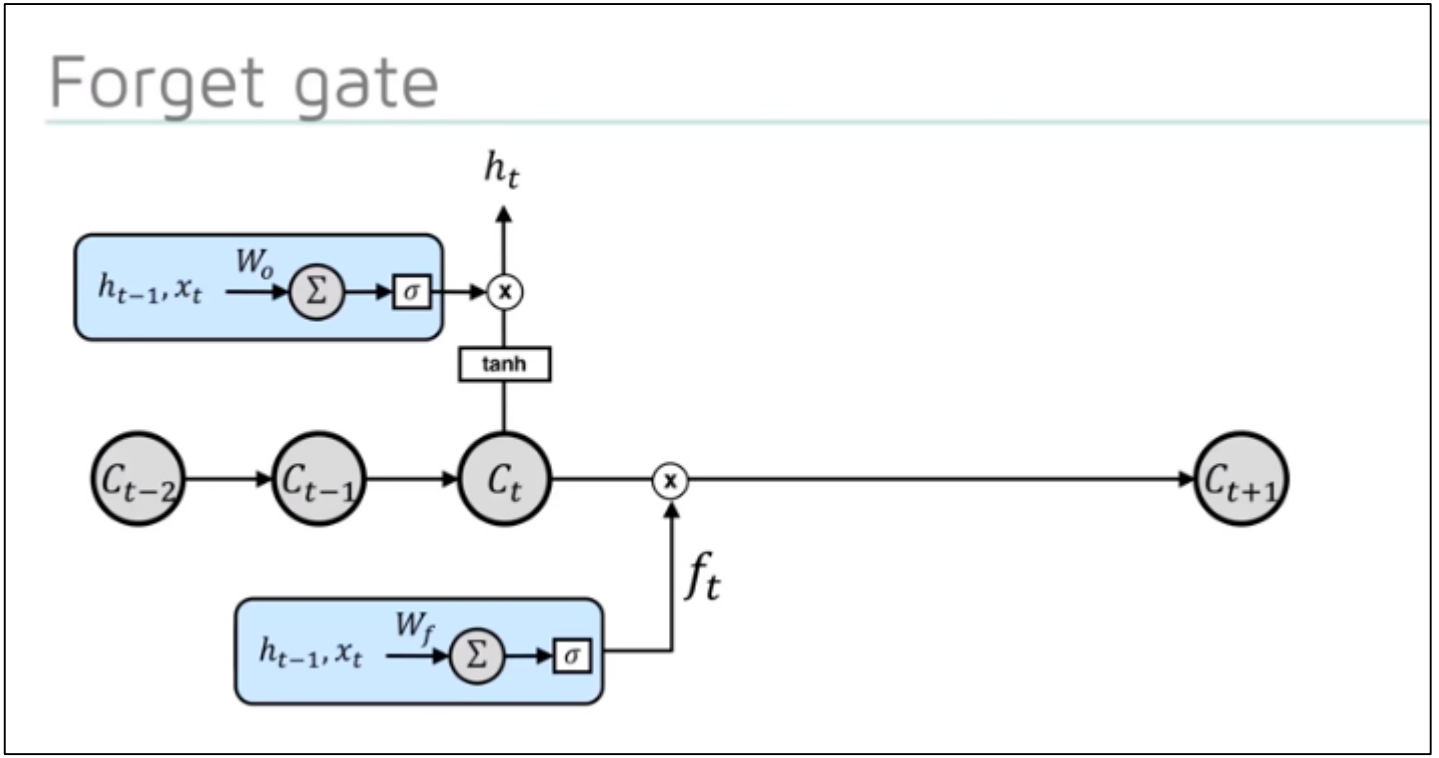


Affine layer!

$$h_t = \sigma(W_o \cdot [x_t, h_{t-1}] + b_o) \odot \tanh(C_t)$$
$$= o_t \odot \tanh(C_t)$$

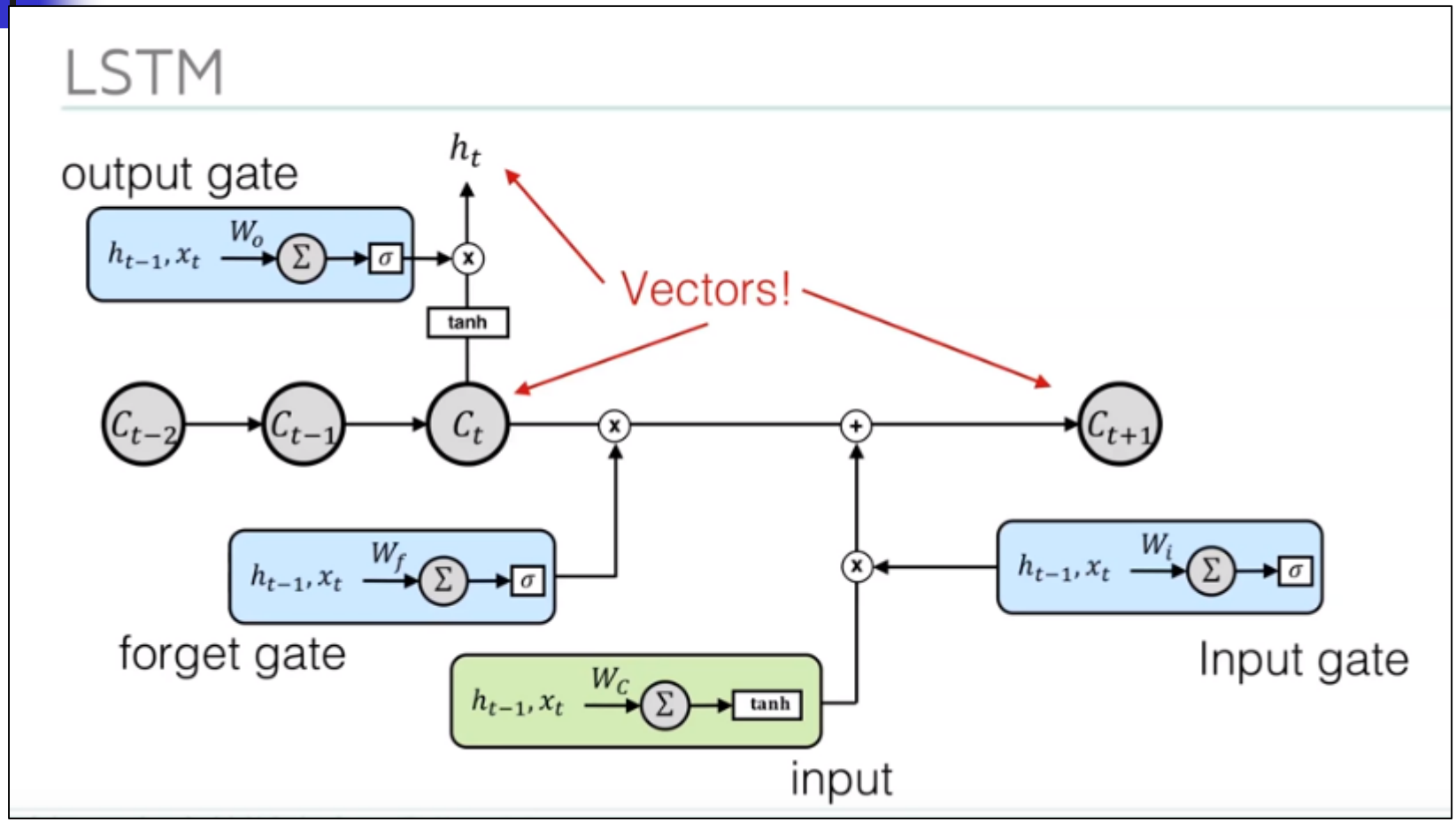
Copyright 2019 - Dr. Ash Pahwa

Forget Gate
Clear Memory

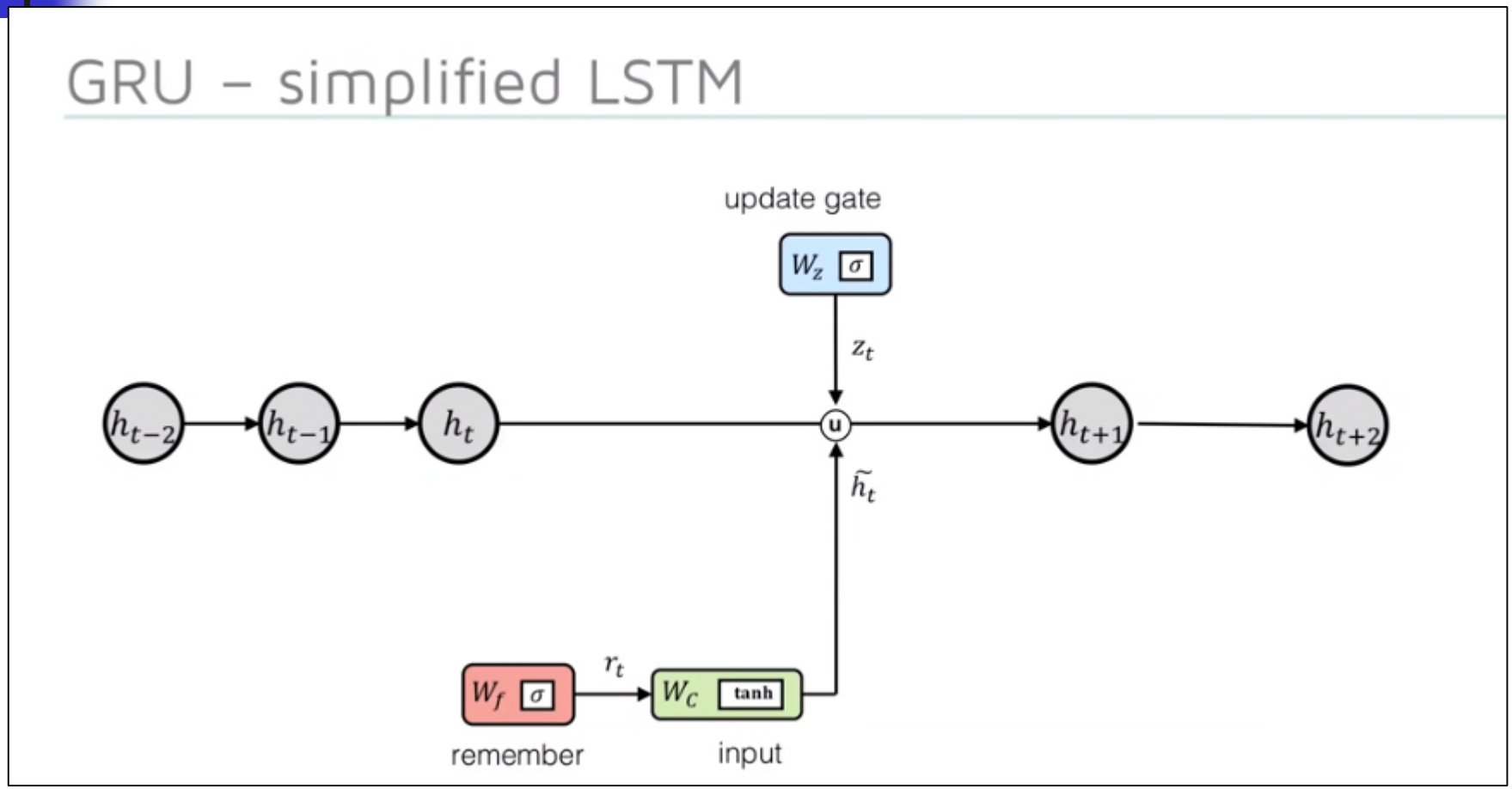


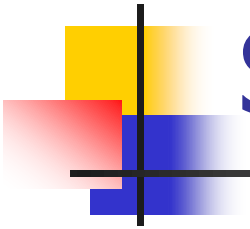
Copyright 2019 - Dr. Ash Pahwa

Input Gate
Add to the Memory



Gated Recurrent Unit Simplified LSTM





Summary

- Types of Recurrent Neural Networks (RNN)
- Definition of RNN
- Example: RNN
- Mathematics of RNN
  - Long Short-Term Memory (LSTM)
  - GRU – Gated Recurrent Unit
- RNN Implementation using TensorFlow
  - Next Lesson#8.2

Copyright 2019 - Dr. Ash Pahwa