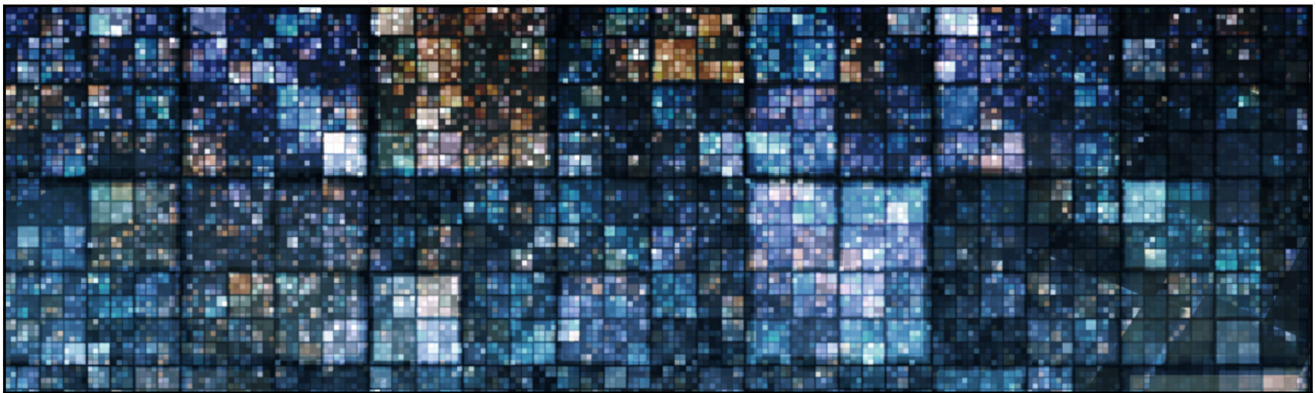


COMP47590

Advanced Machine Learning

Deep Learning – Recurrent Networks

Dr. Brian Mac Namee

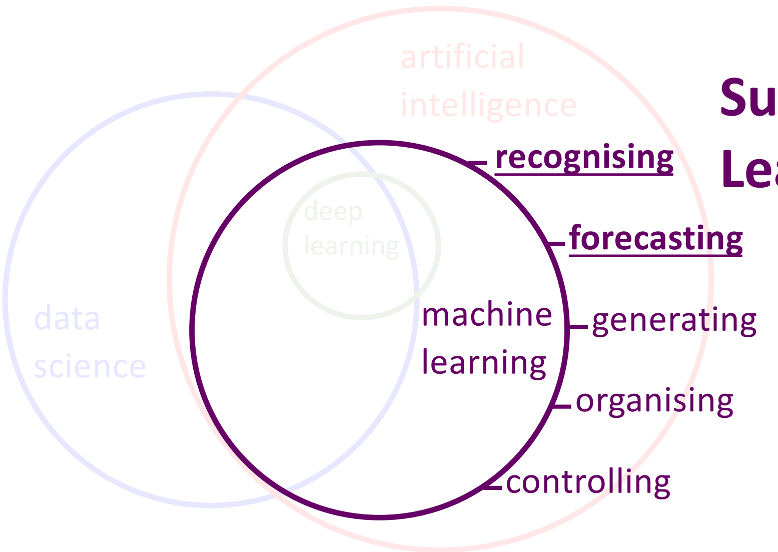
Information

Email: Brian.MacNamee@ucd.ie

Course Materials: All material posted on UCD CS moodlehttps://csmoodle.ucd.ie/moodle/course/view.php?id=663

Enrolment key **UCDAdvML2017**

Supervised Learning

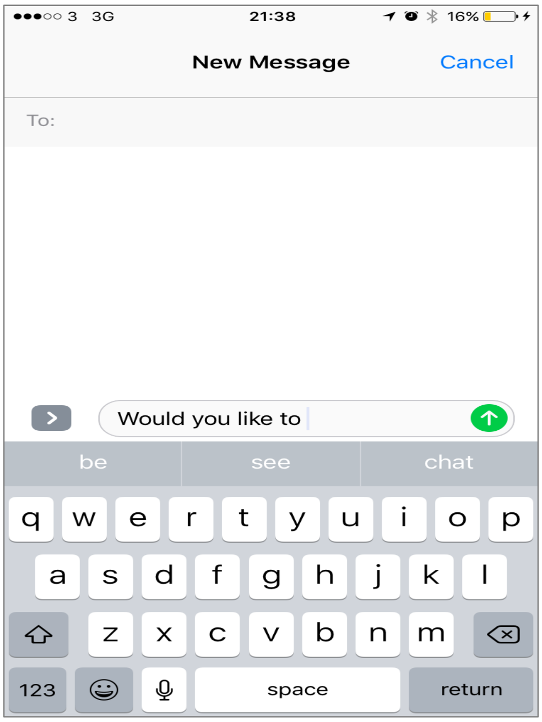


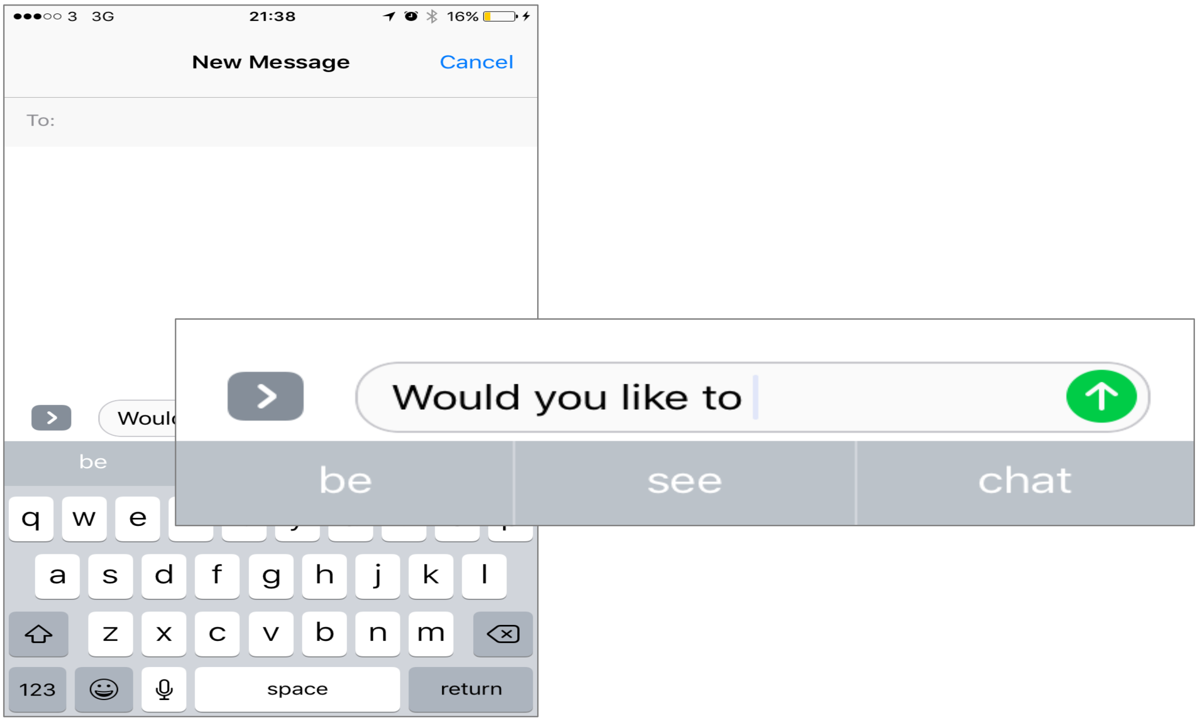
Section Outline

In this section we will cover:

- Word embeddings
- Recurrent neural networks
- Unrolling RNNs
- Different kinds of RNNs

WORD VECTOR EMBEDDING ORIGINS





Learning Language Models

i am feeling __________

the weather today is really __________

would you like to play a __________

are you going for a __________

Learning Language Models

i am feeling __happy___

the weather today is really __good___

would you like to play a __game___

are you going for a __pint___

Learning Language Models

i am feeling __happy___

the weather today is really __good___

would you like to play a __game___

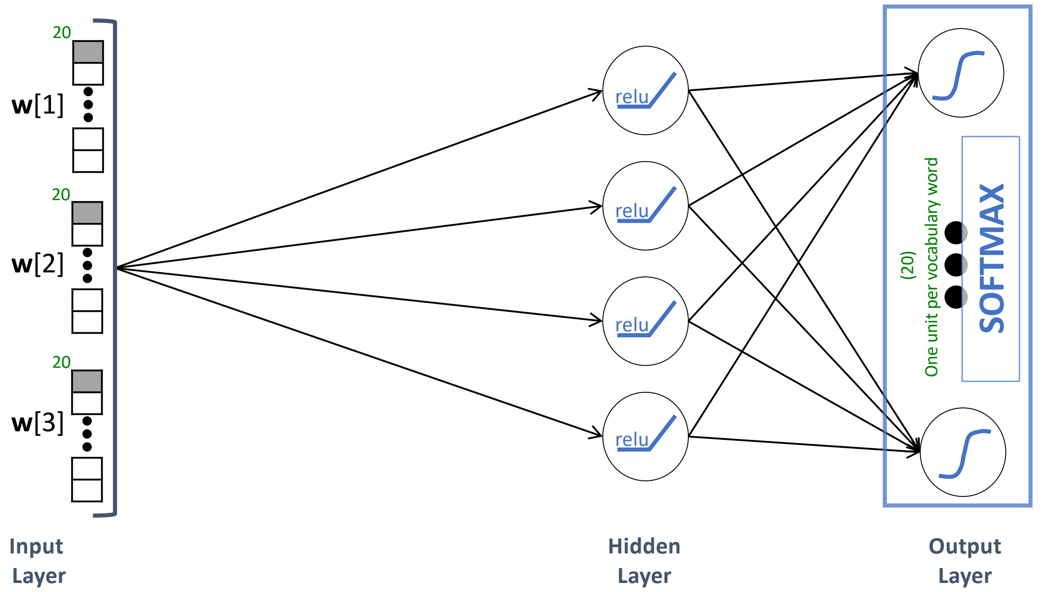
are you going for a __pint___

Learning Language Models

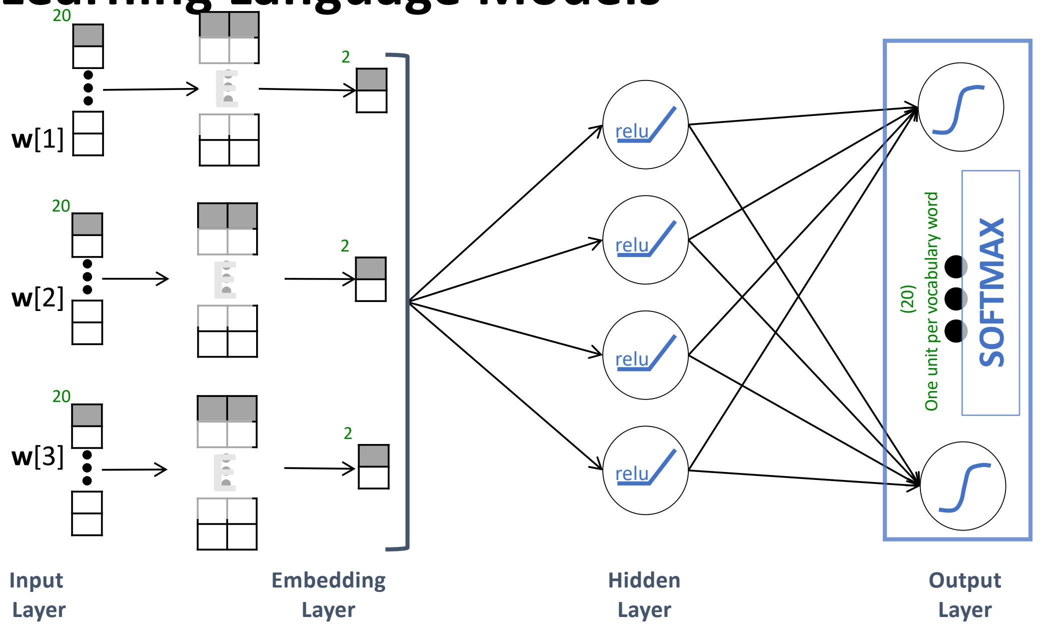
a
am
feeling
for
game
going
good
happy
i
is
like
pint
play
really
the
to
today
weather
would
you

		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	a	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	am	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	feeling	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	for	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	game	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	going	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	good	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
7	happy	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
8	i	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
9	is	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
10	like	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
11	pint	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
12	play	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
13	really	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
14	the	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
15	to	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
16	today	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
17	weather	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
18	would	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
19	you	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

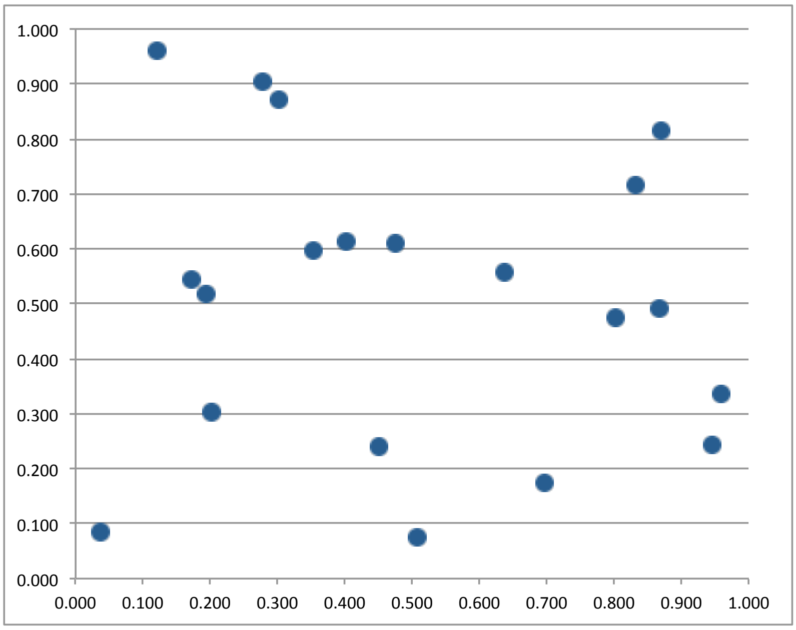
Learning Language Models



Learning Language Models



		0	1
0	a	0.944	0.243
1	am	0.832	0.717
2	feeling	0.403	0.612
3	for	0.302	0.870
4	game	0.868	0.493
5	going	0.476	0.611
6	good	0.802	0.476
7	happy	0.038	0.084
8	i	0.638	0.559
9	is	0.697	0.176
10	like	0.870	0.814
11	pint	0.506	0.076
12	play	0.202	0.303
13	really	0.277	0.903
14	the	0.960	0.335
15	to	0.450	0.239
16	today	0.352	0.598
17	weather	0.121	0.959
18	would	0.173	0.543
19	you	0.194	0.518



WORD VECTOR EMBEDDING ALGORITHM

Word Vector Embeddings: word2vec (Skip-gram)

The word2vec algorithm is still a very popular approach to building word embeddings

- Build a model to predict target word based on a context
- Use context of *nearby* words (skip-gram)

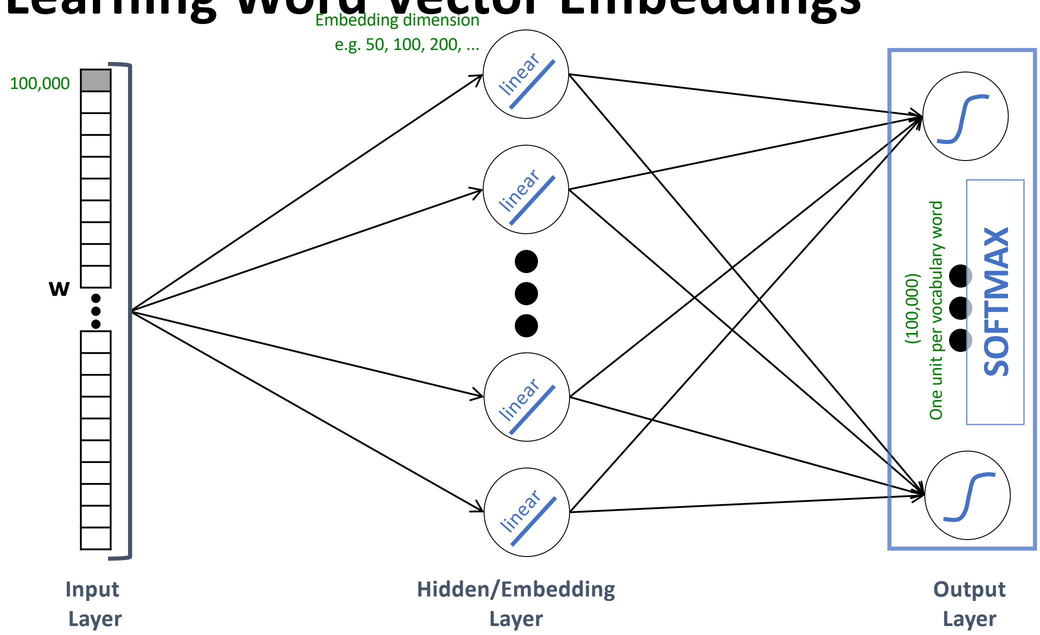
Mikolov et al, "Efficient Estimation of Word Representations in Vector Space", 2013
https://arxiv.org/pdf/1301.3781.pdf

Word Vector Embeddings: word2vec (Skip-gram)

Extend the language model idea to any target-context pairs

the weather today is really good

Learning Word Vector Embeddings

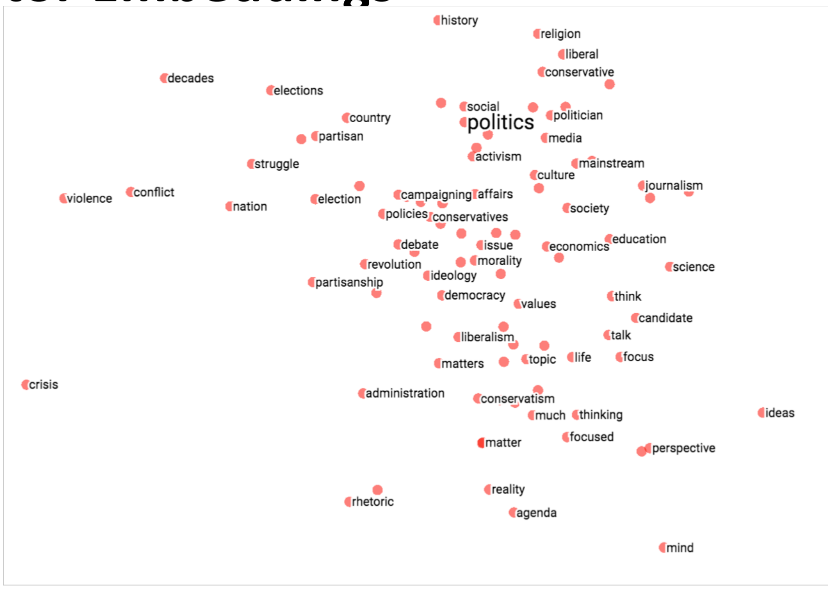


Training word2vec (Skip-gram)

Training word2vec embeddings is reasonably straight-forward

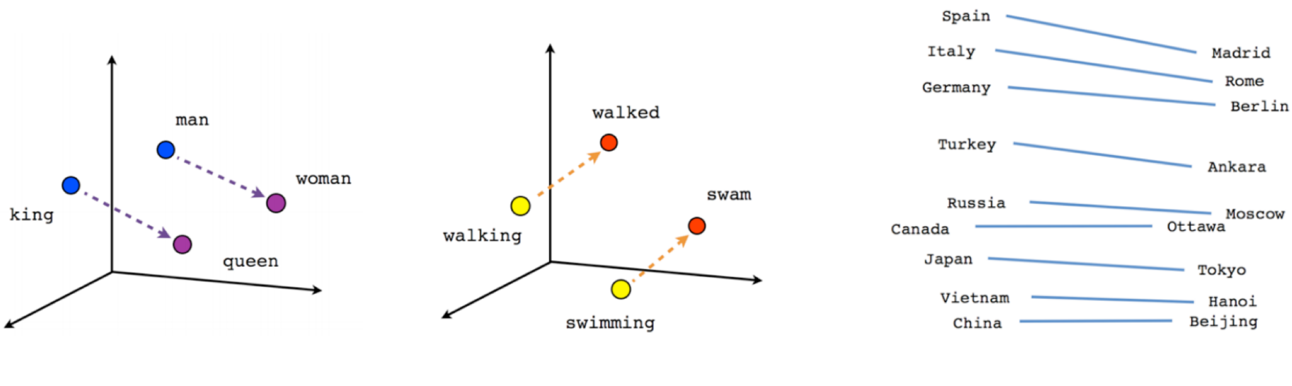
- Generate a massive set of target-context pairs from a large corpus
- Use backpropagation of error algorithm to minimise cross-entropy loss
- Extract set of embeddings for vocabulary words

Word Vector Embeddings



https://www.tensorflow.org/programmers_guide/embedding

Word Vector Embedding Algebra



Male-Female

Verb tense

Country-Capital

Word Vector Embeddings

There are lots of extensions to the word2vec approach

- Use **negative sampling** to introduce counter examples into training
- Use word2vec cbow to predict target based on context instead of vice versa
- Use hierarchical models to reduce computation

There are alternative approaches to generating embeddings, e.g. glove

USING WORD EMBEDDINGS FOR TEXT CLASSIFICATION

Word Vector Embeddings

We can either use a set of pre-trained word embeddings or generate word embeddings of our own

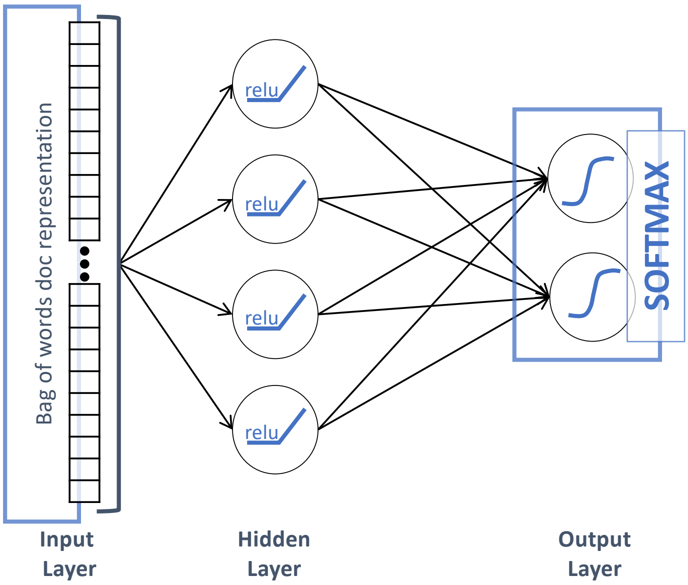
- Training word vector embeddings across very large datasets takes a lot of computation which is avoided by pre-computed embeddings
- Pre-computed embeddings may be out of date and may not suit specialised vocabulary
- Using a pre-trained word-embedding can amplify a smaller dataset for an ML task

Text Classification With Word Vector Embeddings

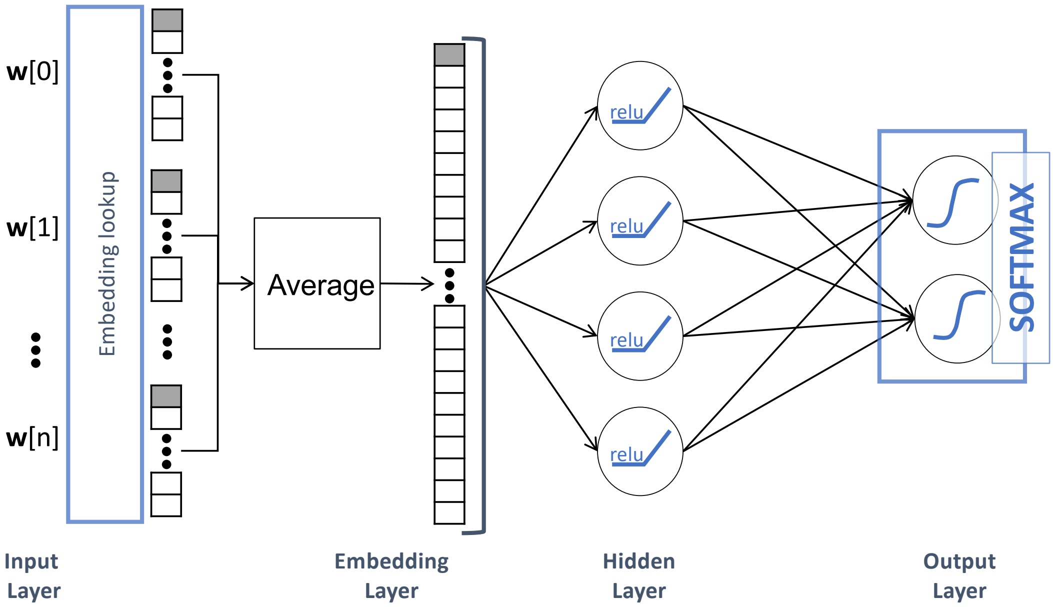
Let's imagine a nice spam classification problem

hi brian, are you going to the meeting?	ham
hi, are you lonely tonight?	spam
bitcoin bitcoin bitcoin	spam
what do you want for dinner?	ham
the match starts at four	ham
respected author, will you publish with us?	spam

Learning Language Models



Learning Language Models



Recurrent Neural Networks

Recurrent Neural Networks

Many datasets that we work with feature sequence relationships between instances

- Speech data
- Sensor readings
- Financial indicaotrs
- DNA sequences
- Text strings

Recurrent neural networks take advantage of these relationships

Recurrent Neural Networks

Let's consider a simple text processing problem - named entity recognition

the	game	was	beautifully	managed	by	porter

henshaw	and	marsh	offered	a	defensive	masterclass

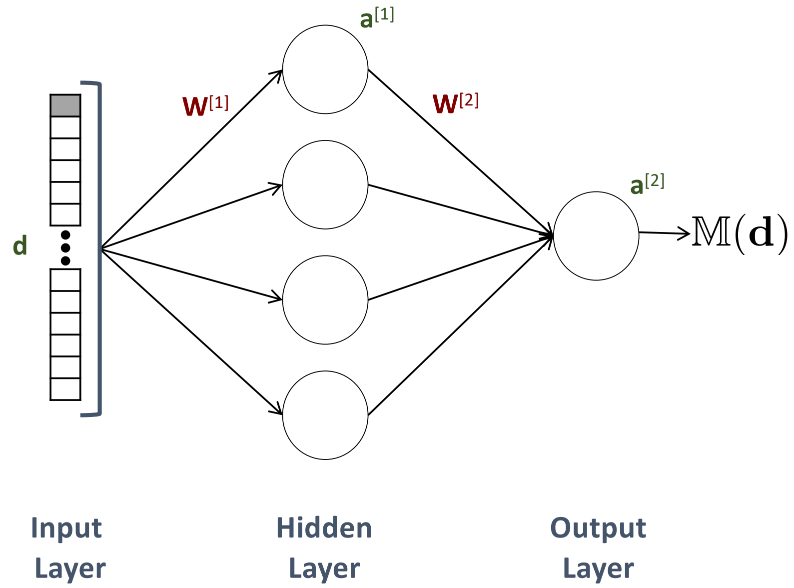
Recurrent Neural Networks

Let's consider a simple text processing problem - named entity recognition

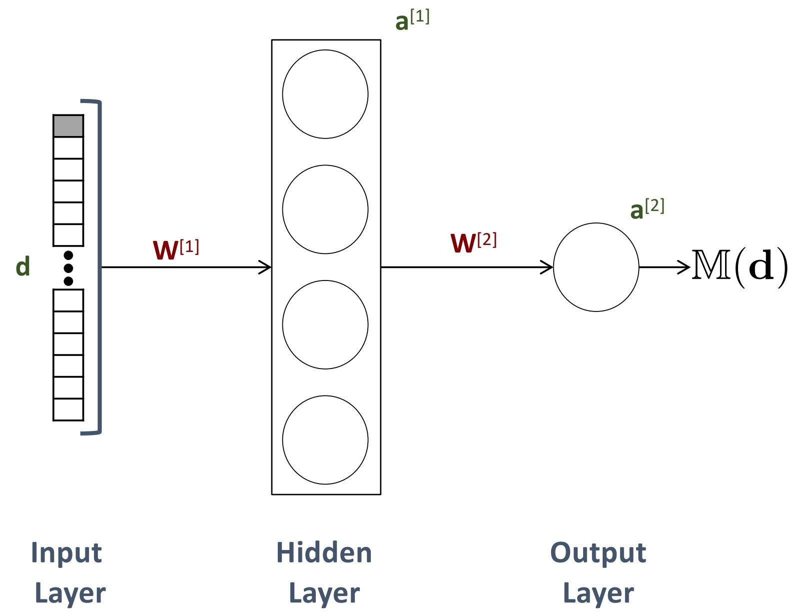
the	game	was	beautifully	managed	by	porter
0	0	0	0	0	0	1

henshaw	and	marsh	offered	a	defensive	masterclass
1	0	1	0	0	0	0

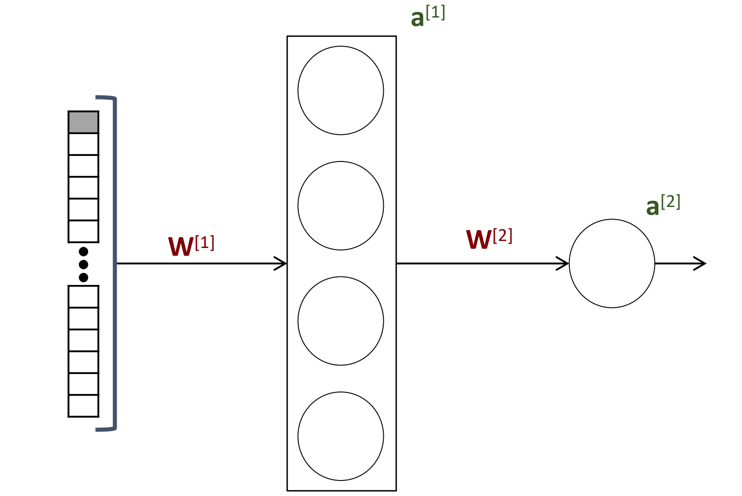
Recurrent Neural Networks



Recurrent Neural Networks

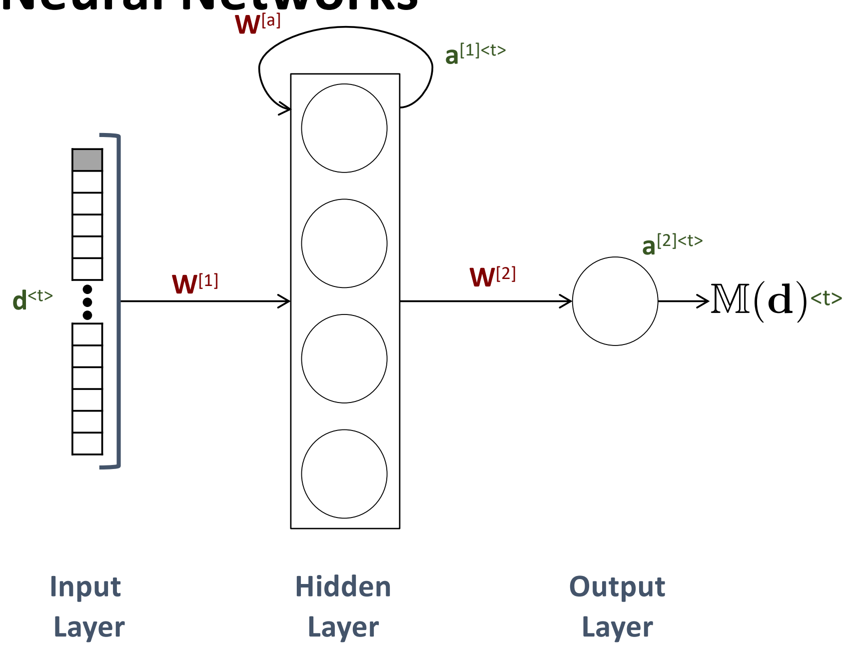


Recurrent Neural Networks

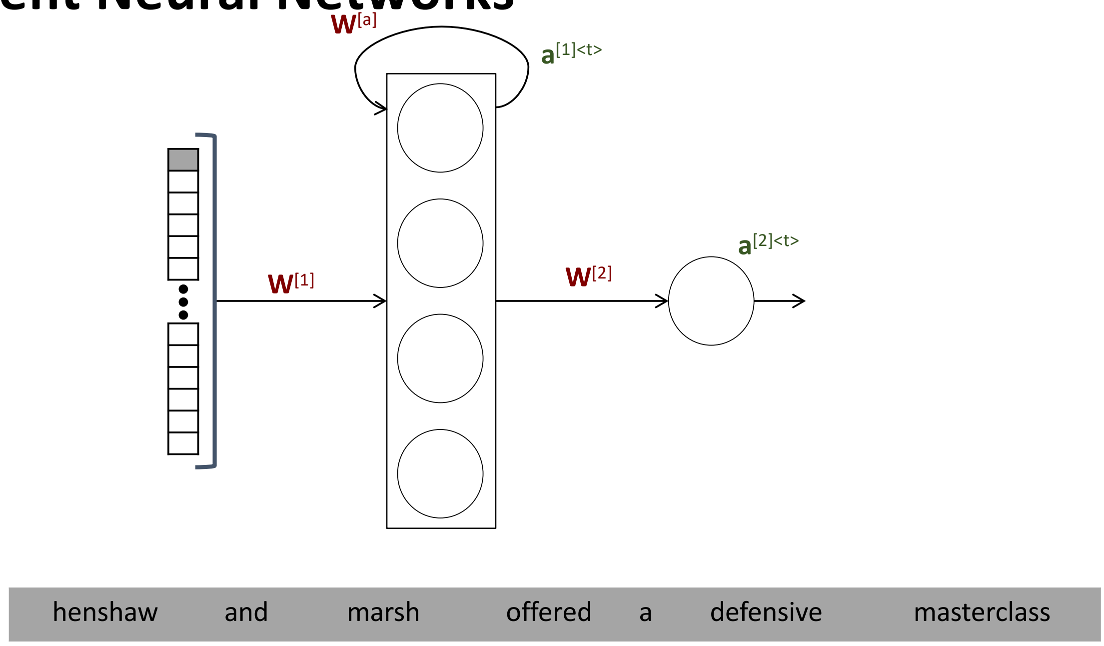


henshaw and marsh offered a defensive masterclass

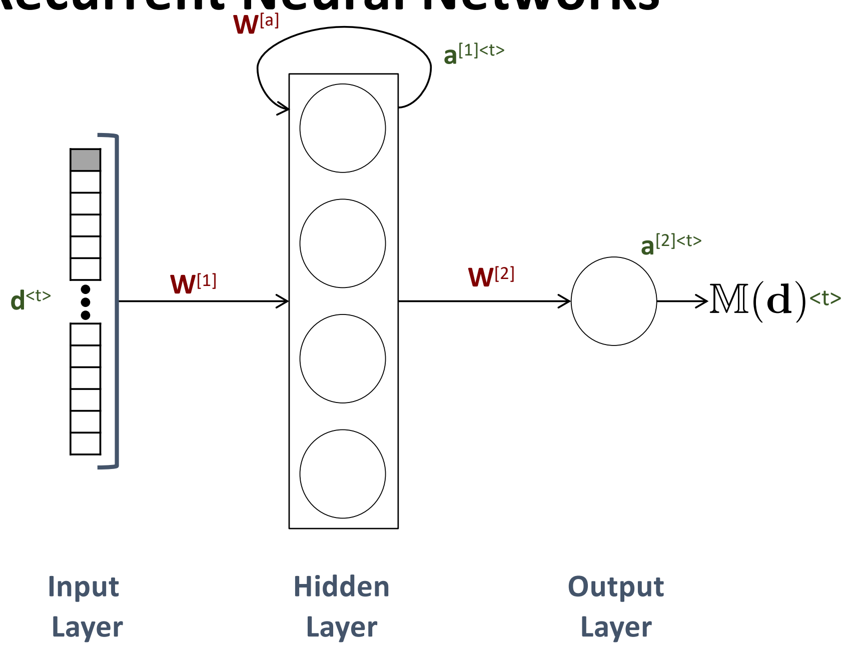
Recurrent Neural Networks



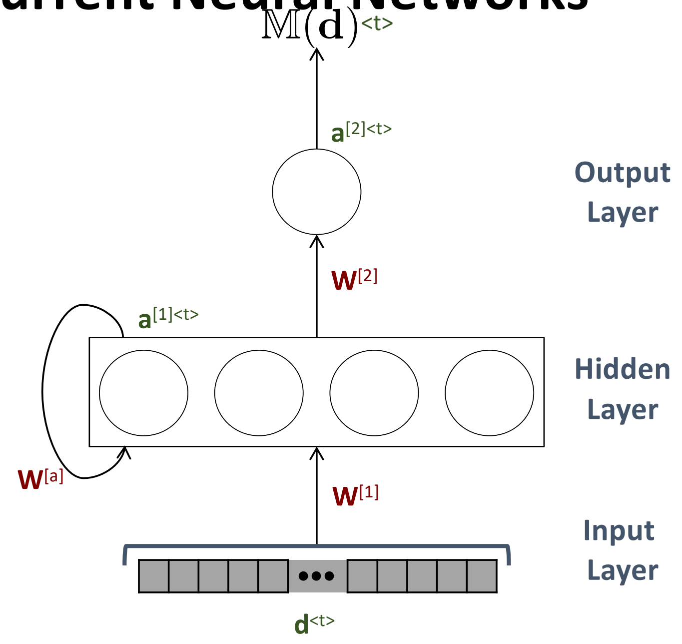
Recurrent Neural Networks



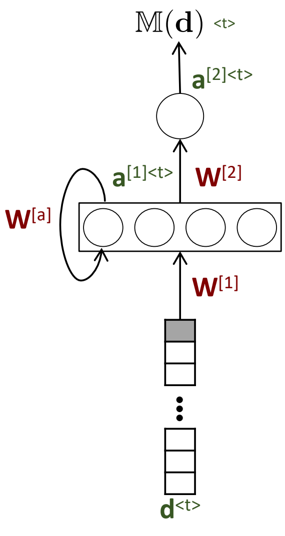
Unrolling Recurrent Neural Networks



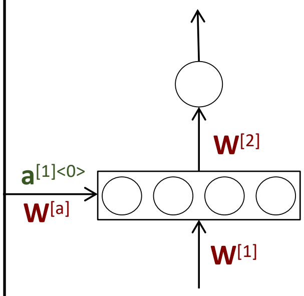
Unrolling Recurrent Neural Networks



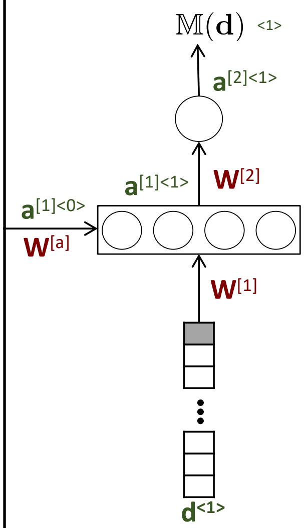
Unrolling Recurrent Neural Networks



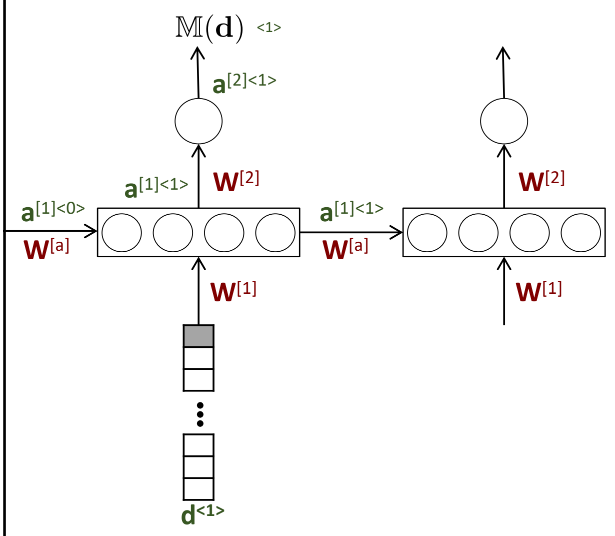
Unrolling Recurrent Neural Networks



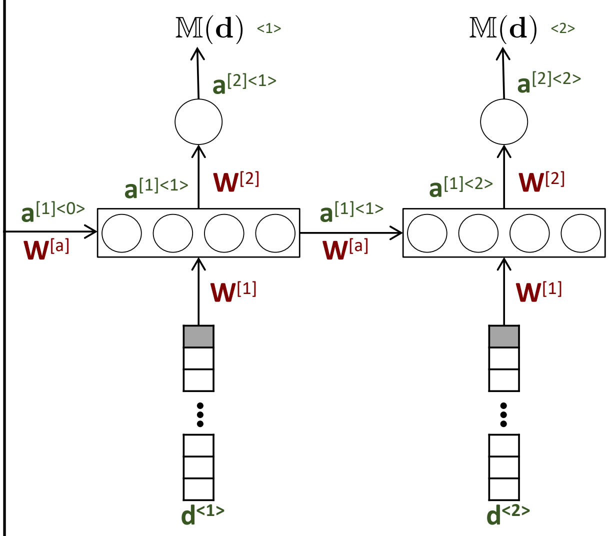
Unrolling Recurrent Neural Networks



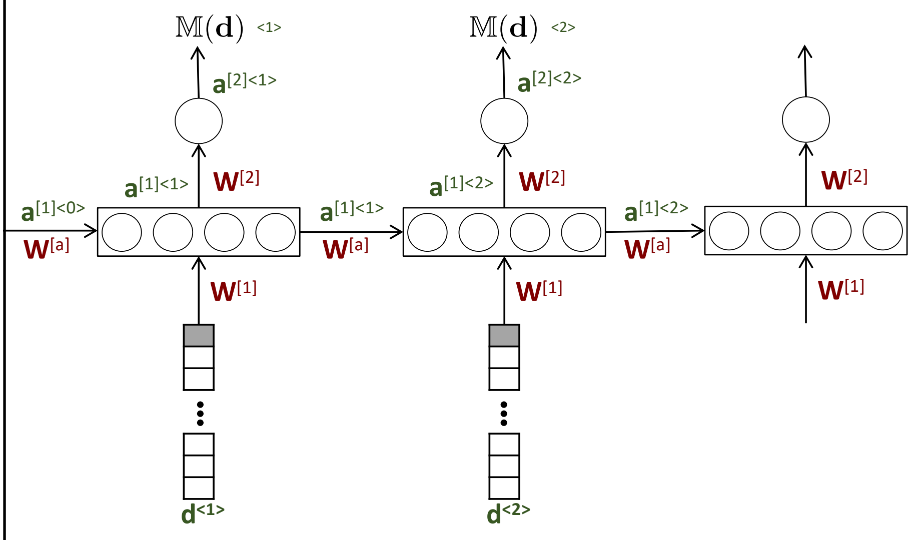
Unrolling Recurrent Neural Networks



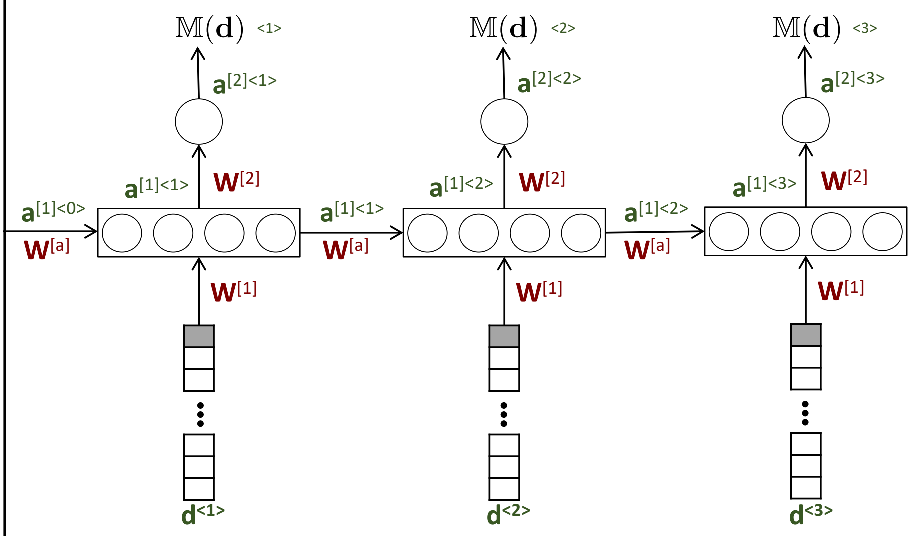
Unrolling Recurrent Neural Networks



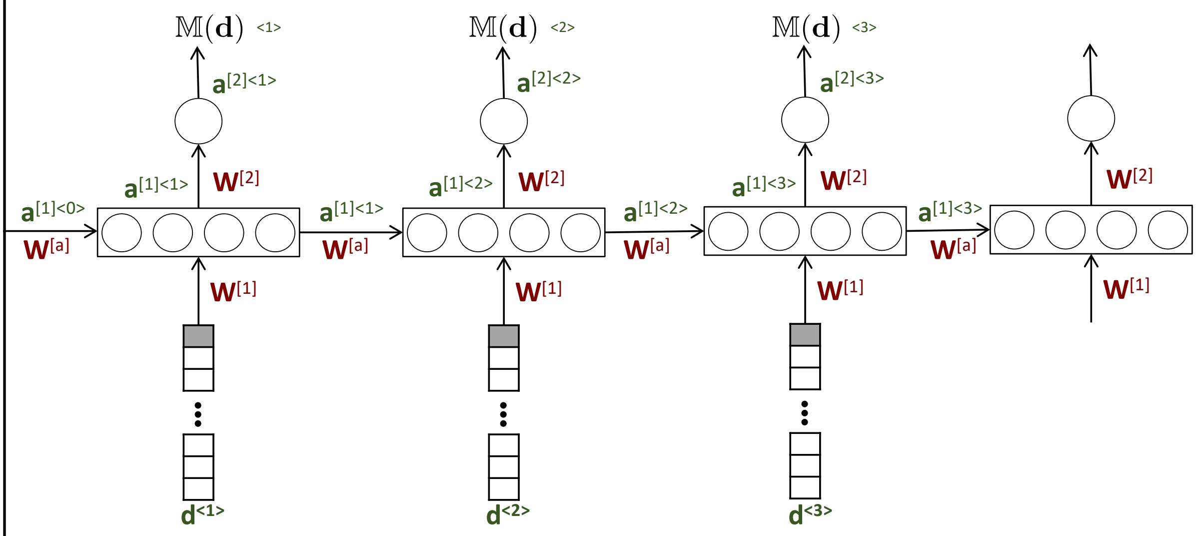
Unrolling Recurrent Neural Networks



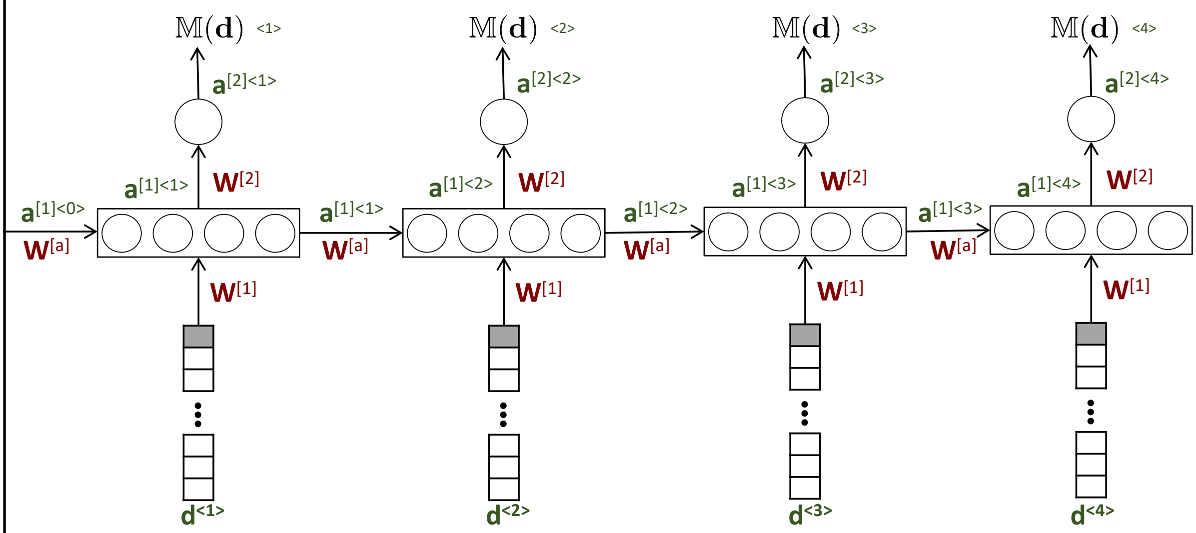
Unrolling Recurrent Neural Networks



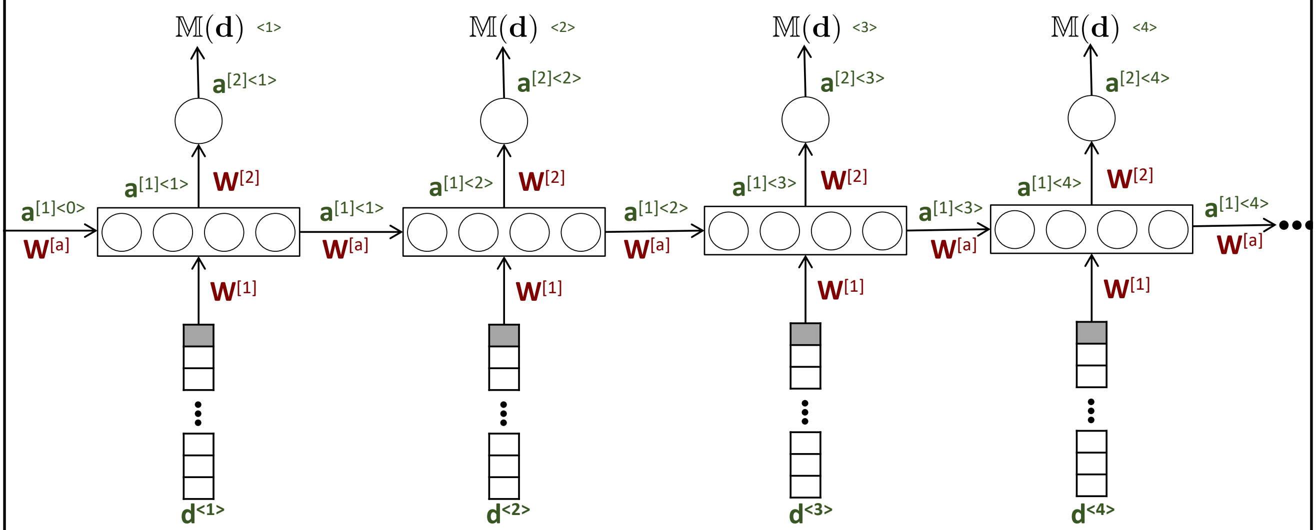
Unrolling Recurrent Neural Networks



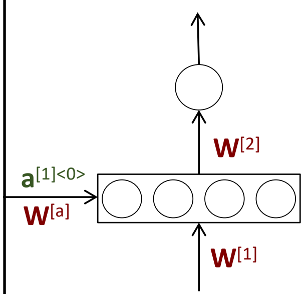
Unrolling Recurrent Neural Networks



Unrolling Recurrent Neural Networks

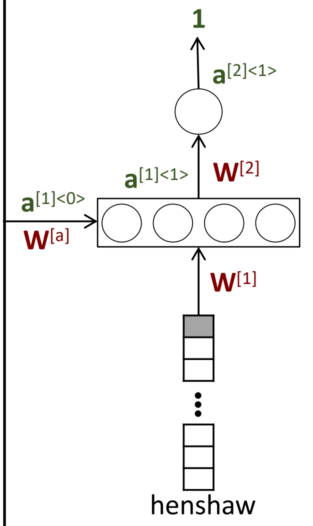


Unrolling Recurrent Neural Networks



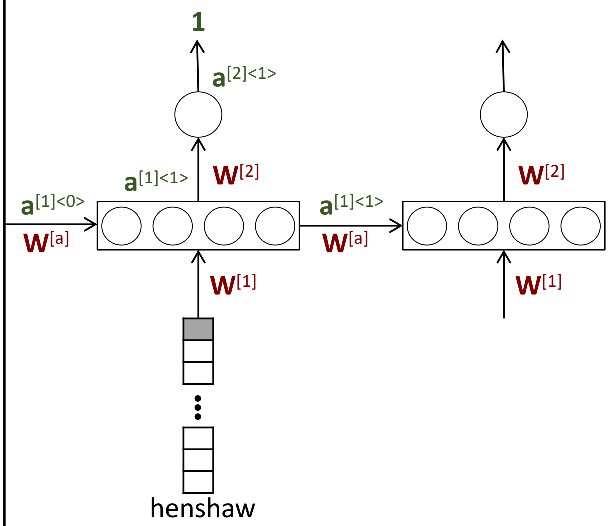
henshaw and marsh offered a defensive masterclass

Unrolling Recurrent Neural Networks



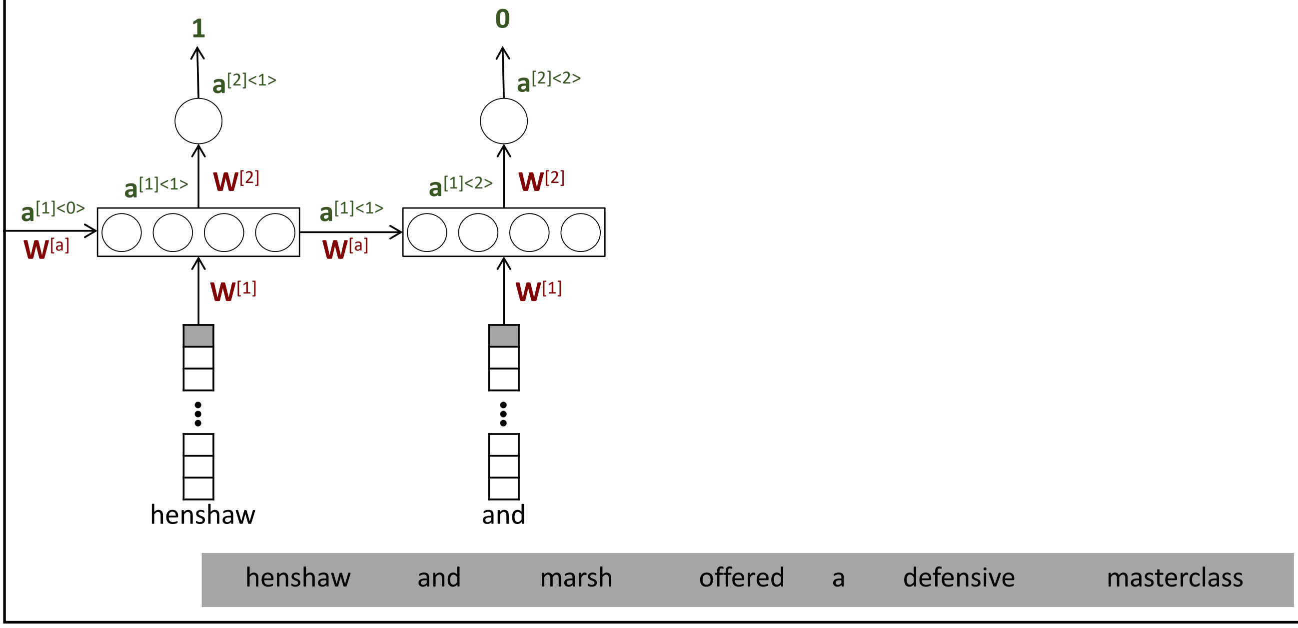
henshaw and marsh offered a defensive masterclass

Unrolling Recurrent Neural Networks

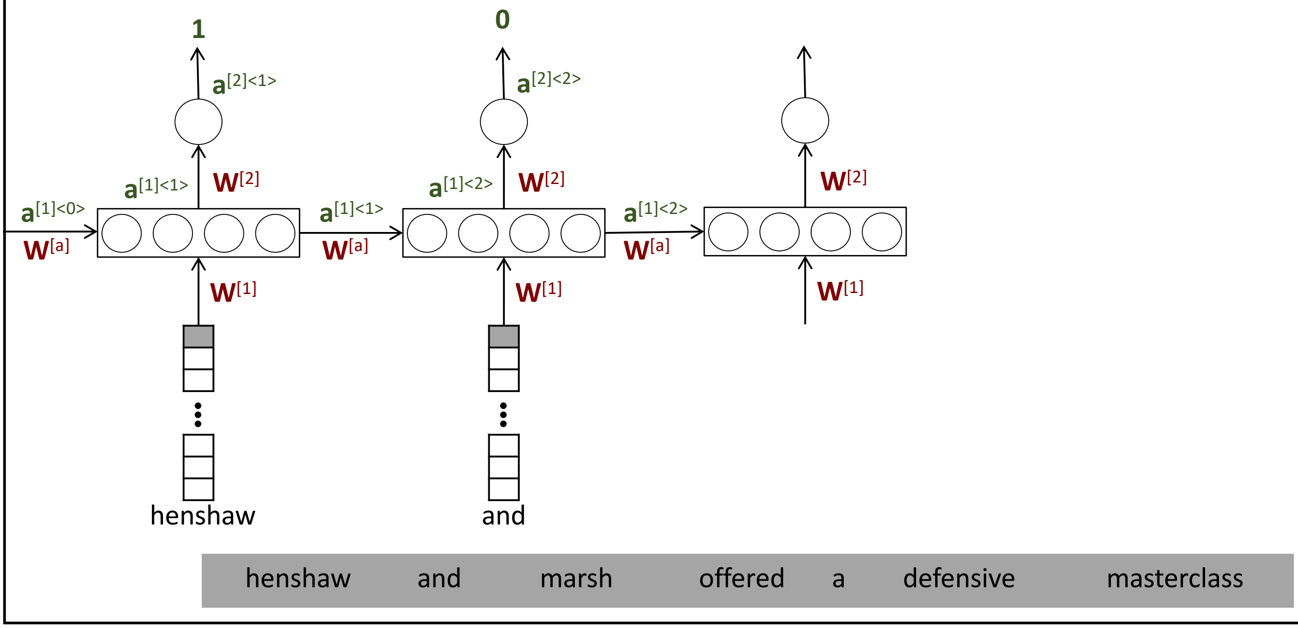


henshaw and marsh offered a defensive masterclass

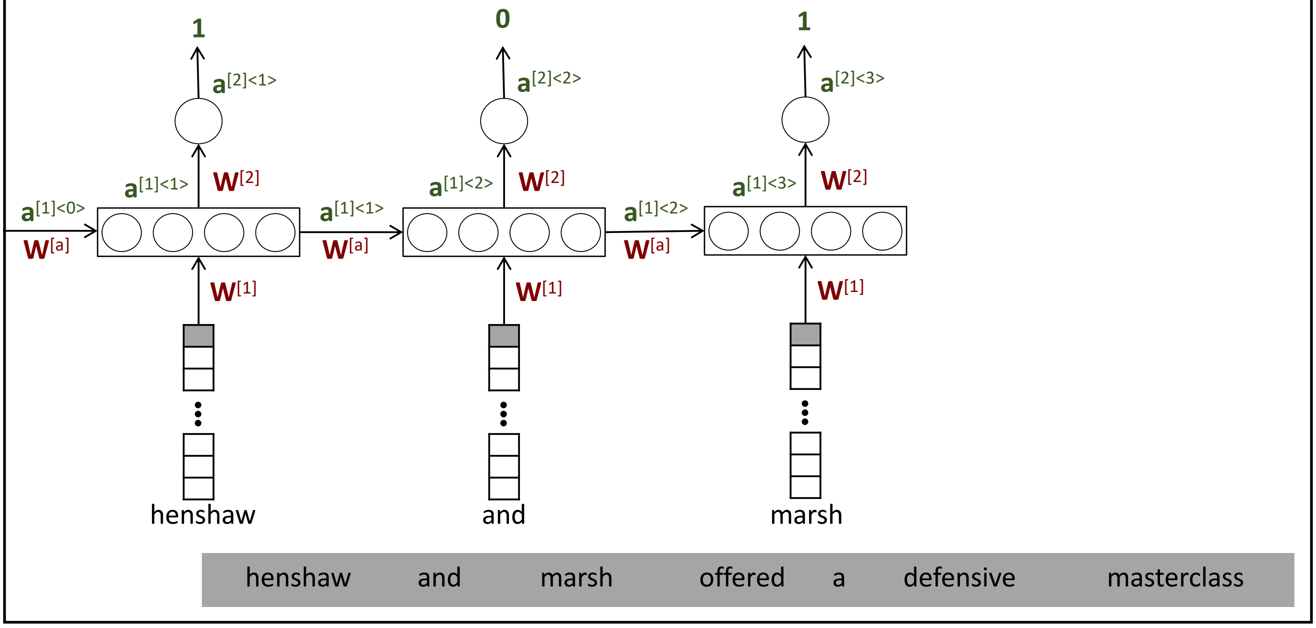
Unrolling Recurrent Neural Networks



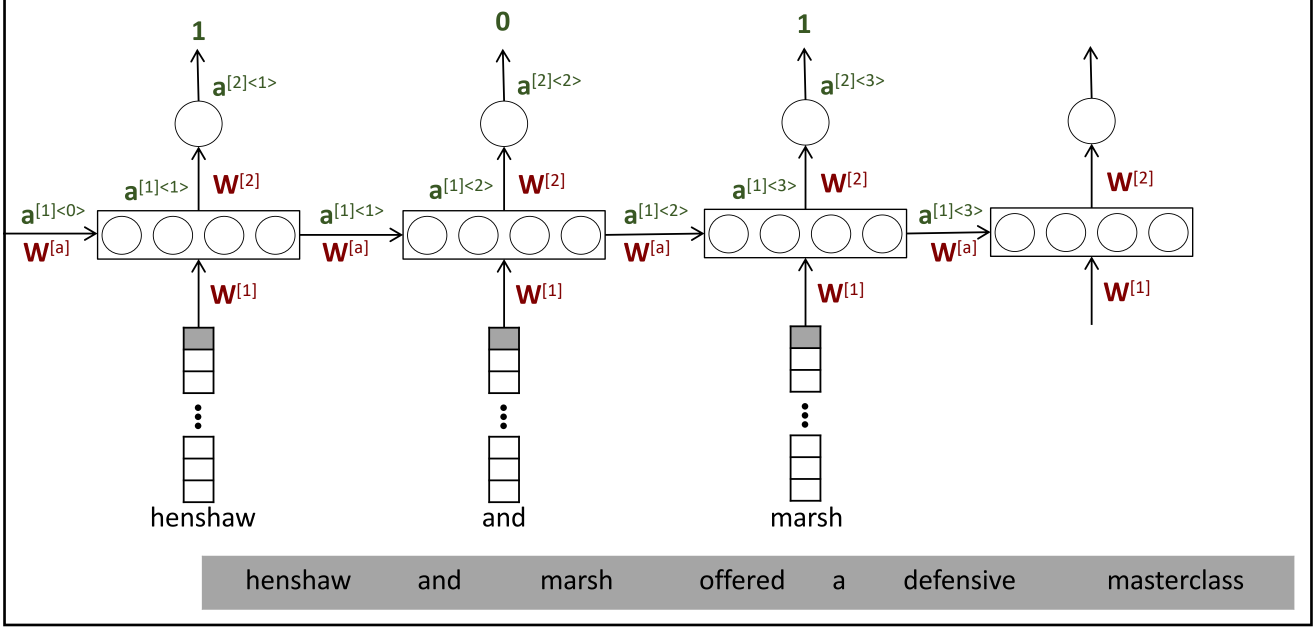
Unrolling Recurrent Neural Networks



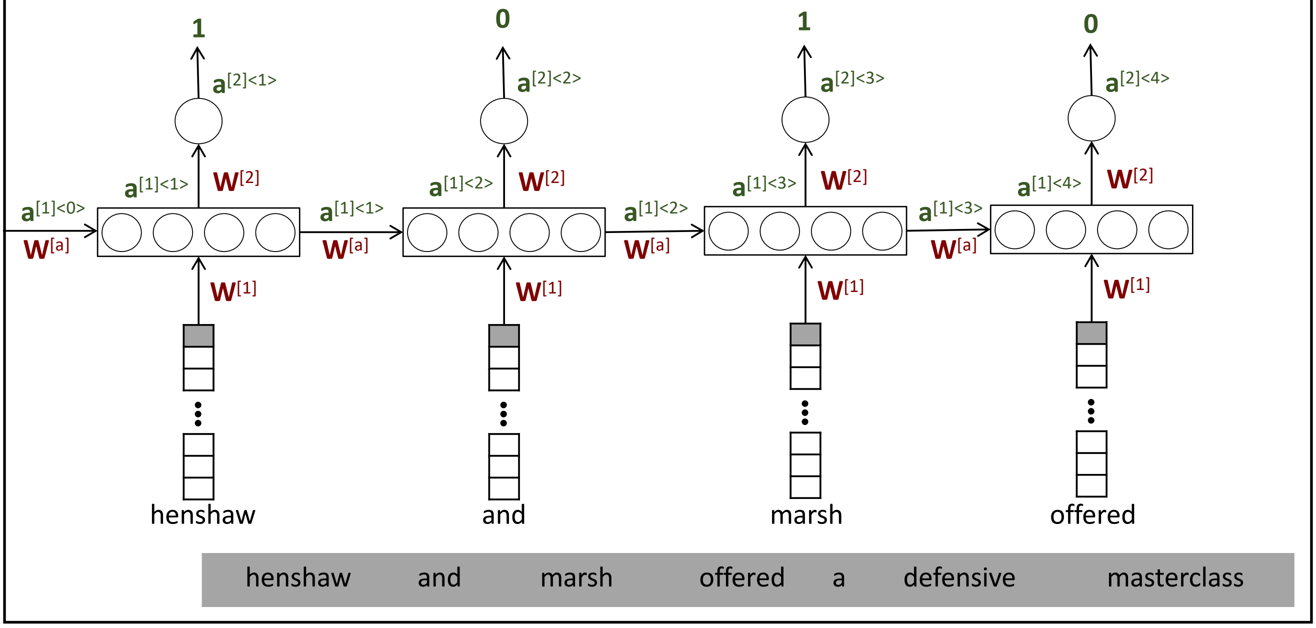
Unrolling Recurrent Neural Networks



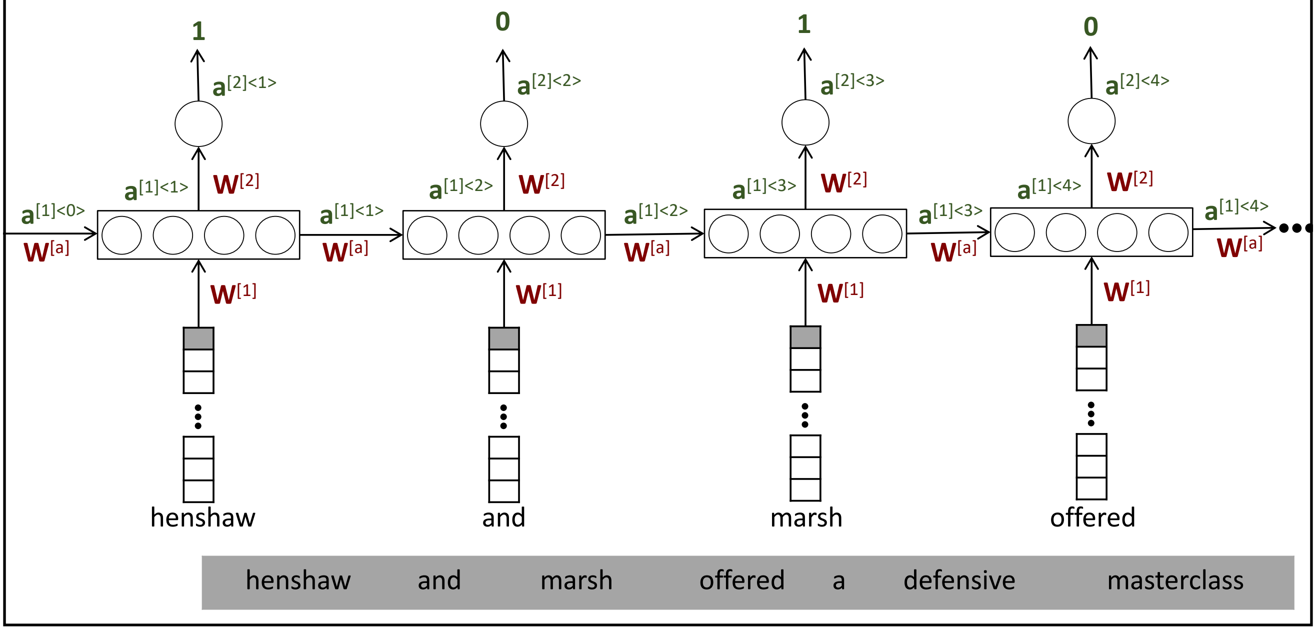
Unrolling Recurrent Neural Networks



Unrolling Recurrent Neural Networks



Unrolling Recurrent Neural Networks

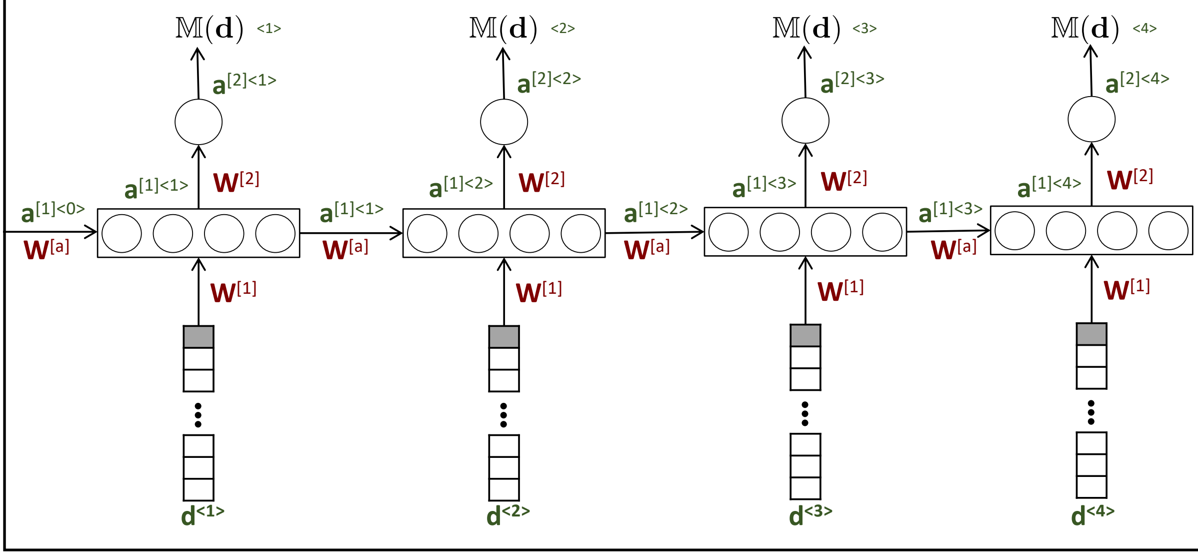


Training RNNs

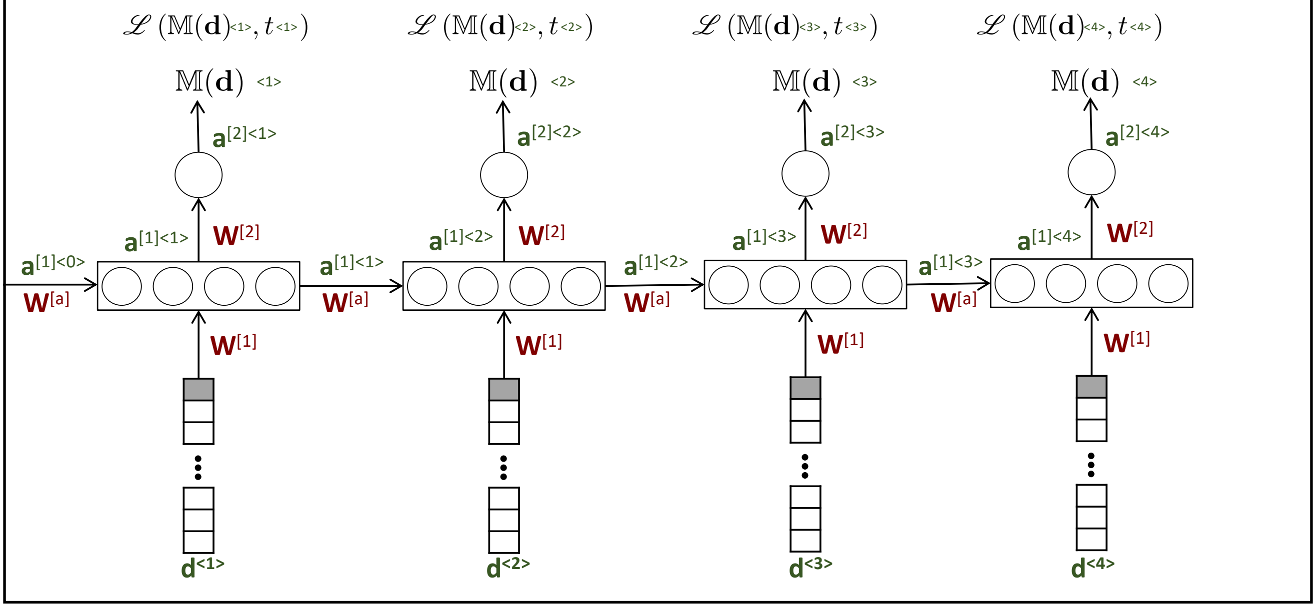
Training RNNs requires very little modifiction to our basic backpropagation of error algorithm

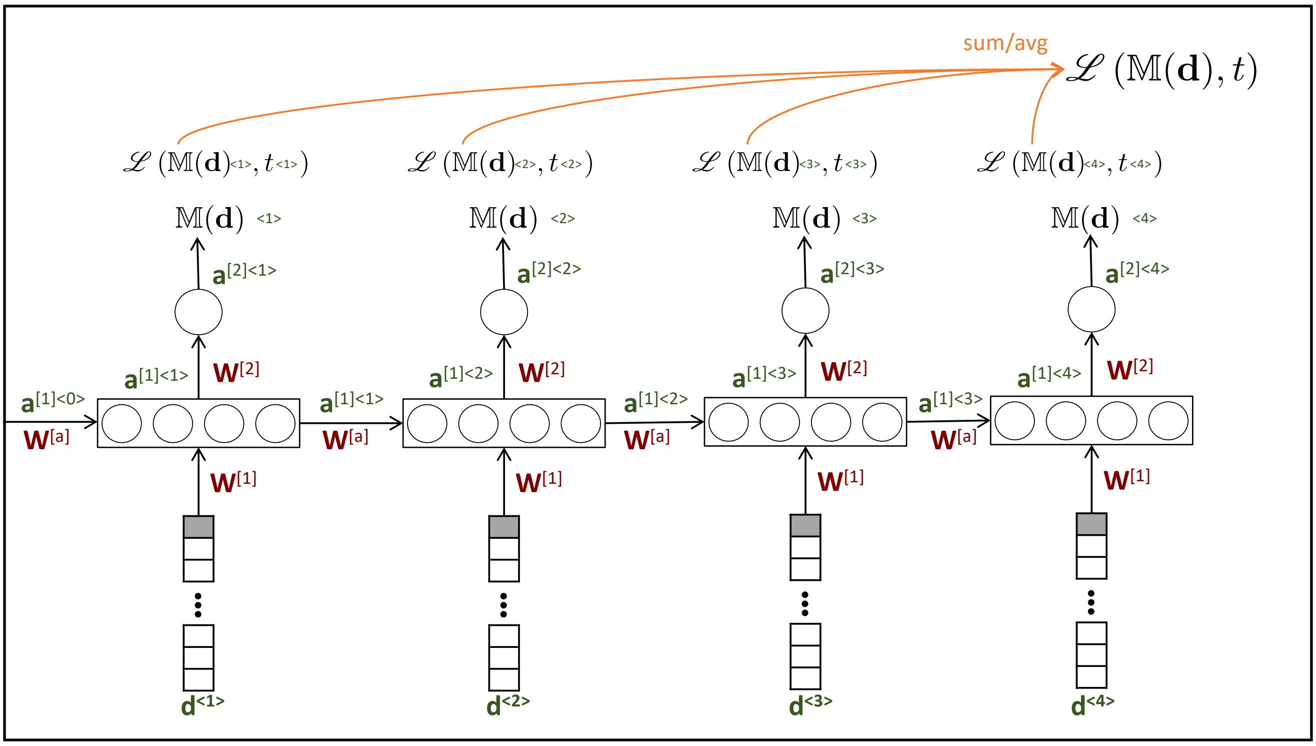
The only difference is that the backward pass goes through different iterations of the forward passing network - **backpropagation of errors through time**

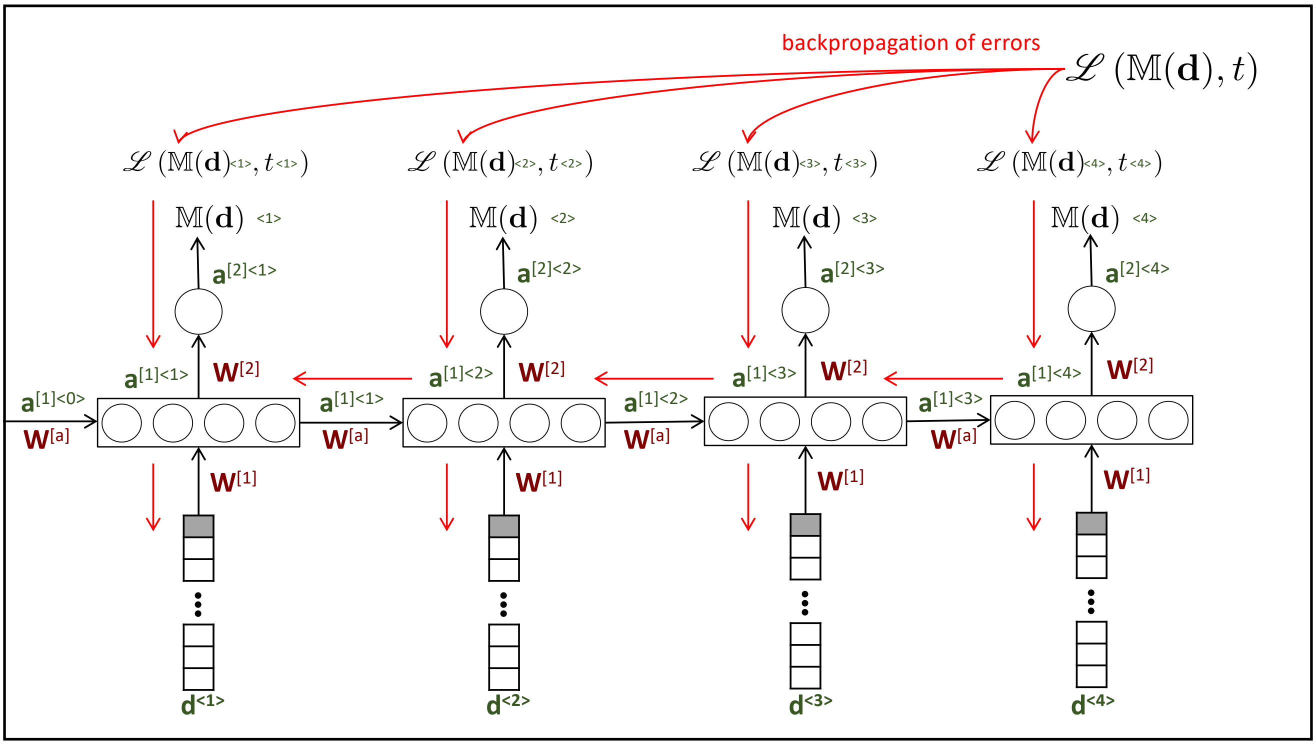
Backpropagation of Error Through Time



Backpropagation of Error Through Time

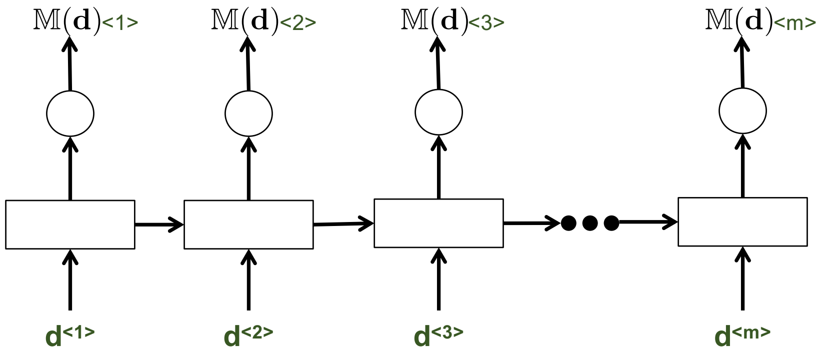




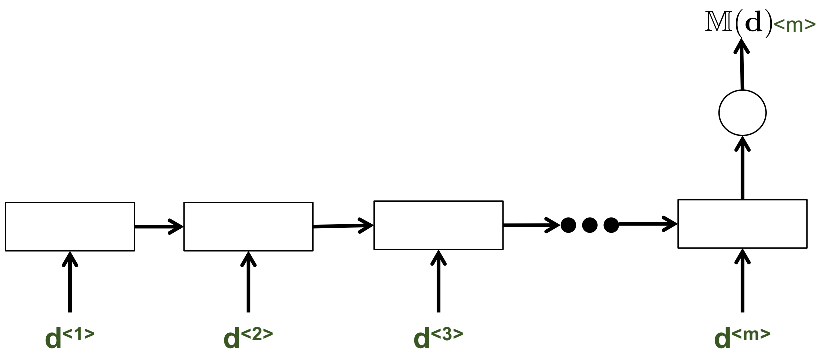


Different Types of RNN

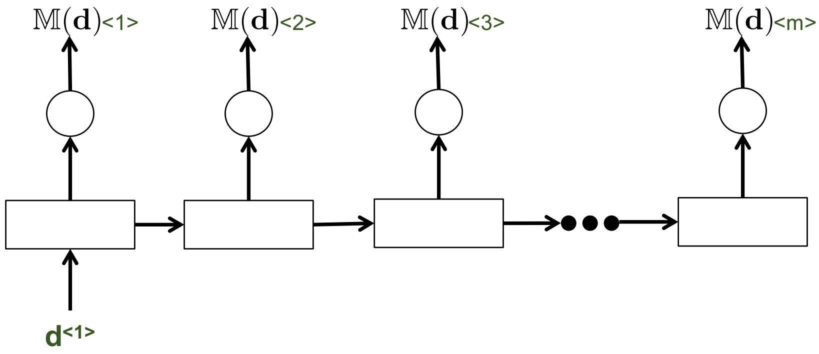
Types Of RNN



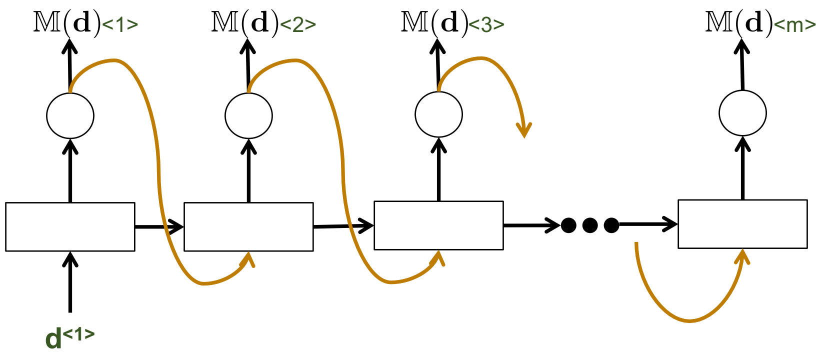
Types Of RNN



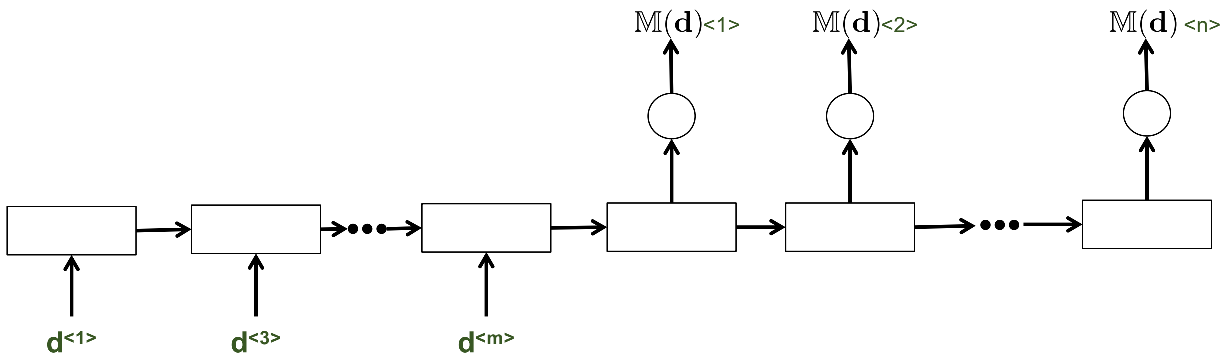
Types Of RNN



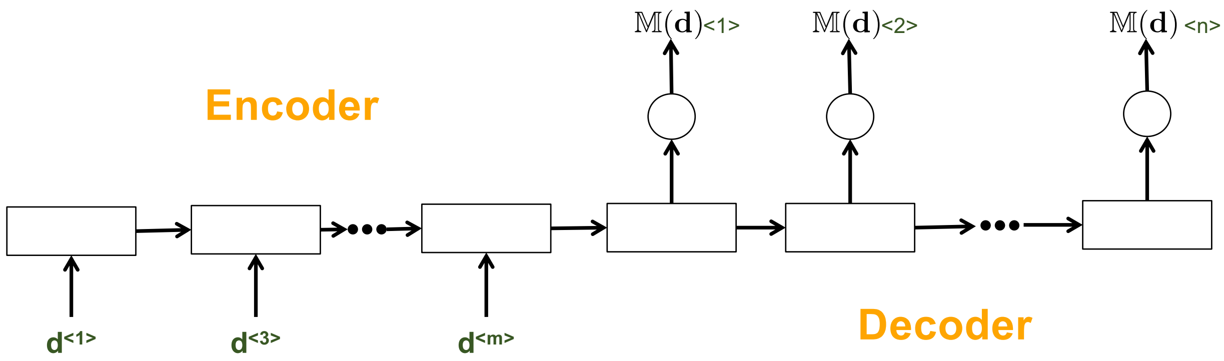
Types Of RNN



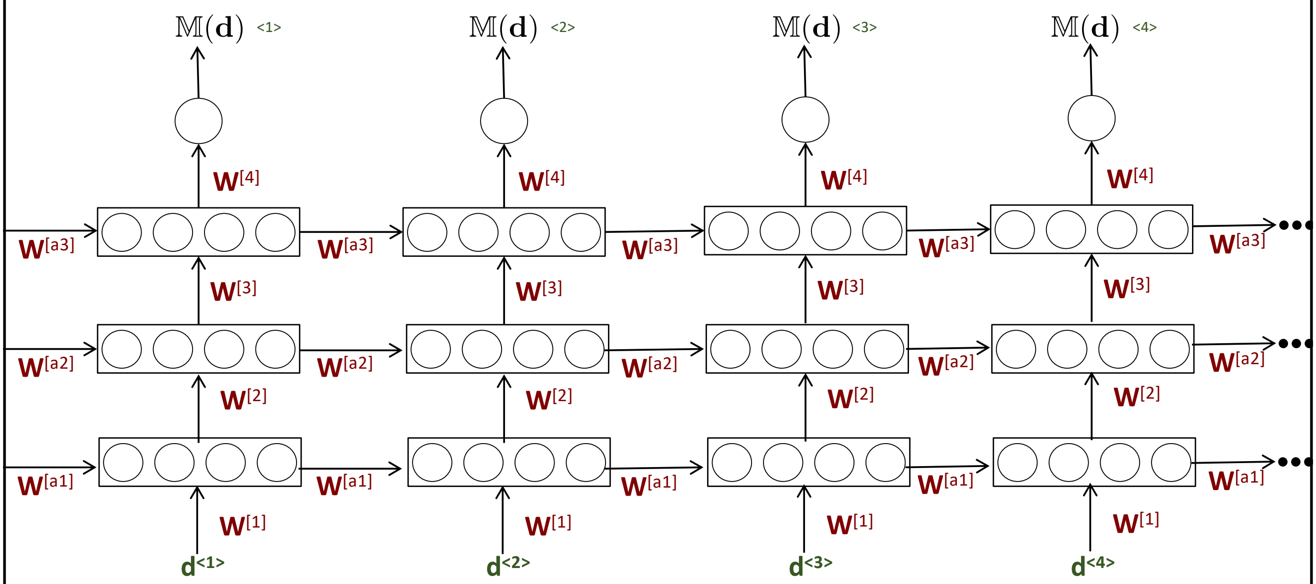
Types Of RNN



Types Of RNN

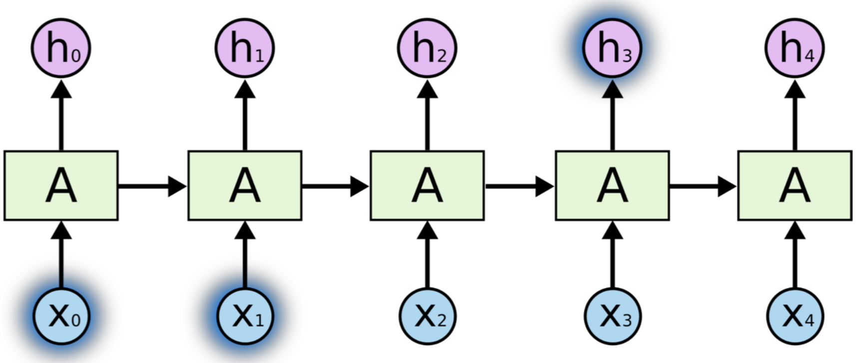


Deep Recurrent Neural Networks



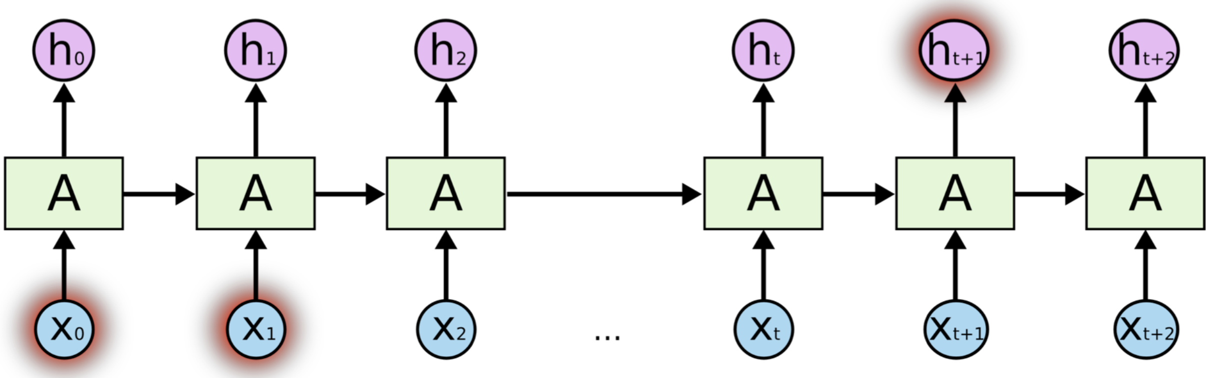
LSTMS

Distant Patterns



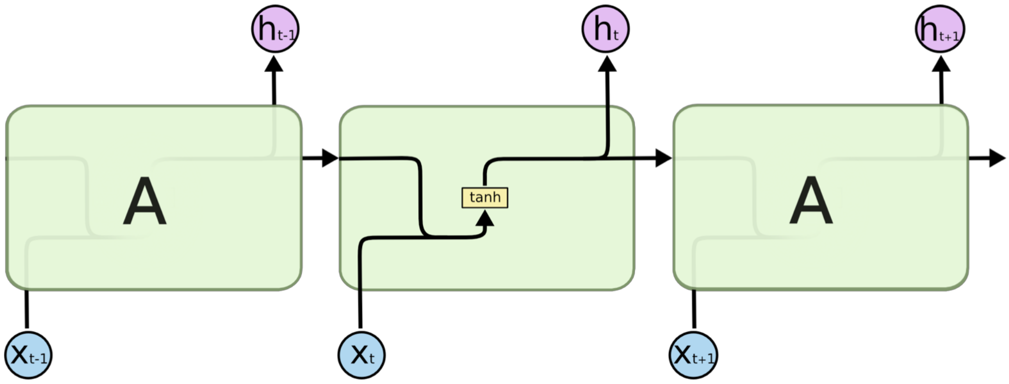
Understanding LSTM Networks
http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Distant Patterns



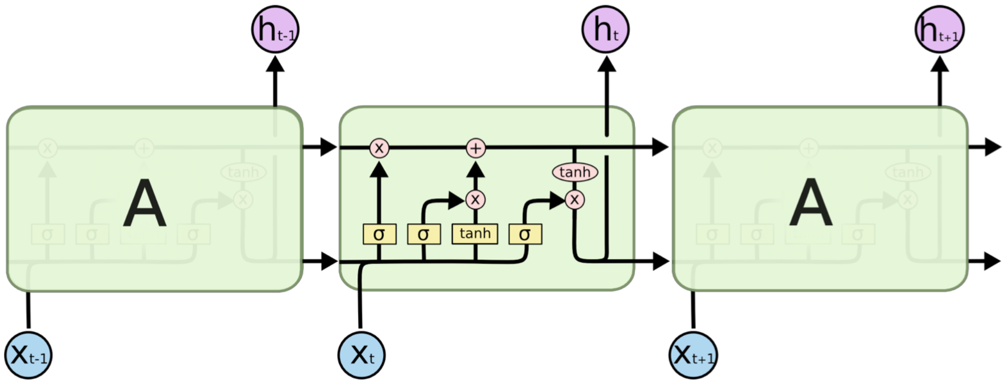
Understanding LSTM Networks
http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Distant Patterns



Understanding LSTM Networks
http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Distant Patterns



Understanding LSTM Networks
http://colah.github.io/posts/2015-08-Understanding-LSTMs/

LSTMs

Long short term memory (LSTM) networks are an RNN variant designed to learn longer term patterns in sequential data

Have become extremely popular and demosntrated to be very effective in sequential learning problems

Summary

Section Takeaways

The main takeaway messages from this section are:

- Recurrent neural networks are neural networks where the activation of units is circulated as an input to the network
- This allows us capture sequential patterns in data
- We often view a recurrent network as an unrolling
- There are different types of RNNs but all can be trained with the backpropogation of error through time algorithm

Questions

?