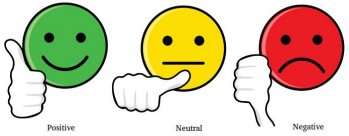
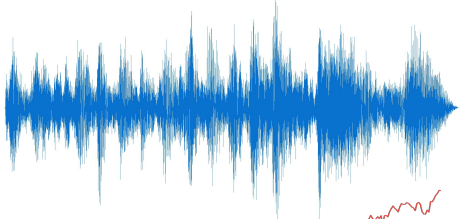
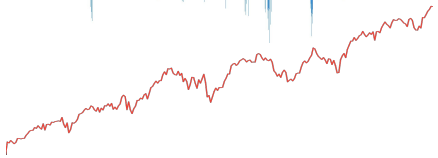
Recurrent Neural Networks

CS344
Deep Learning

WELLESLEY

Sequence Data

Word labeling	I like red apples	pron verb adj noun
Machine translation	Do you have a pet?	¿Tienes una mascota?
Text generation	Write a poem	Roses are red...
Sentiment classification	Good, cheap food!	
Speech recognition		I stay out too late
Time series prediction		54.7

Word Encoding

Apple	College	Ruby	Studying	Fox	Pi
1	2	3	4	5	6

Word Encoding

Apple	College	Ruby	Studying	Fox	Pi
1	0	0	0	0	0
0	1	0	0	0	0
0	0	1	0	0	0
0	0	0	1	0	0
0	0	0	0	1	0
0	0	0	0	0	1
0	0	0	0	0	0

Word Embedding

	Apple	College	Ruby	Studying	Fox	Pi
Size	-0.5	0.6	-0.8	0.1	0.3	-0.6
Red	0.8	0.03	0.91	0.0	0.7	0.01
Verb	0.01	-0.01	-0.07	0.99	0.4	0.0
Scholastic	0.2	0.97	0.03	0.87	0.02	0.3
Animal	0.05	0.01	-0.04	-0.02	0.99	-0.1
Numerical	-0.02	0.21	0.0	0.3	0.01	0.99
Cost	-0.8	0.92	0.94	0.2	0.04	0.06

Word Embedding

Apple	College	Ruby	Studying	Fox	Pi
0.52	-1.23	0.16	0.29	0.44	0.4
-0.83	1.42	0.91	0.35	0.06	1.07
0.5	-0.69	-0.55	-0.87	0.16	0.44
1.29	-1.16	1.39	-0.73	0.93	0.64
0.12	0.0	-0.14	-0.08	0.19	0.33
⋮	⋮	⋮	⋮	⋮	⋮
0.27	0.32	-0.25	-0.11	1.51	0.15

Embedding words in a sentence

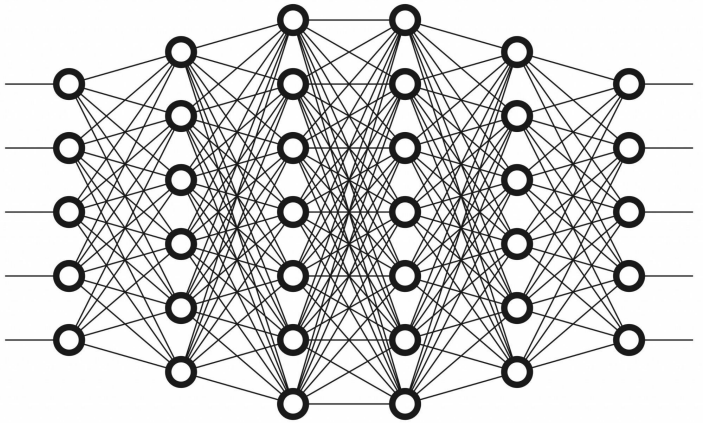
I	like	red	apples
-0.87	1.61	0.83	0.55
1.19	0.92	0.62	-0.73
-1.62	-0.44	-0.78	0.12
-0.74	1.07	-0.09	1.48
0.8	-0.52	-0.85	0.31
⋮	⋮	⋮	⋮
0.23	-0.13	-0.25	0.56

Why use *recurrent* NN rather than MLP?

An RNN (like CNN) uses what it's learned about one part of input on other parts of input

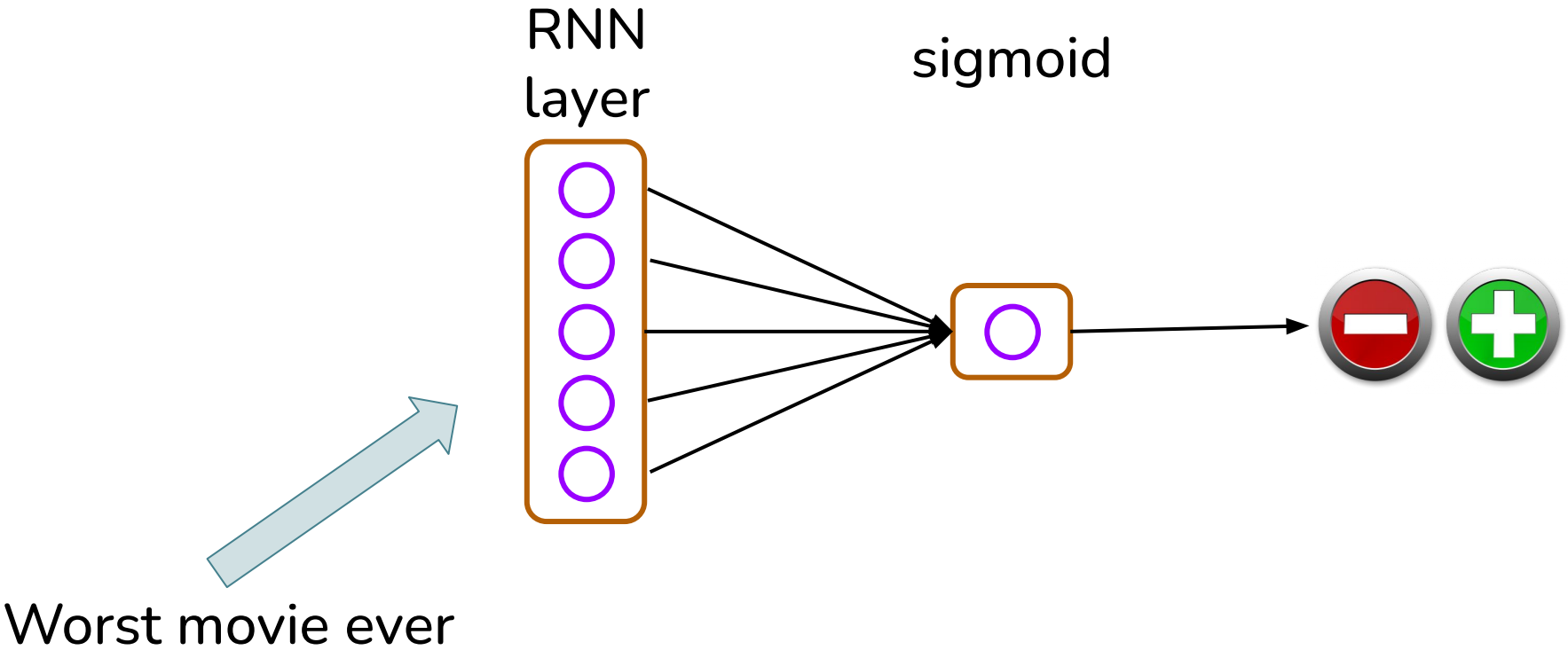
An RNN (like CNN) uses fewer parameters per layer

RNN allows for different length inputs and outputs

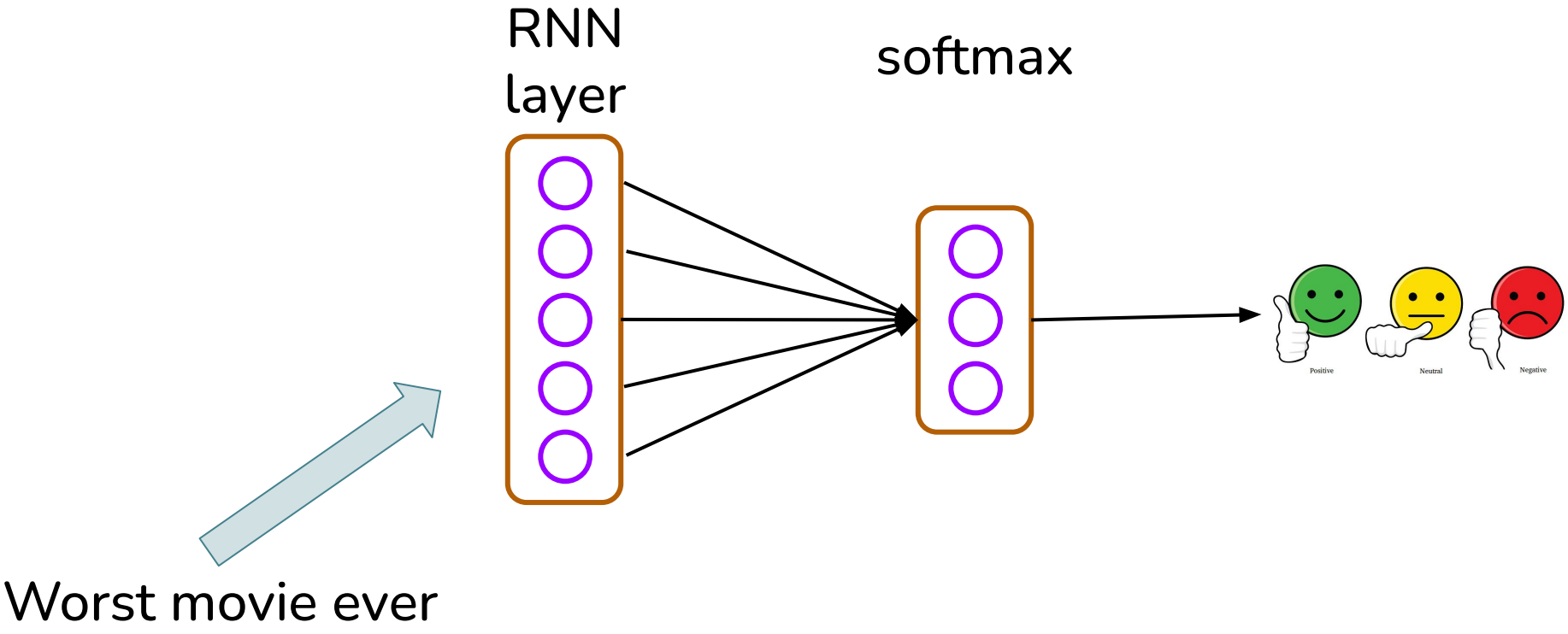


An RNN is well suited to modeling the sequential nature of data

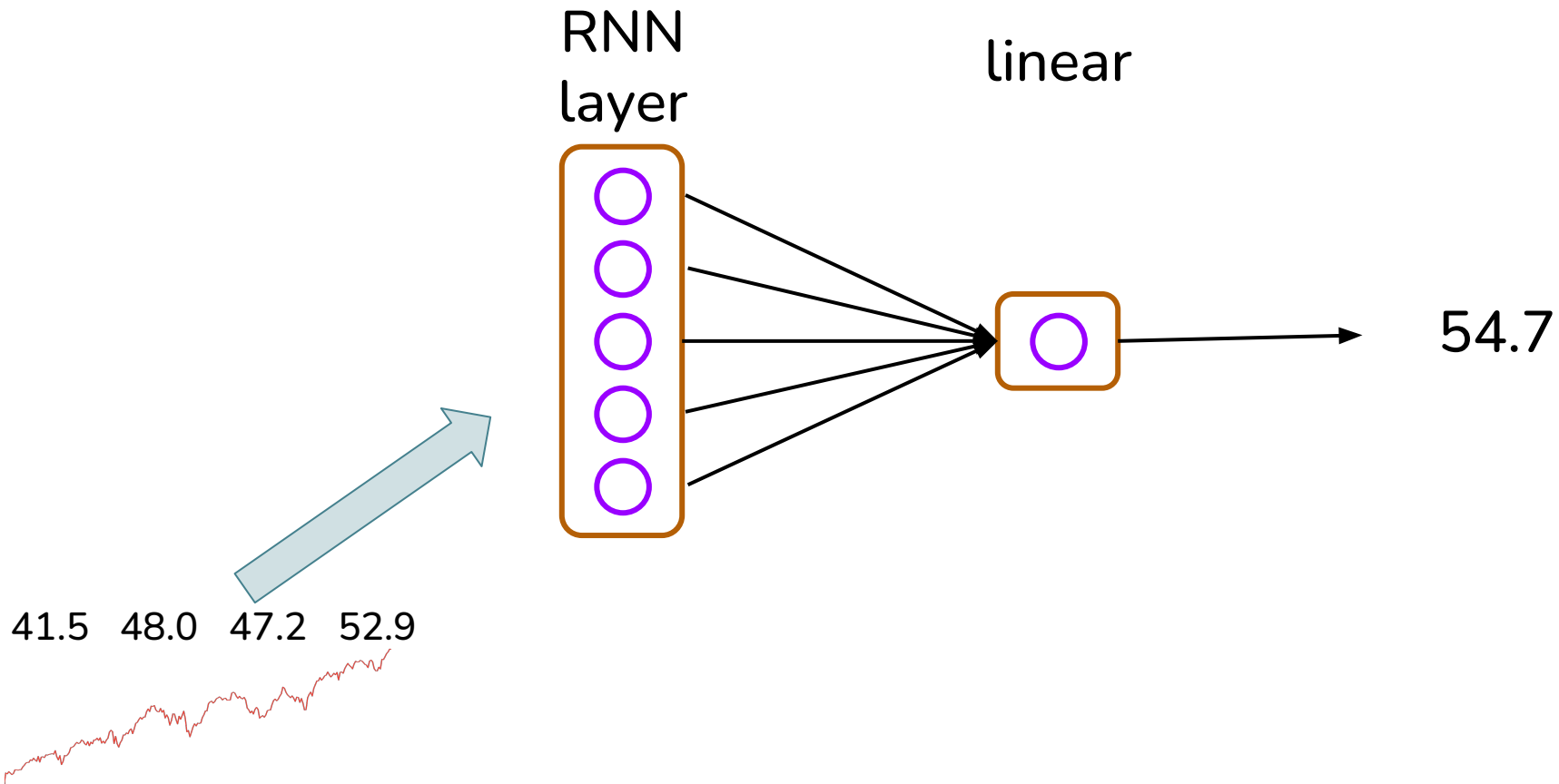
Sentiment Classification



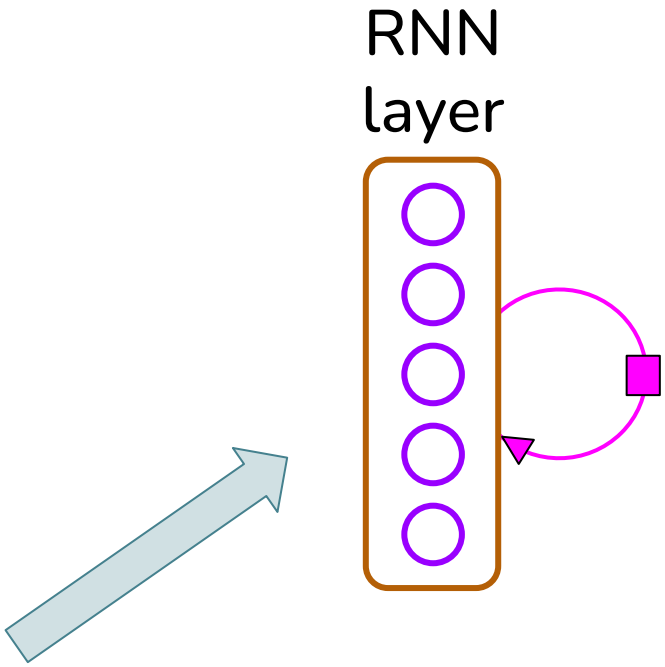
Sentiment Classification



Time Series Prediction

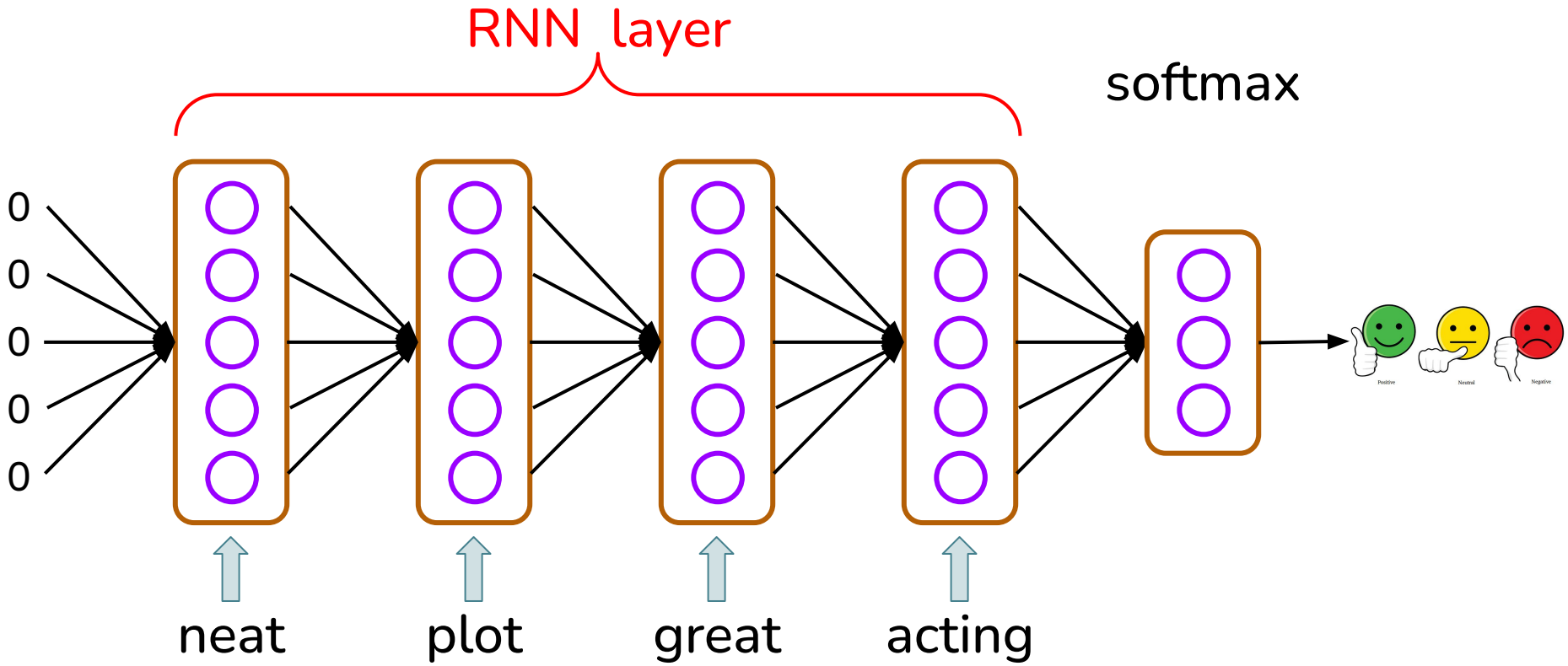


RNN Layer

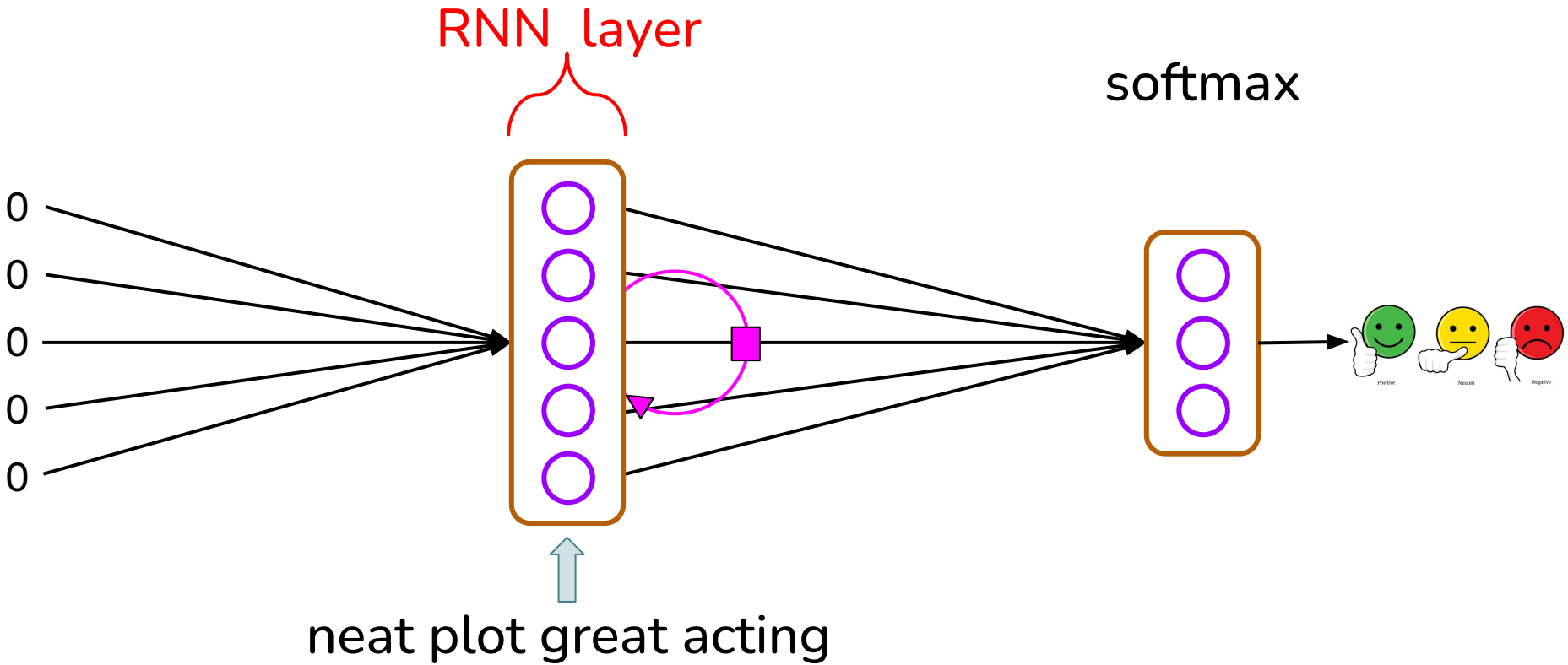


Neat plot. Great acting.

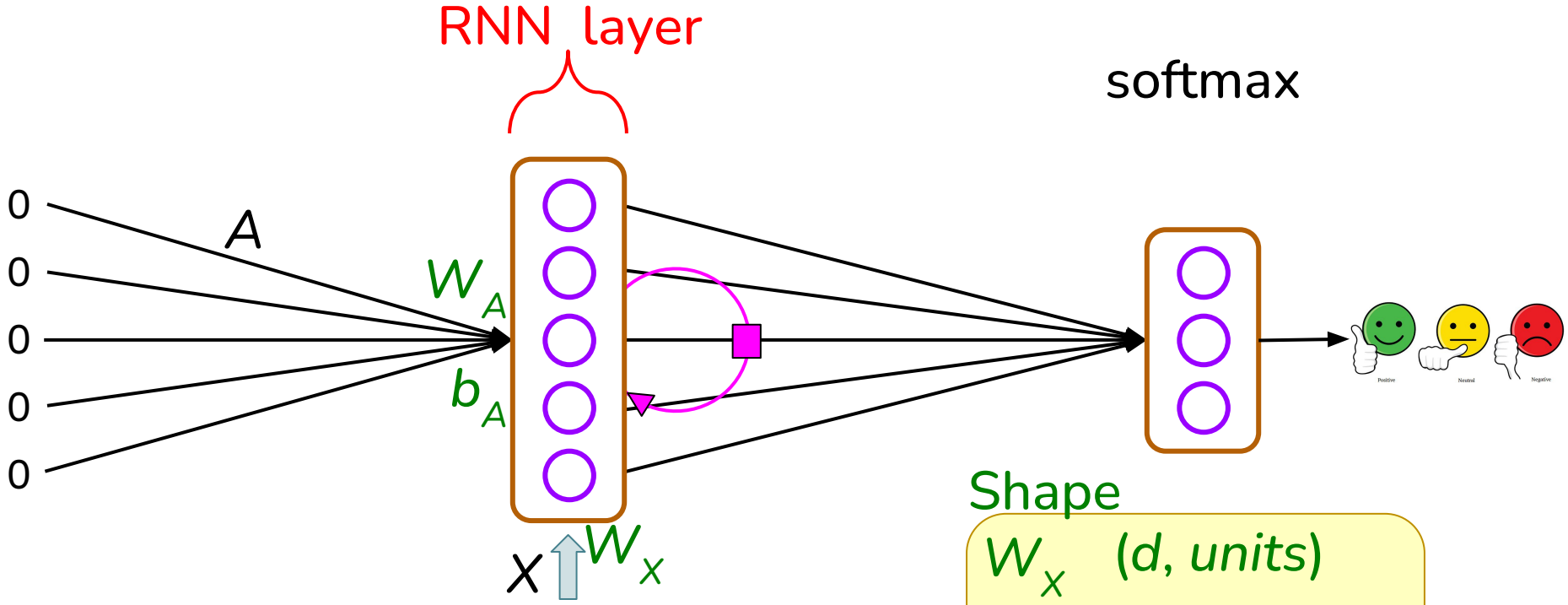
RNN Layer


RNN layer
softmax
0
0
0
0
0
neat
plot
great
acting
Positive
Neutral
Negative

RNN Layer


RNN layer
softmax
0
0
0
0
0
neat plot great acting
Positive
Neutral
Negative

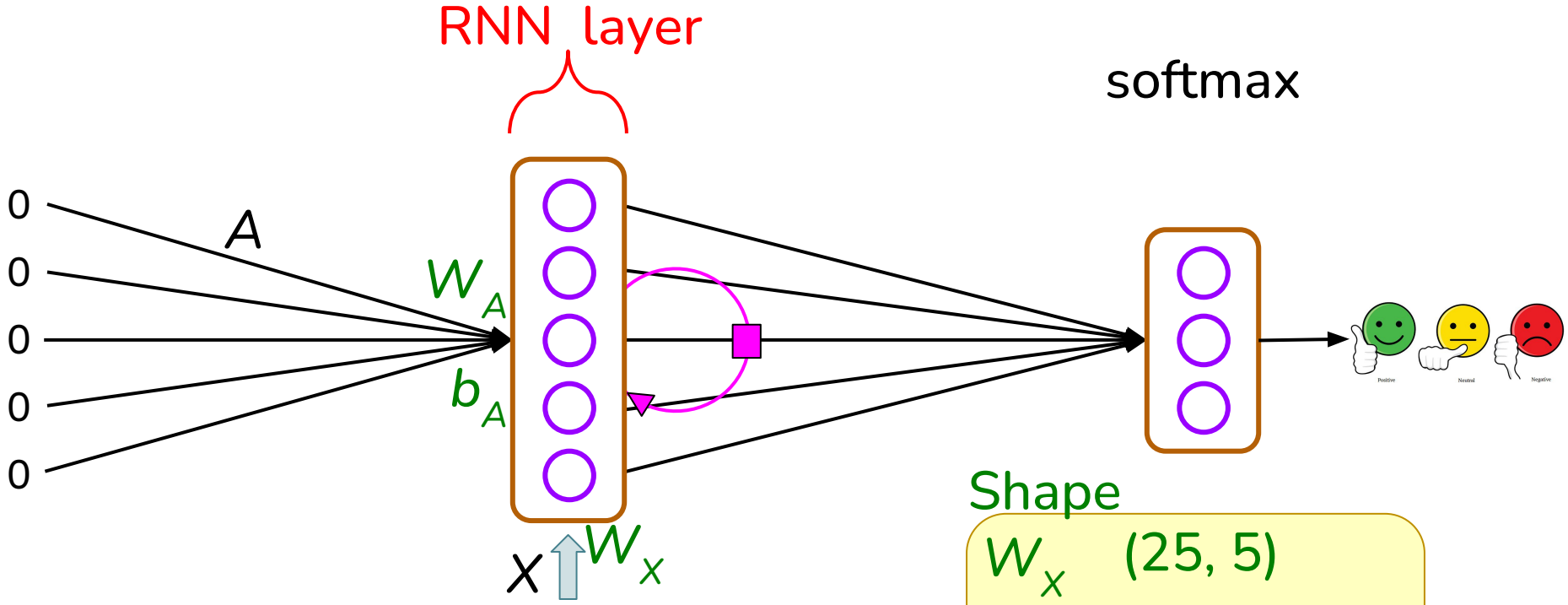
RNN Layer *Parameters*



Shape

W_X	(*d*, *units*)
W_A	(*units*, *units*)
b_A	(*units*,)

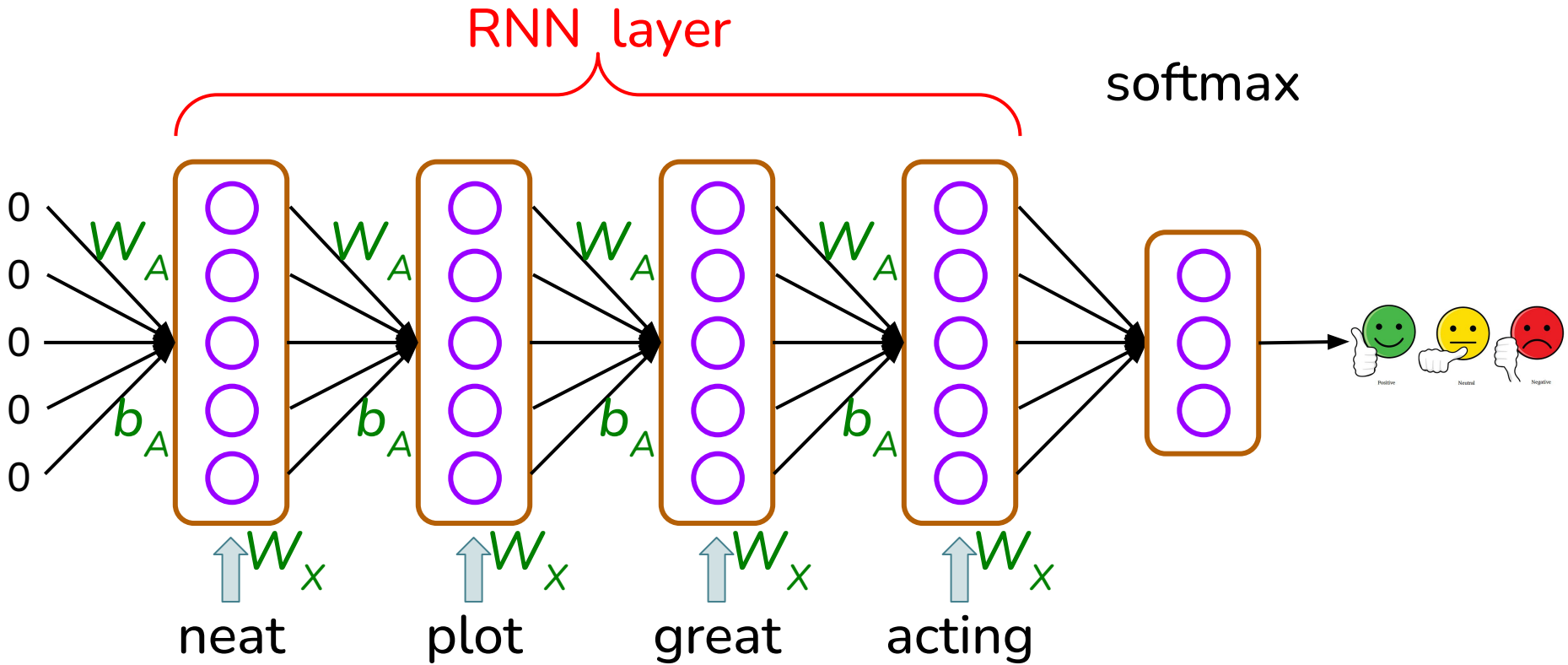
RNN Layer *Parameters*



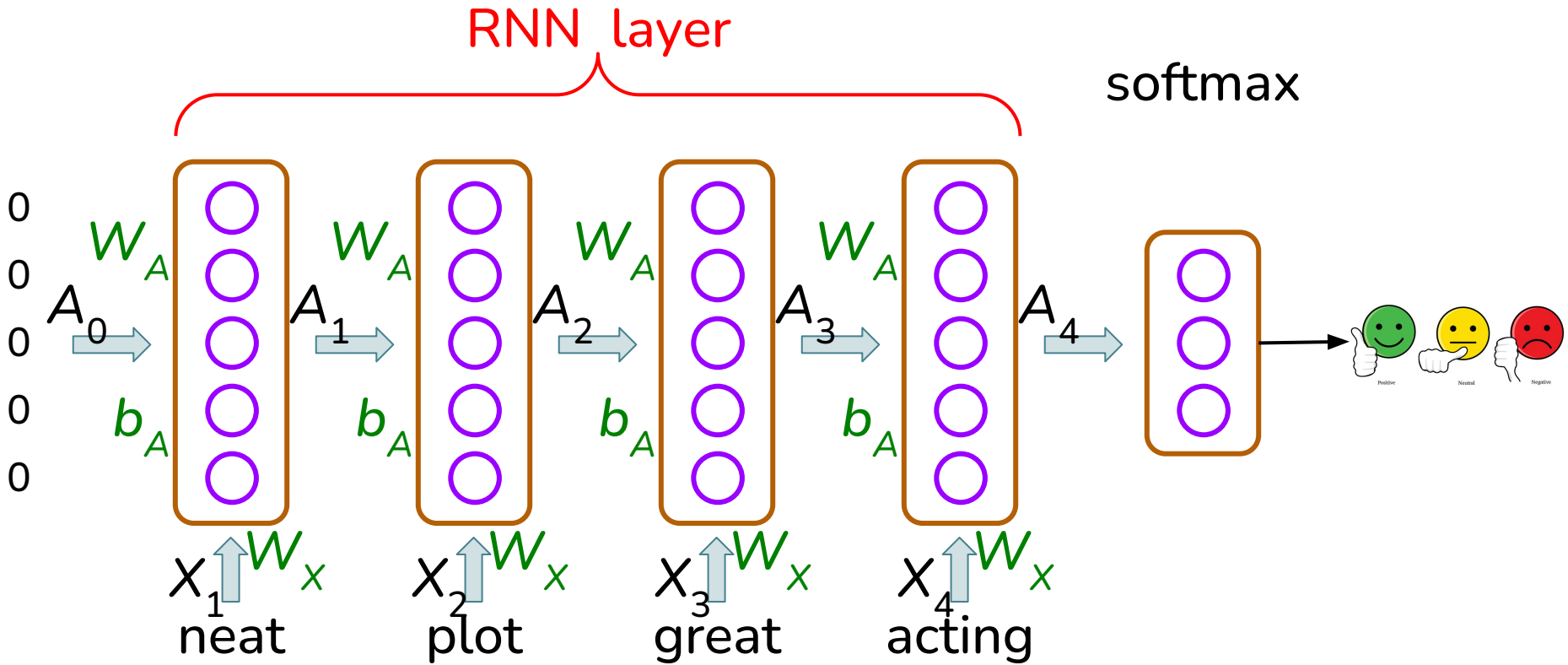
Shape

W_X	(25, 5)
W_A	(5, 5)
b_A	(5,)

RNN Layer *Parameters*



RNN Layer *Parameters*



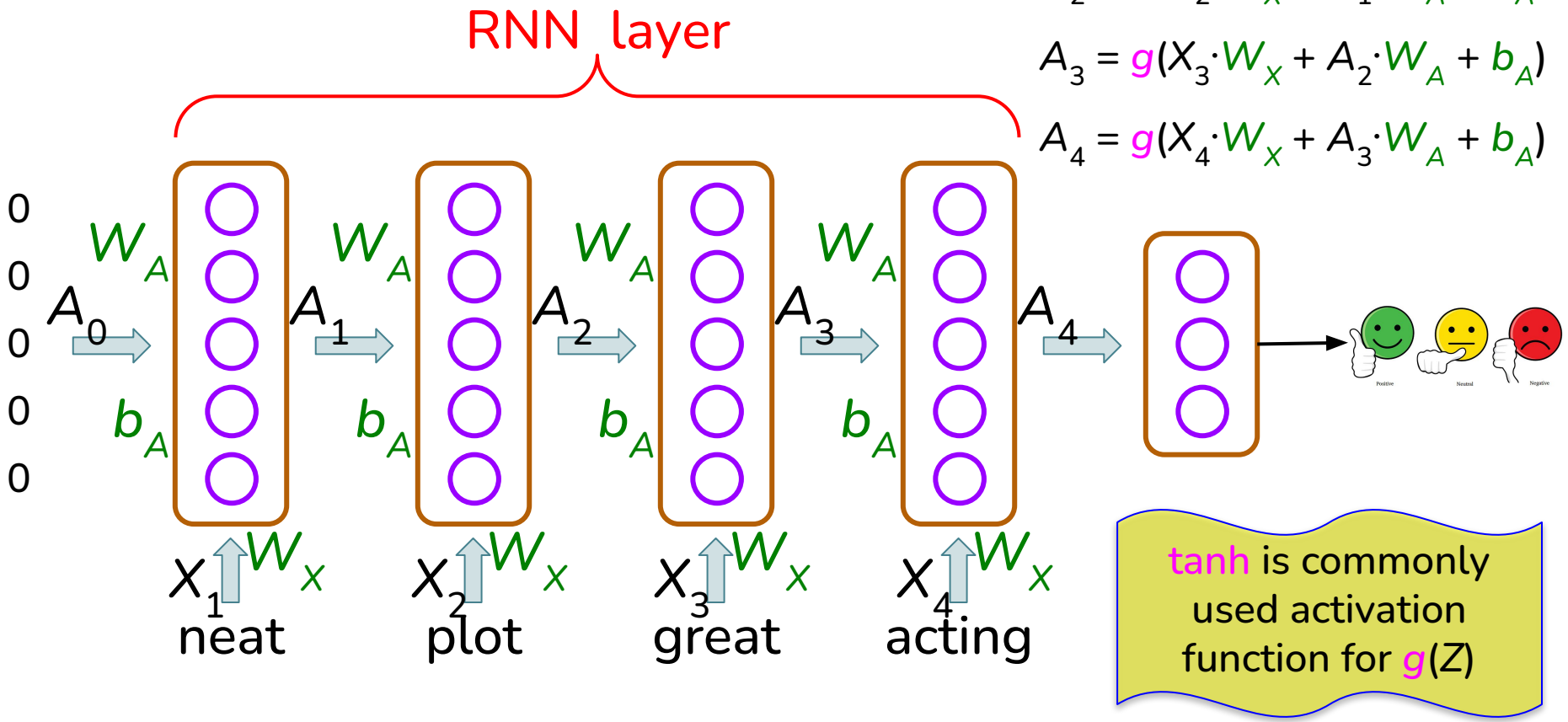
RNN Layer *Parameters*

$$A_1 = g(X_1 \cdot W_X + A_0 \cdot W_A + b_A)$$

$$A_2 = g(X_2 \cdot W_X + A_1 \cdot W_A + b_A)$$

$$A_3 = g(X_3 \cdot W_X + A_2 \cdot W_A + b_A)$$

$$A_4 = g(X_4 \cdot W_X + A_3 \cdot W_A + b_A)$$



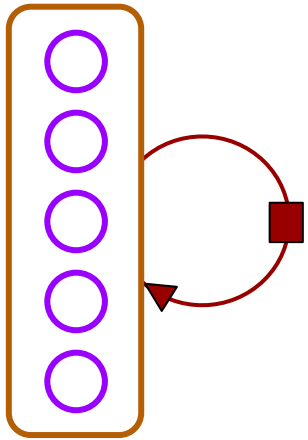
tanh is commonly used activation function for $g(Z)$

RNN Forward Propagation

Shape

W_X	(*d*, *units*)
W_A	(*units*, *units*)
b_A	(*units*,)
A	(1, *units*)

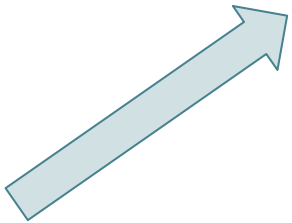
RNN layer



$A = [[0\ 0\ 0\ \ldots\ 0]]$

For $t = 0$ to T-1:

$$A = g(X_t \cdot W_X + A \cdot W_A + b_A)$$

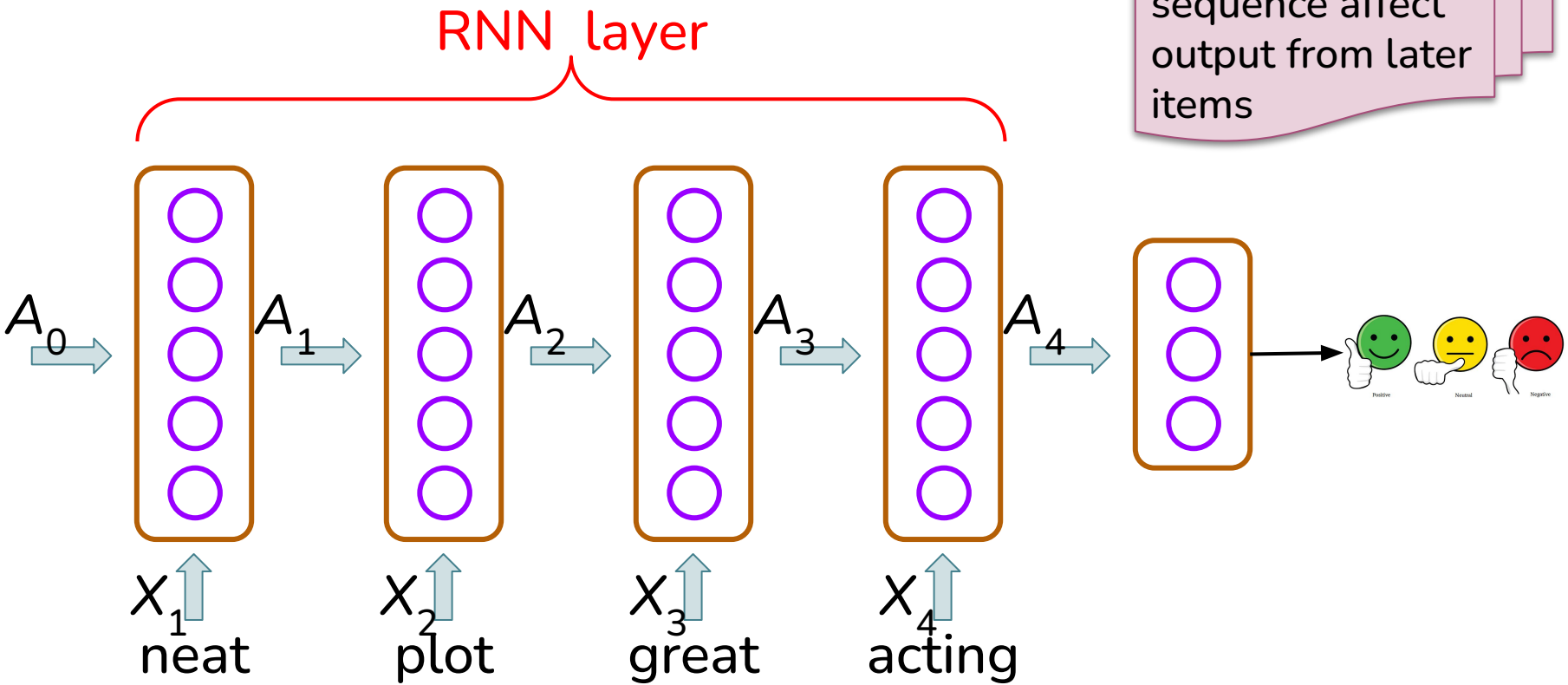


Neat plot. Great acting.

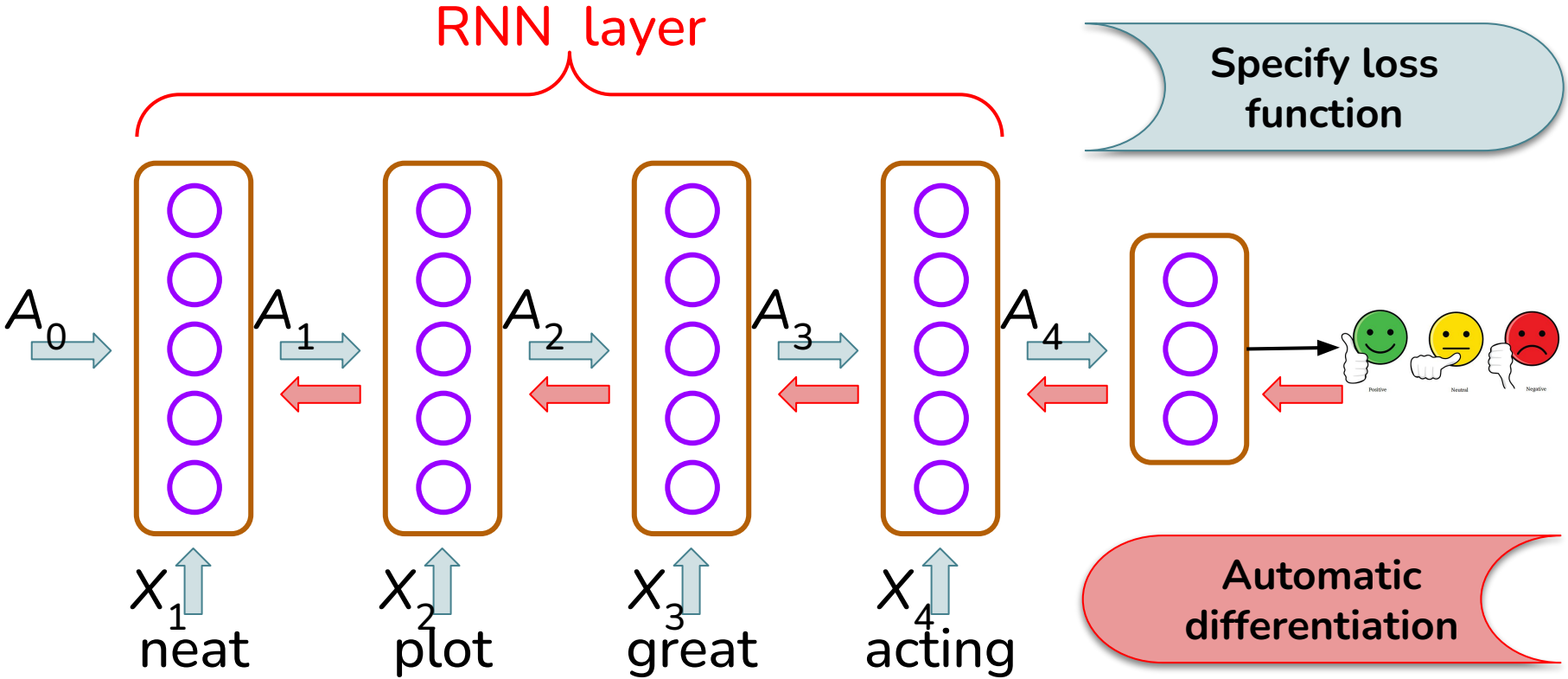
T is number of elements in sequence

RNN Forward Propagation

Early items in sequence affect output from later items



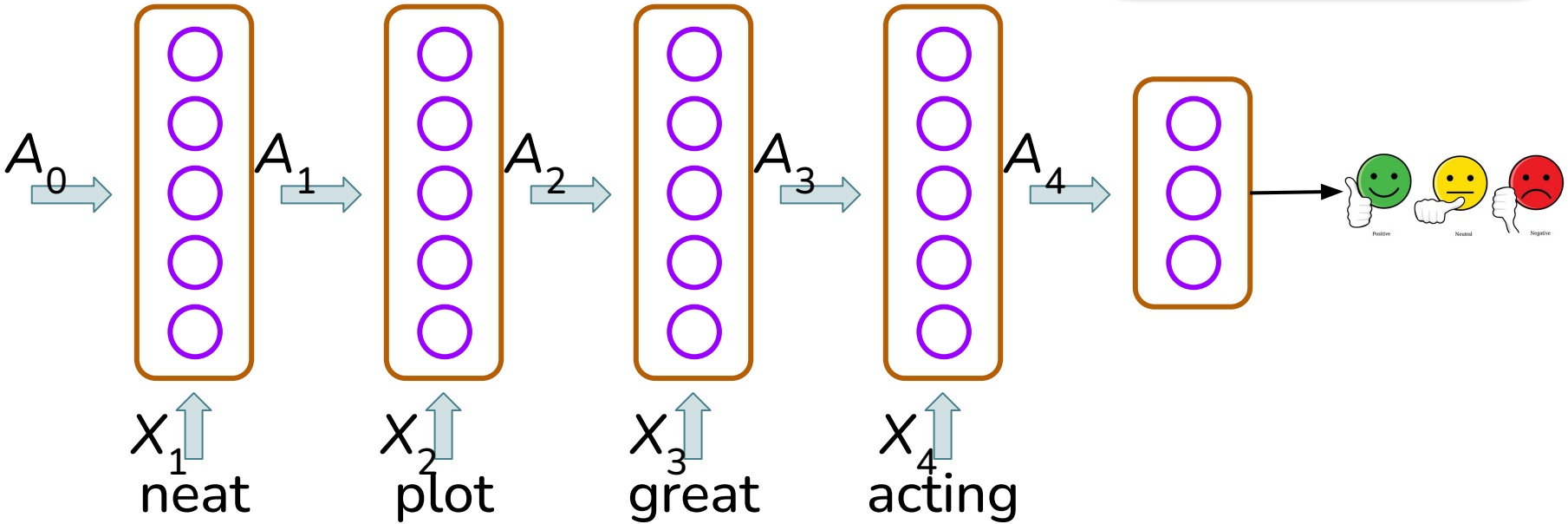
RNN Backward Propagation



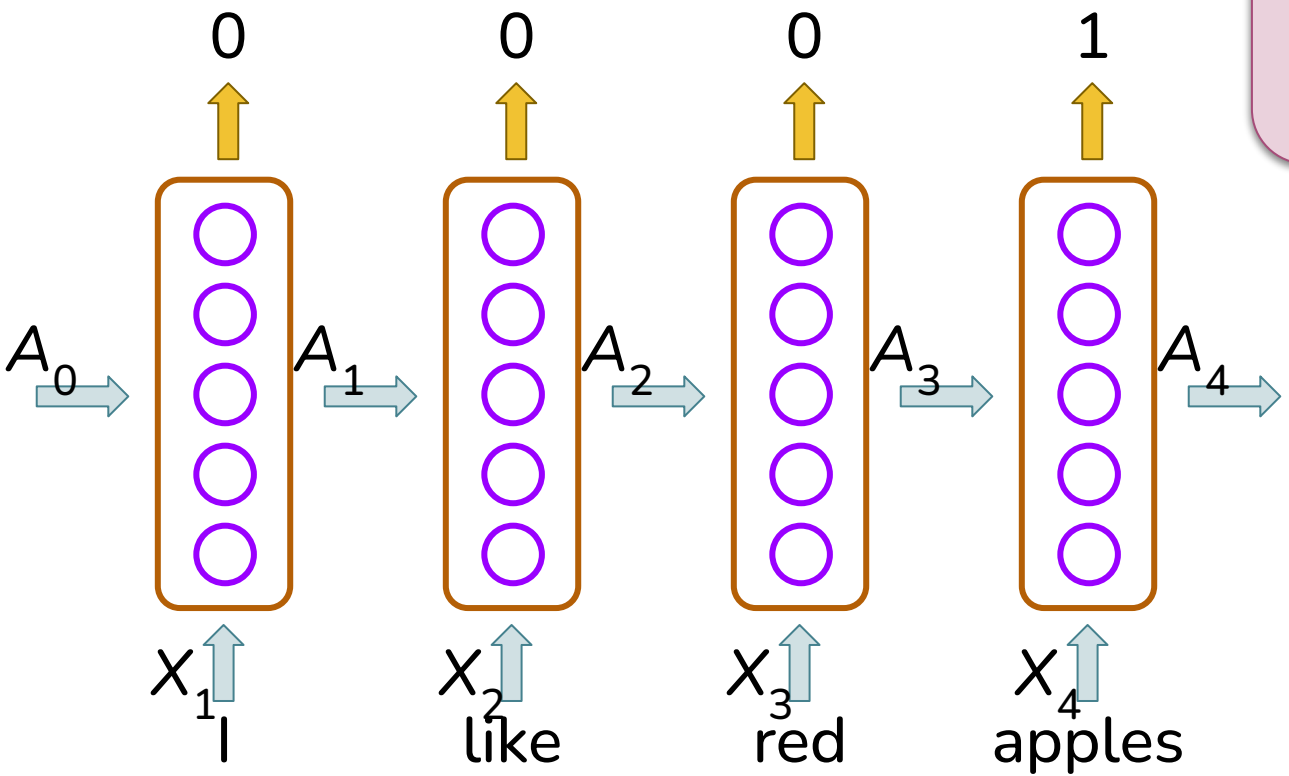
RNN Output

Sentiment Classification

Output is one value, not a sequence



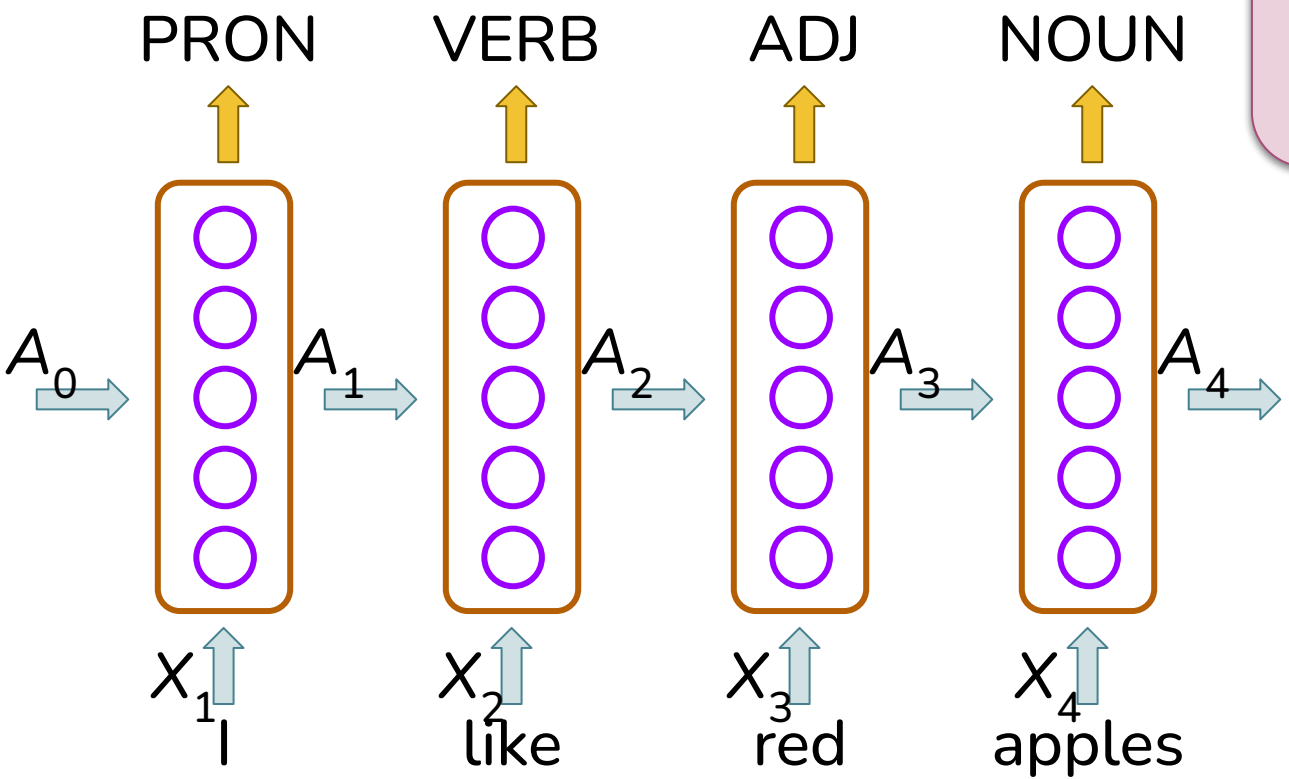
RNN Output



Word Labeling

Output is sequence, one value per input

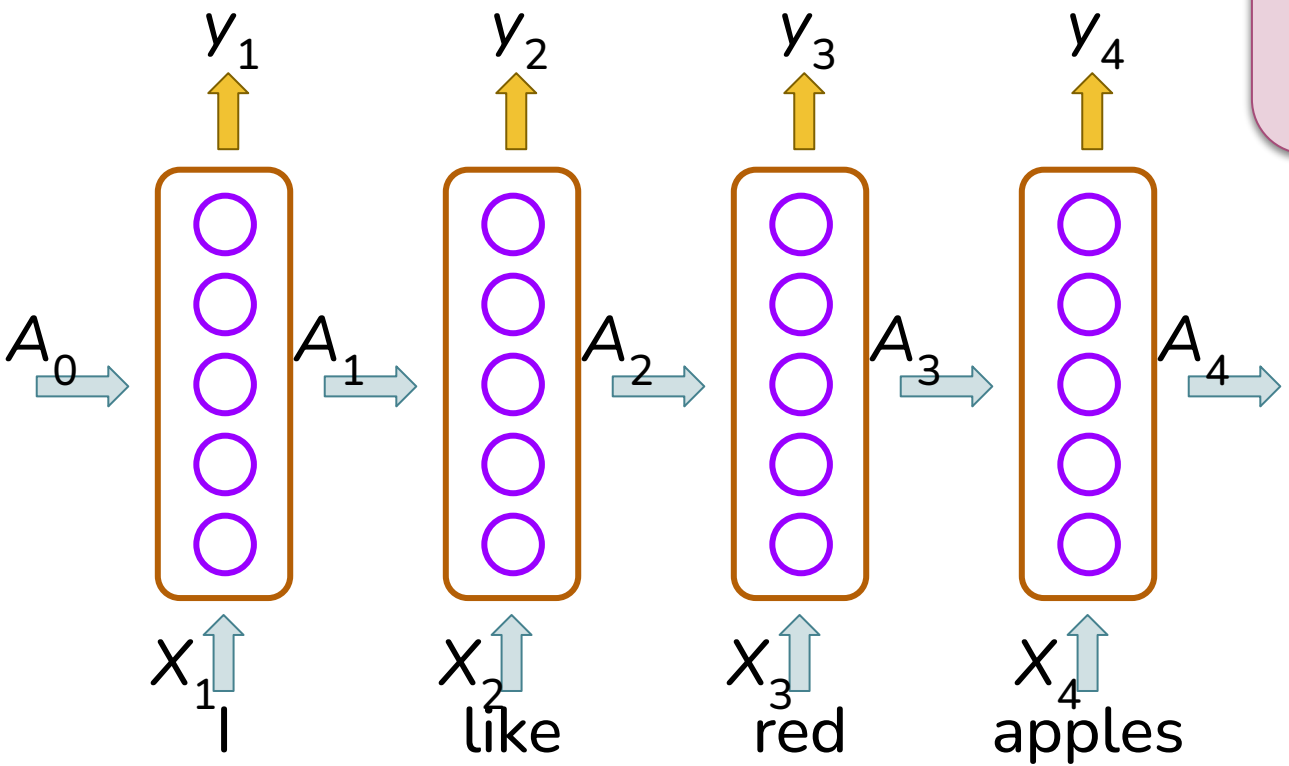
RNN Output



Word Labeling

Output is sequence, one value per input

RNN Output



Word Labeling

Output is sequence, one value per input

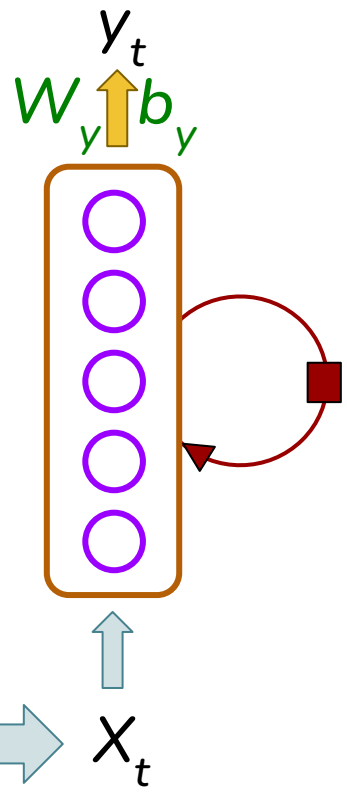
RNN Output *Parameters*

$$y_t = g(A \cdot W_y + b_y)$$

Shape

W_y (*units*, ?)

b_y (?,)



I like red apples

Activation function g depends on problem

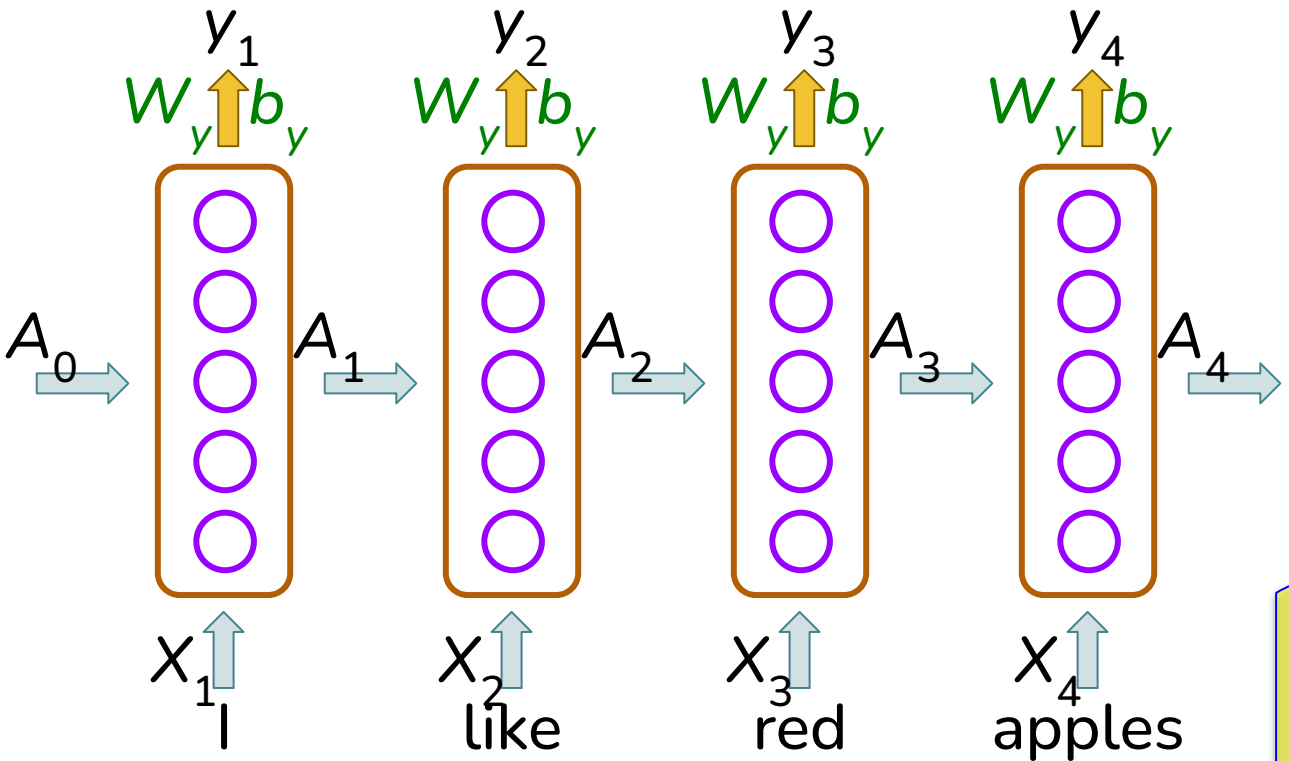
RNN Output

$$y_1 = g(A_1 \cdot W_y + b_y)$$

$$y_2 = g(A_2 \cdot W_y + b_y)$$

$$y_3 = g(A_3 \cdot W_y + b_y)$$

$$y_4 = g(A_4 \cdot W_y + b_y)$$

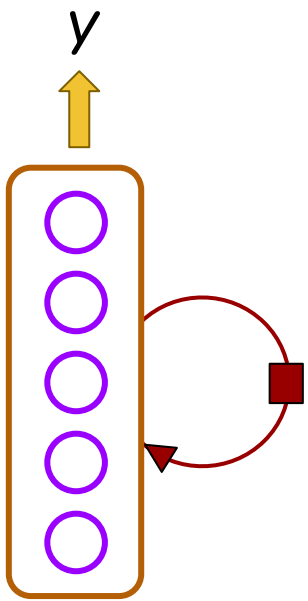


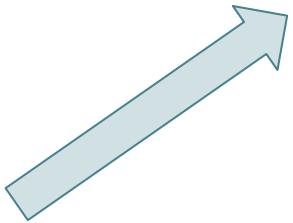
Activation function g depends on problem

RNN Forward Propagation

Shape

W_X	(*d*, *units*)
W_A	(*units*, *units*)
b_A	(*units*,)
W_y	(*units*, ?)
b_y	(?,)





I like red apples

A = [[0 0 0 … 0]] # *units*

y = [0 0 … 0] # *T*

For t = 0 to T-1:

$$A = g(X_t \cdot W_X + A \cdot W_A + b_A)$$

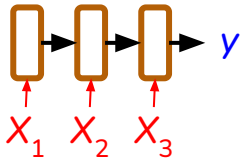
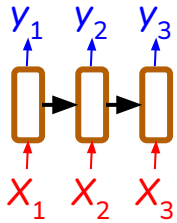
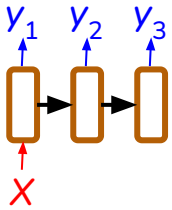
$$y_t = g(A \cdot W_y + b_y)$$

tanh

sigmoid, softmax, linear

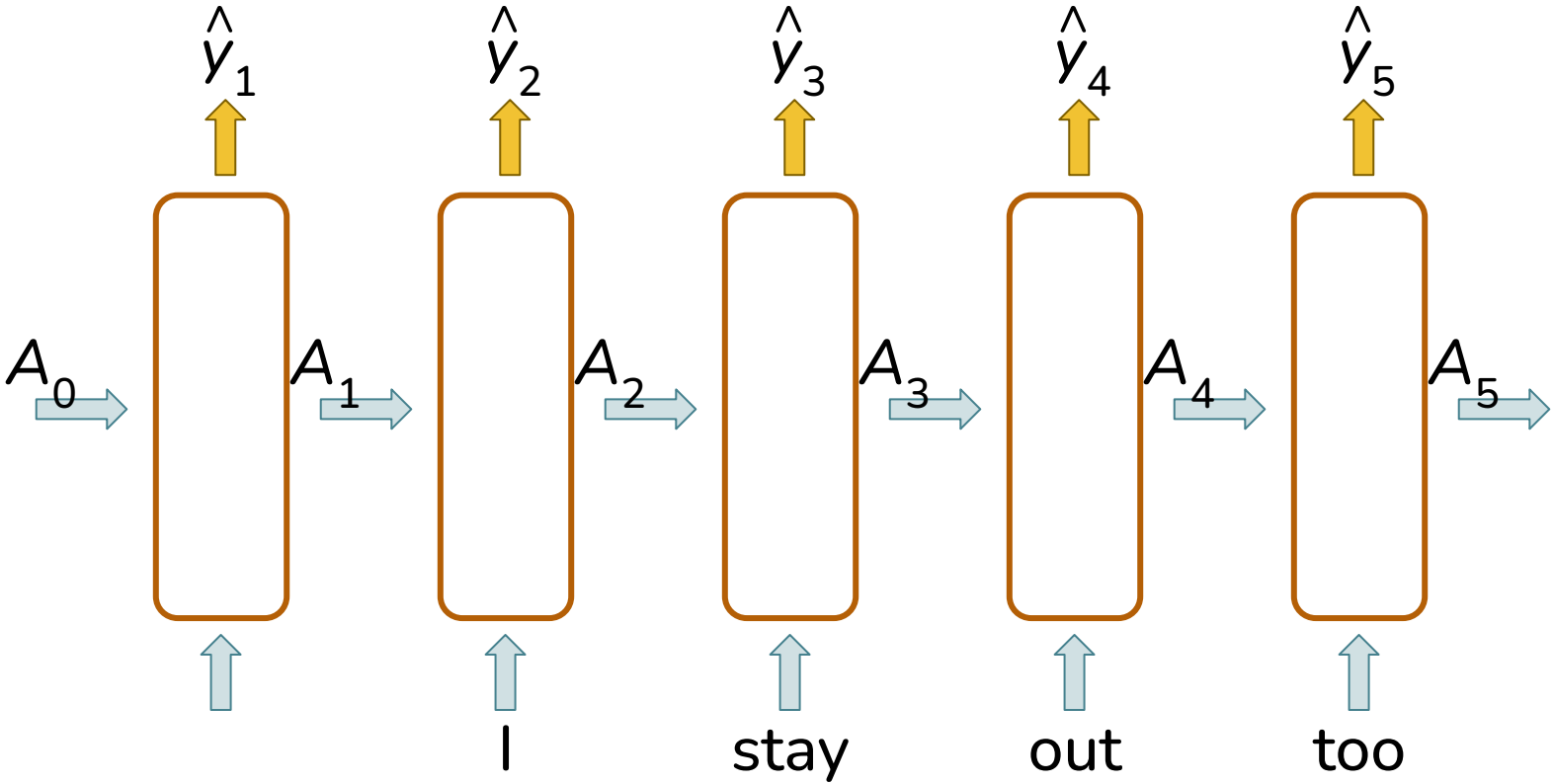
T is number of elements in sequence

Different RNNs

Input	Output	Example	Architecture
Sequence	Non-sequence	Sentiment classification	
Sequence	Sequence (same-length)	Word labeling	
Non-sequence	Sequence	Text generation	
Sequence	Sequence (different-length)	Translation	

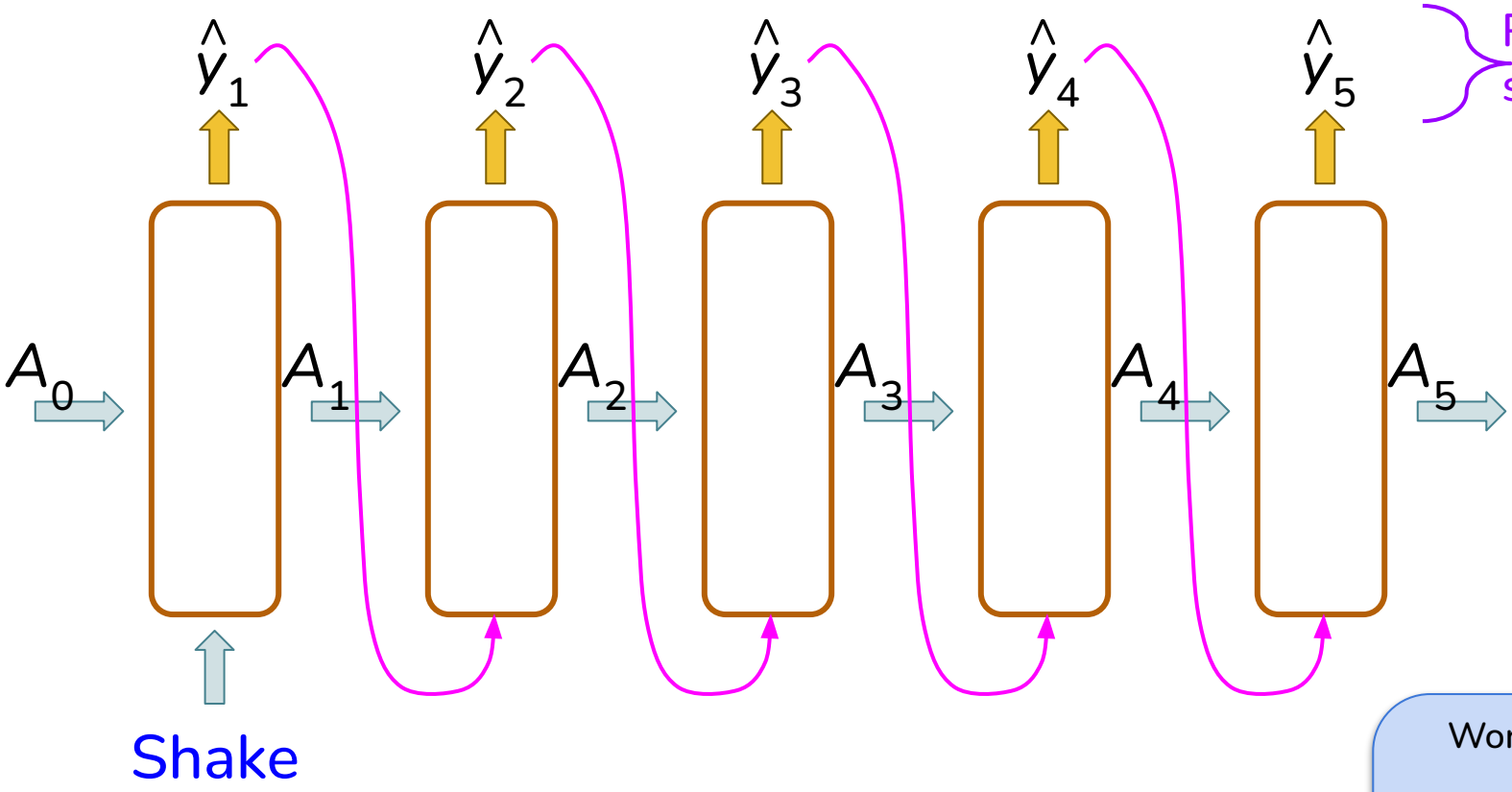
Text Generation: Training

Specify loss function



I stay out too late

Text Generation: Generation



Word-based
or
Character-based