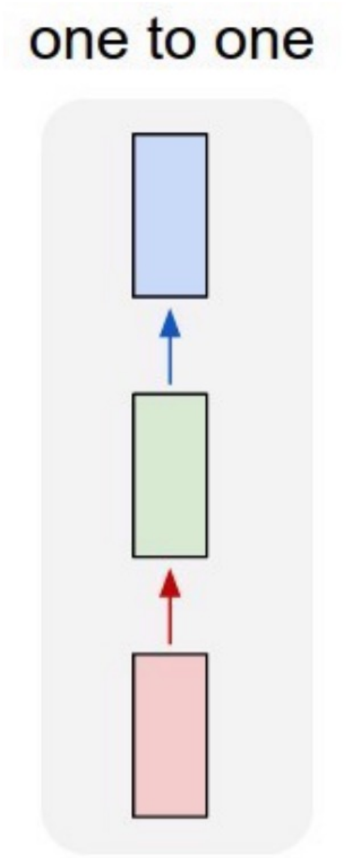
Recurrent Neural Networks

Neural Networks Design And Application

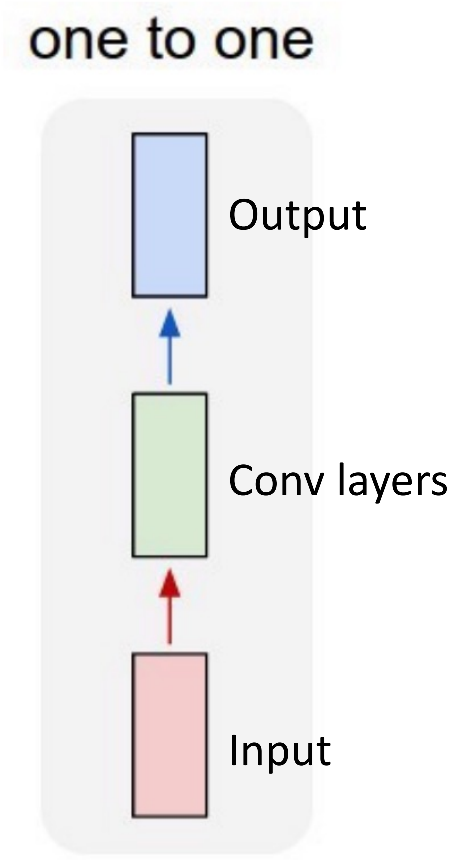
Recurrent neural networks



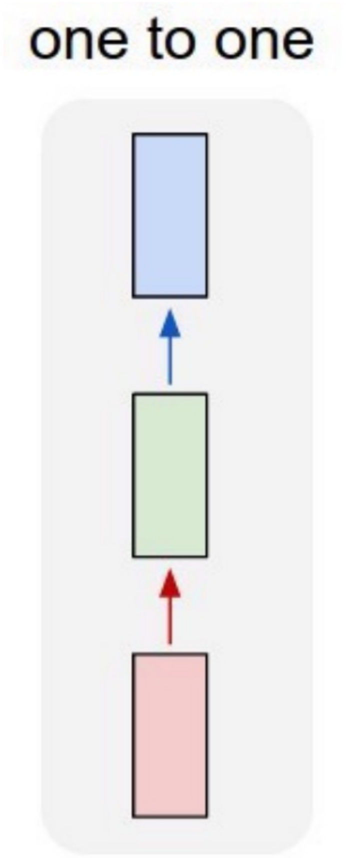
Vanilla
CNNs

Recurrent neural networks

Vanilla
CNNs



Recurrent neural networks

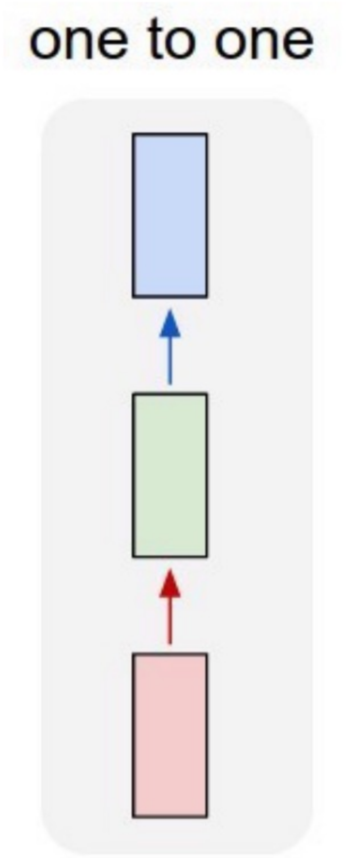


Vanilla
CNNs

Video data: multiple frames per second

Recurrent neural networks

Vanilla
CNNs



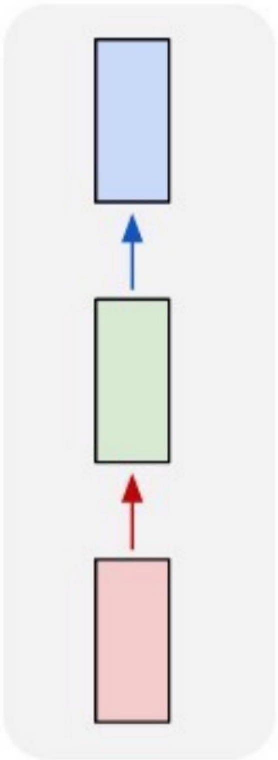
Video data: multiple frames per second

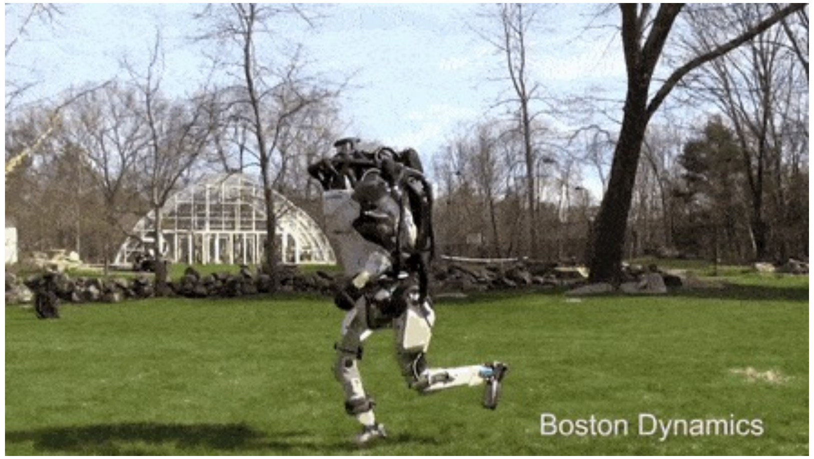
Action recognition

Recurrent neural networks

one to one

Vanilla
CNNs





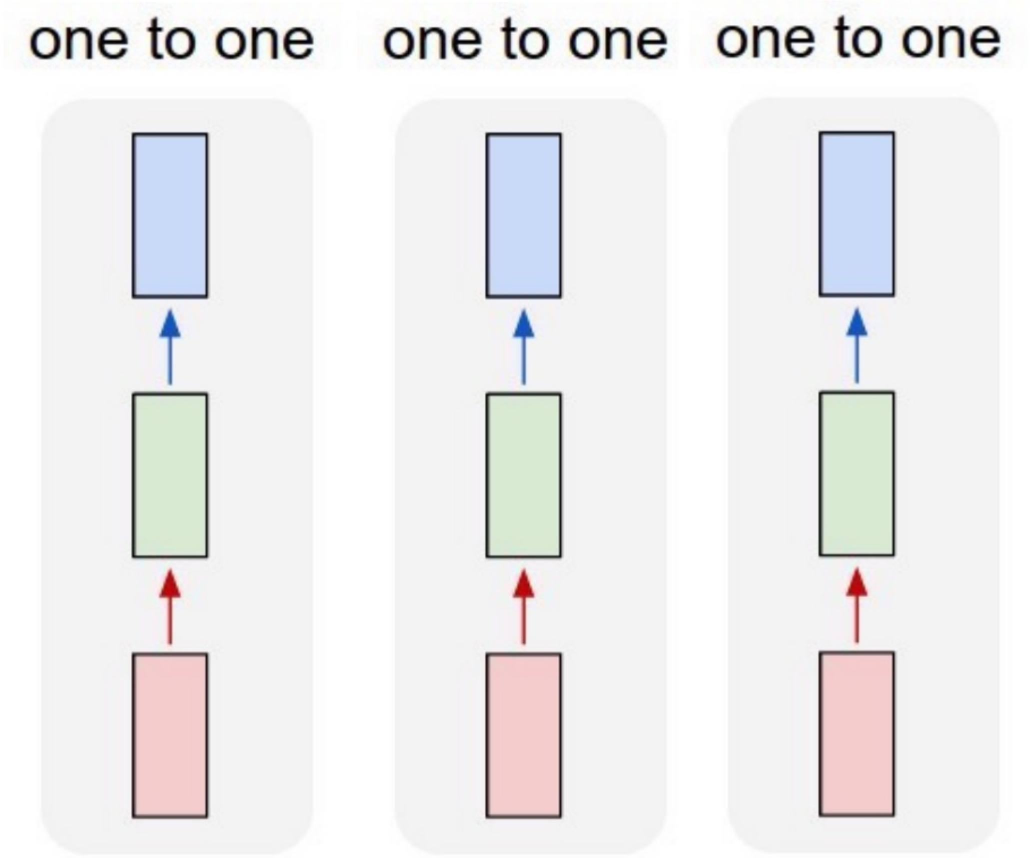
Video data: multiple frames per second

Action recognition

Image credit: Boston dynamics

Recurrent neural networks

Vanilla CNNs

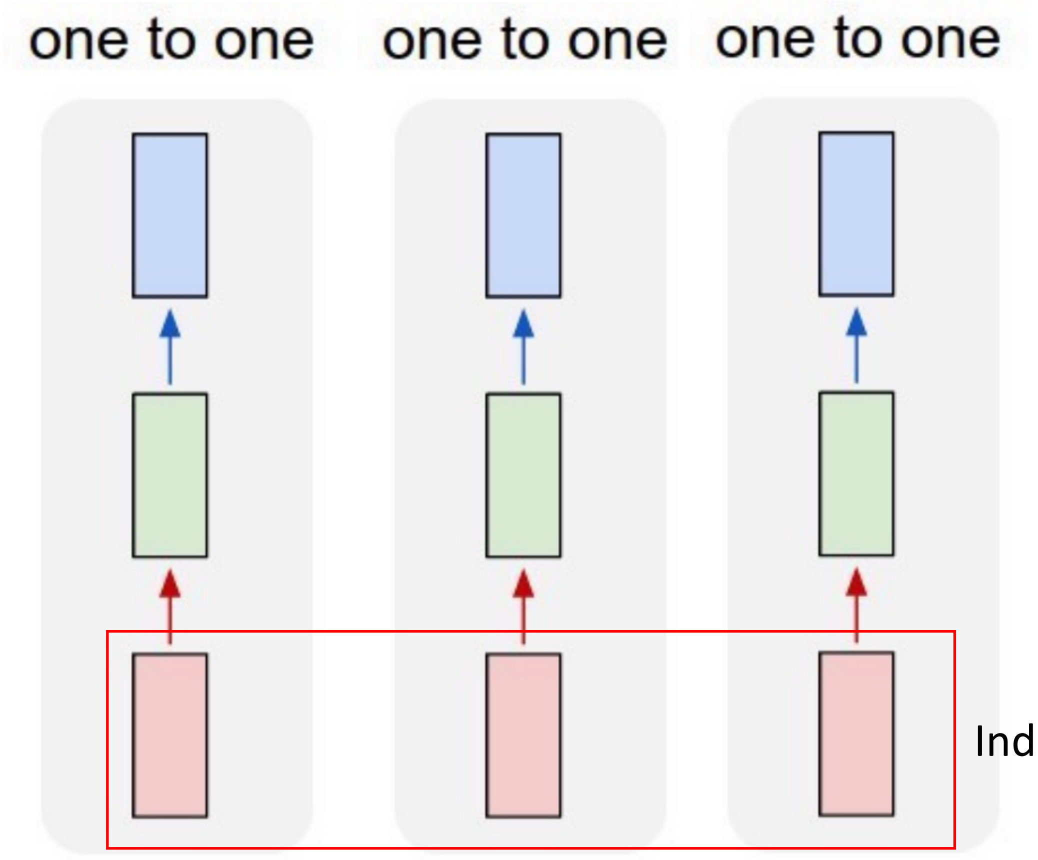


Video data: multiple frames per second

Action recognition

Recurrent neural networks

Vanilla
CNNs



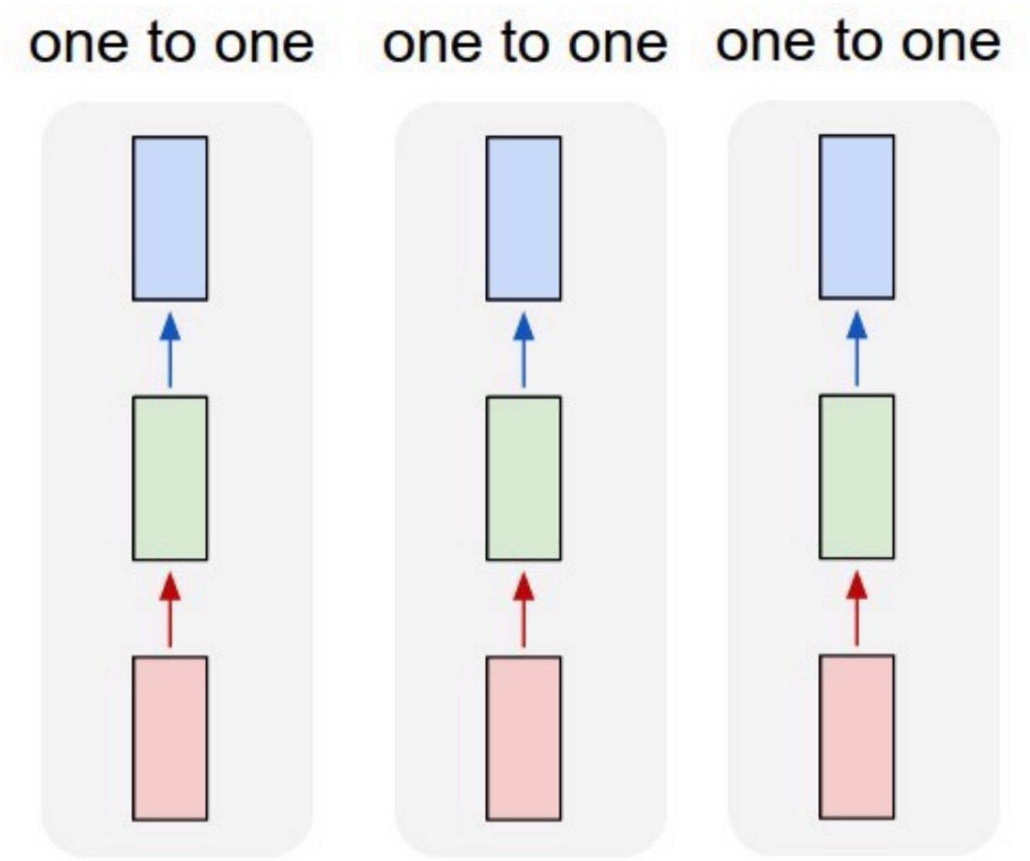
Independently

Video data: multiple frames per second

Action recognition

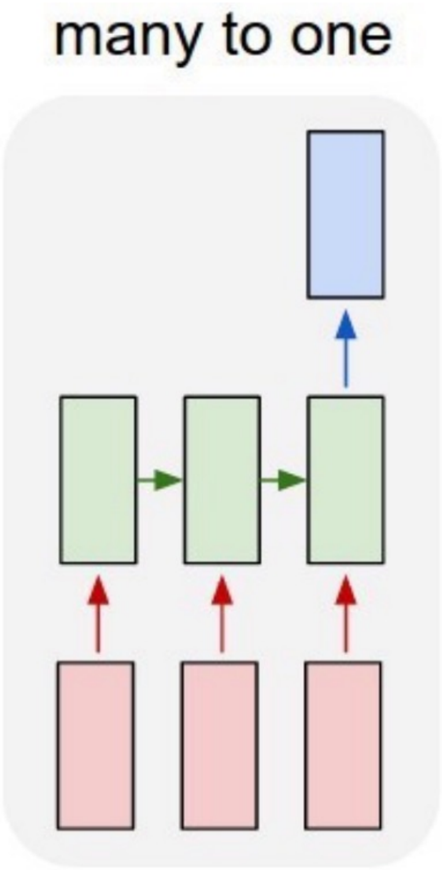
Recurrent neural networks

Vanilla CNNs



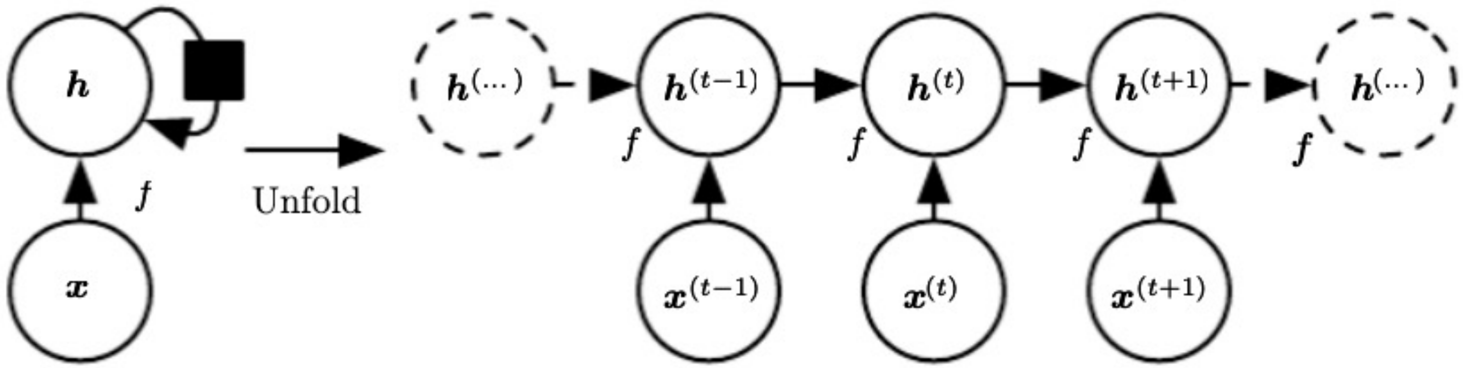
Video data: multiple frames per second

Action recognition

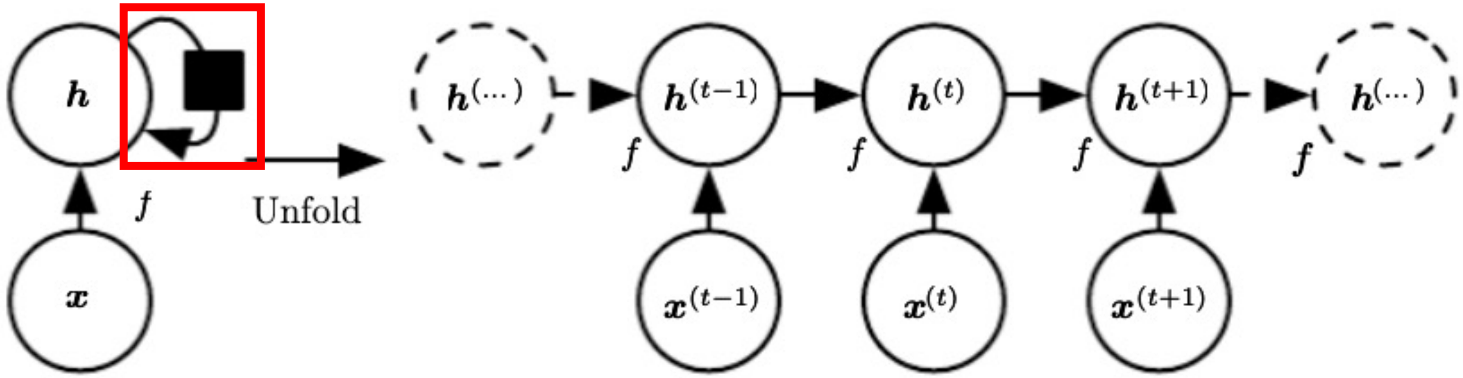


Video data: multiple frames per second

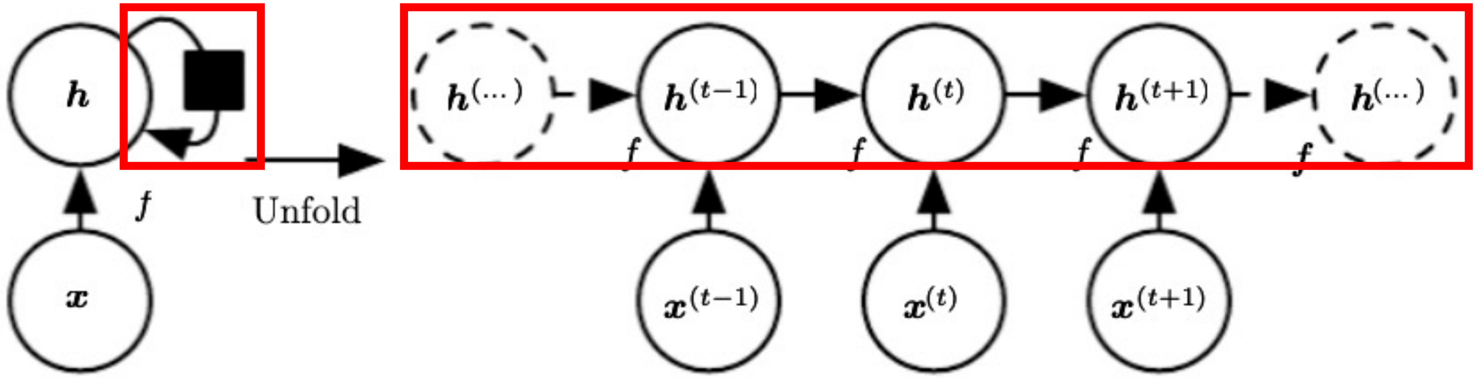
Recurrent neural networks



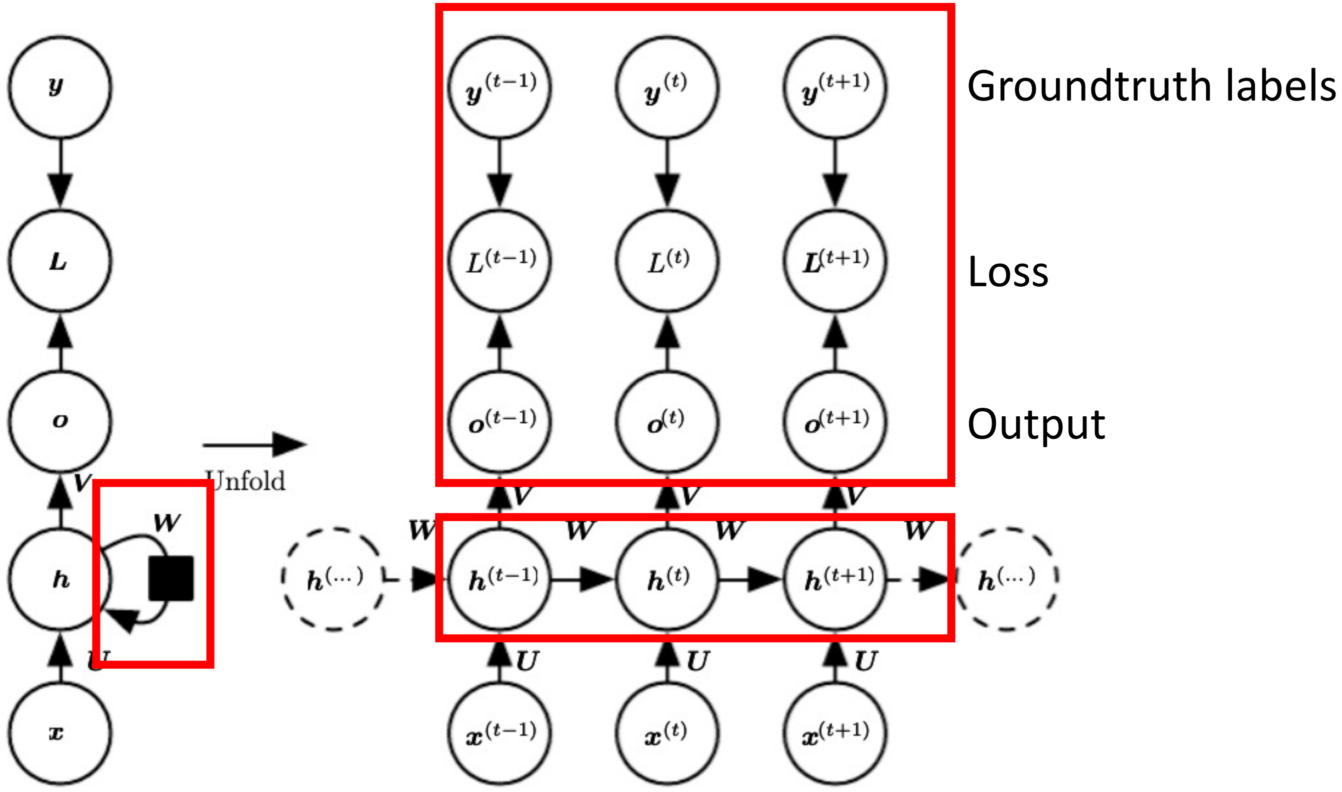
Recurrent neural networks



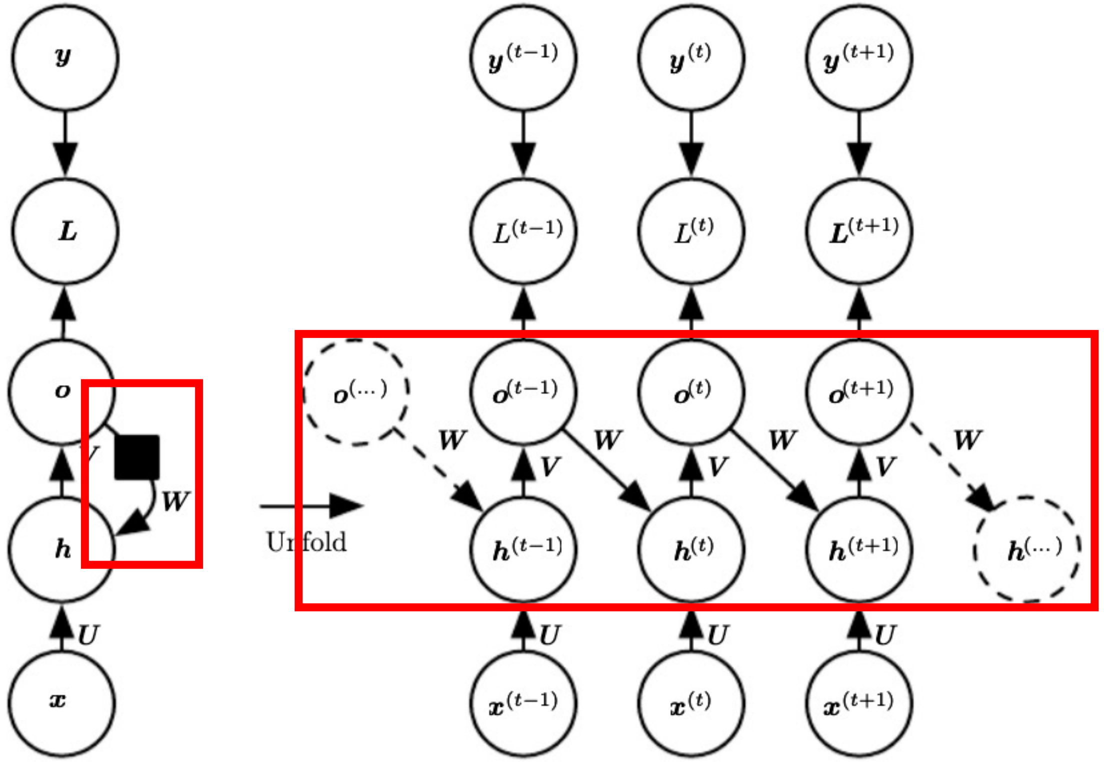
Recurrent neural networks



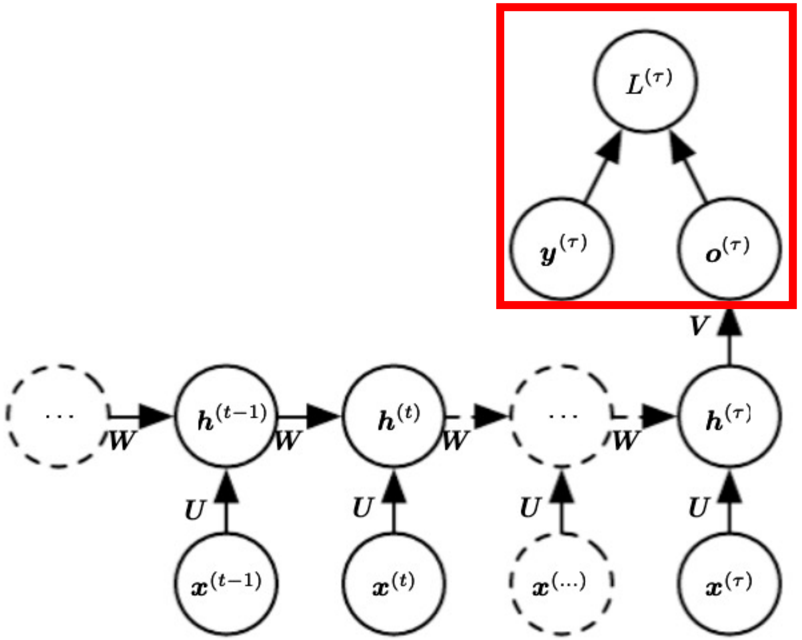
Recurrent networks



Recurrent networks



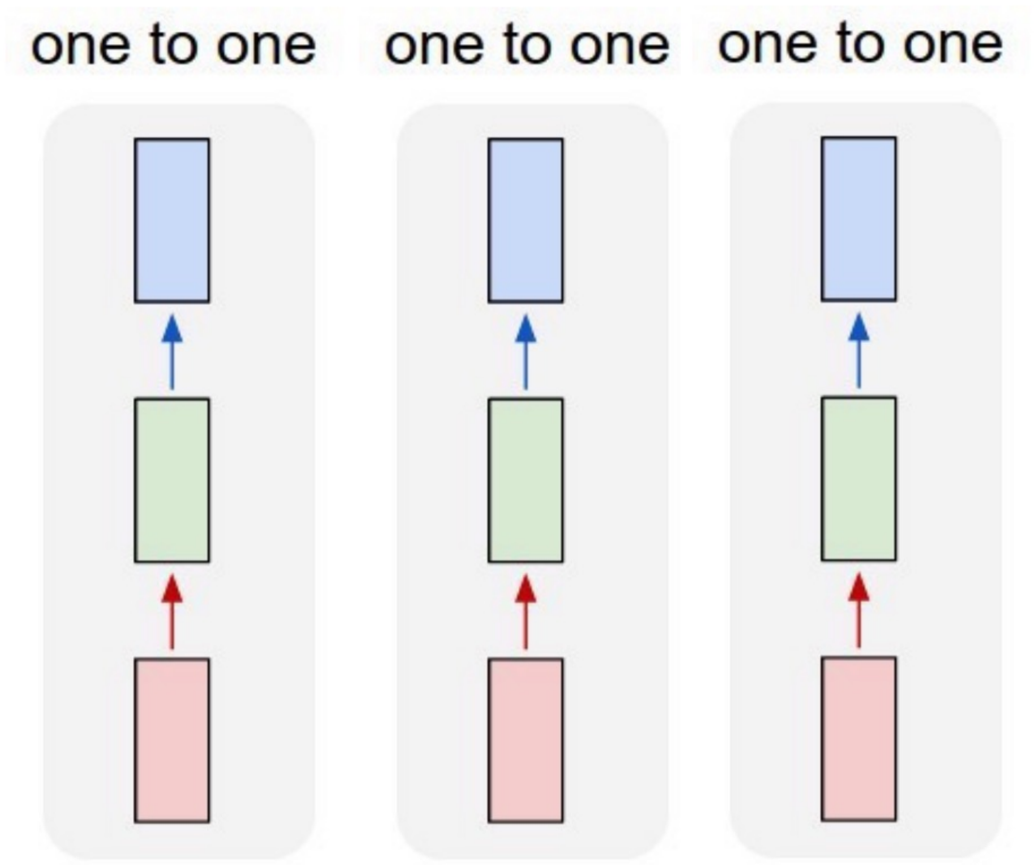
Recurrent networks



Only one output

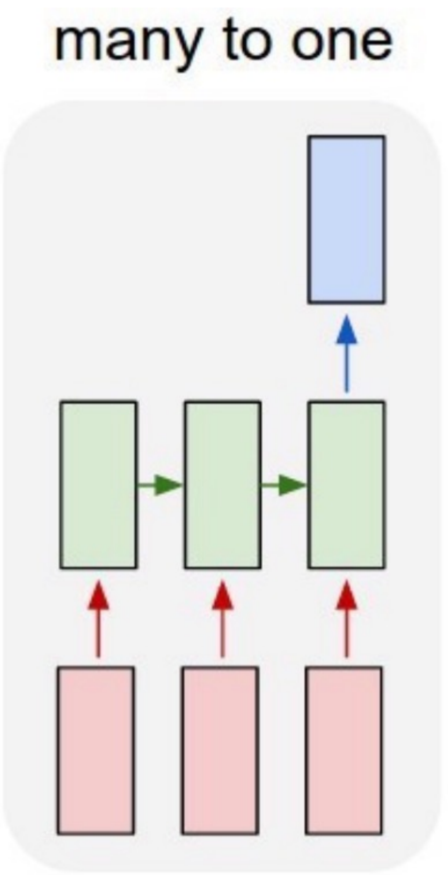
Recurrent neural networks

Vanilla CNNs



Video data: multiple frames per second

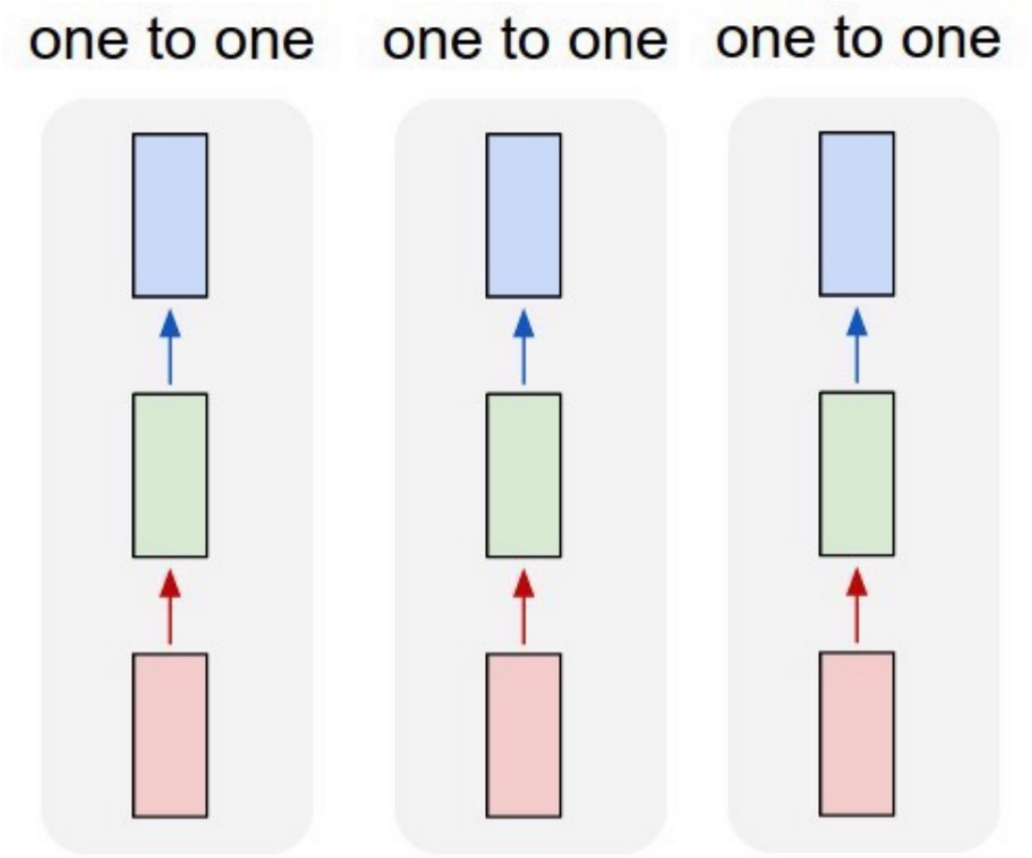
Action recognition



Video data: multiple frames per second

Recurrent neural networks

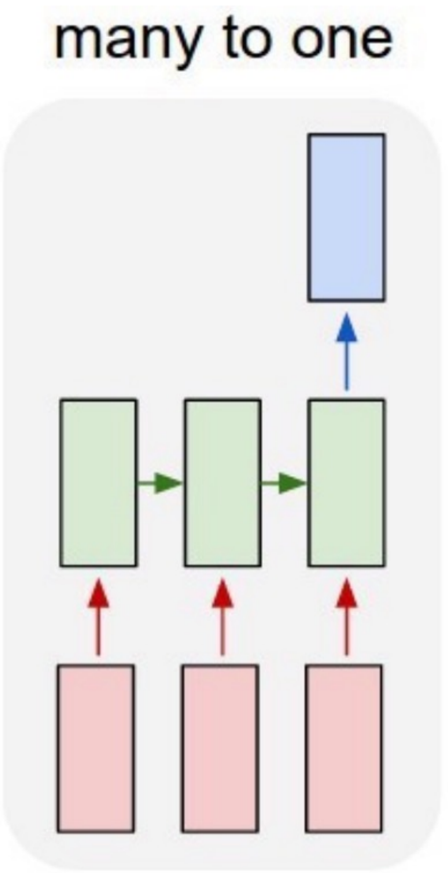
Vanilla CNNs



Video data: multiple frames per second

Action recognition

Action recognition



Video data: multiple frames per second

Recurrent neural networks in practice

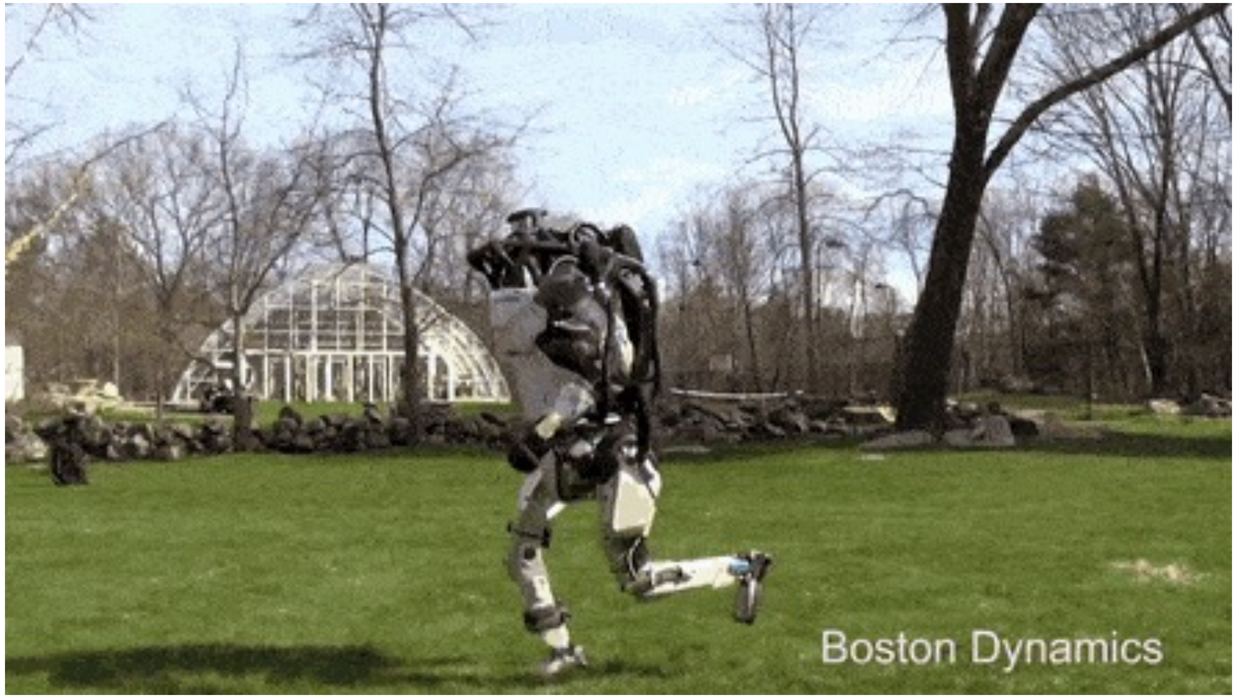
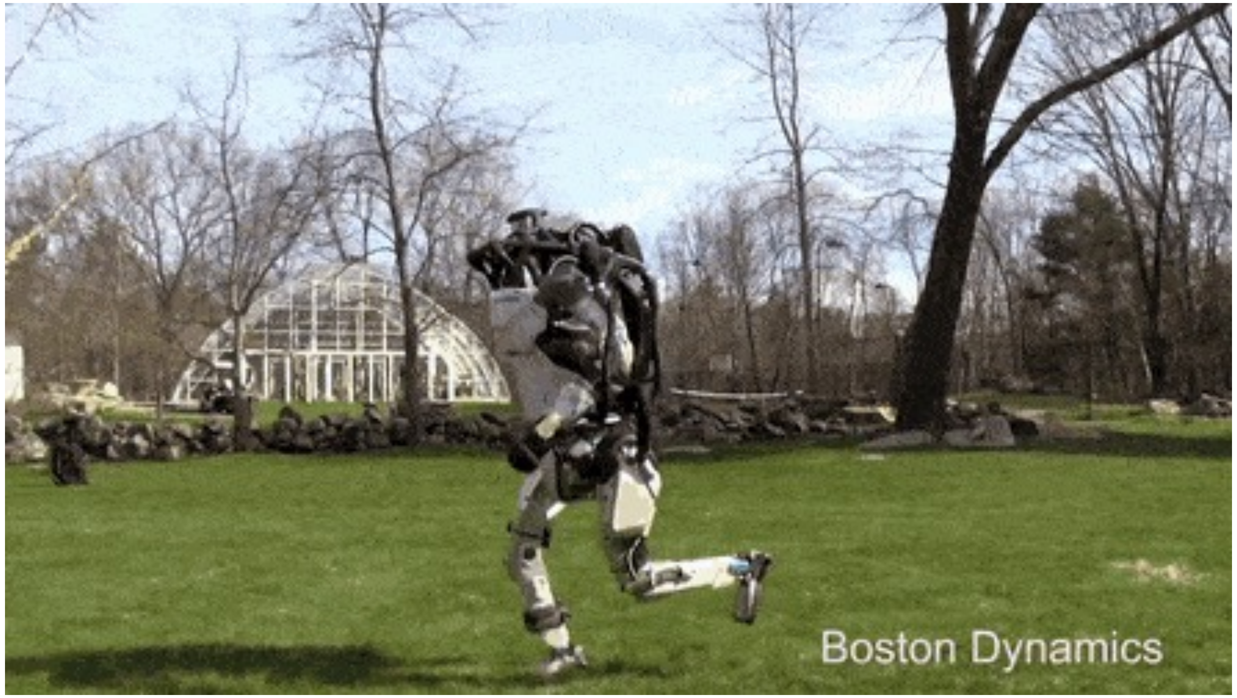


Image credit: Boston dynamics

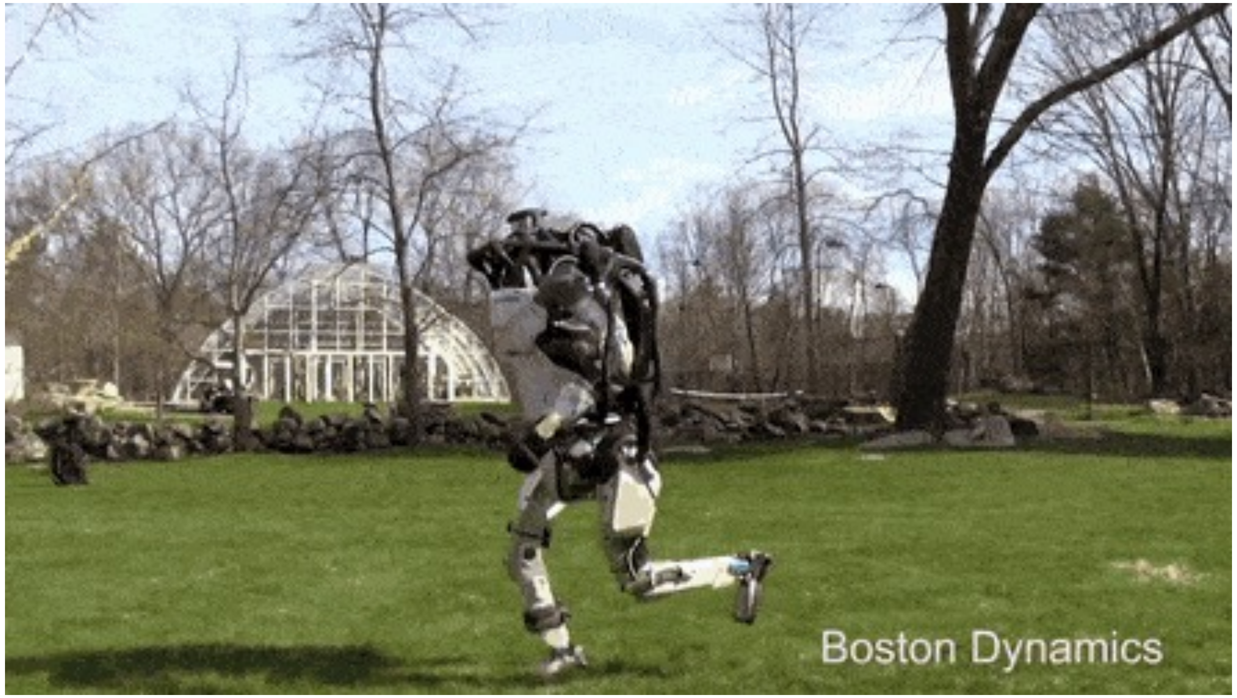
Recurrent neural networks in practice



Q: what is the action?

Image credit: Boston dynamics

Recurrent neural networks in practice



Q: what is the action?

Running or opening a door?

Image credit: Boston dynamics

Recurrent neural networks in practice

Q: what is the action?

Image credit: Boston dynamics

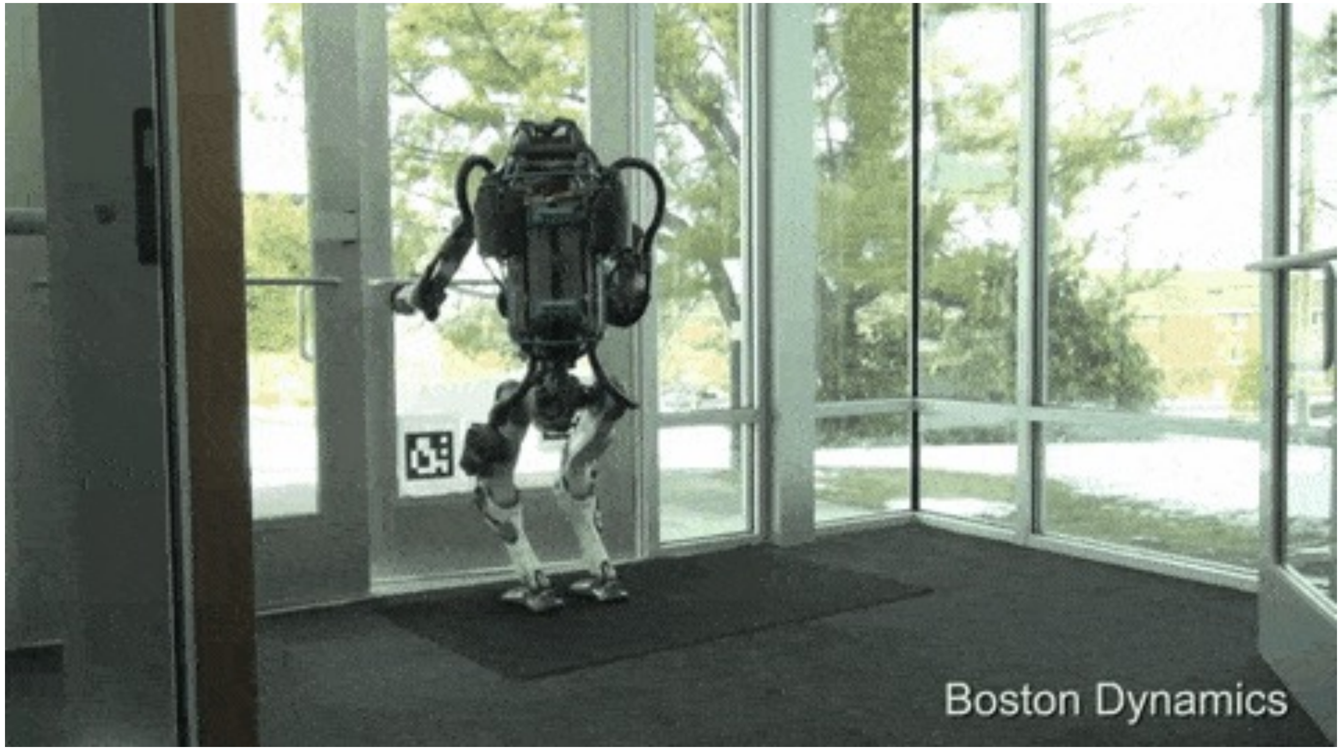
Recurrent neural networks in practice

Q: what is the action?

Running or opening a door?

Image credit: Boston dynamics

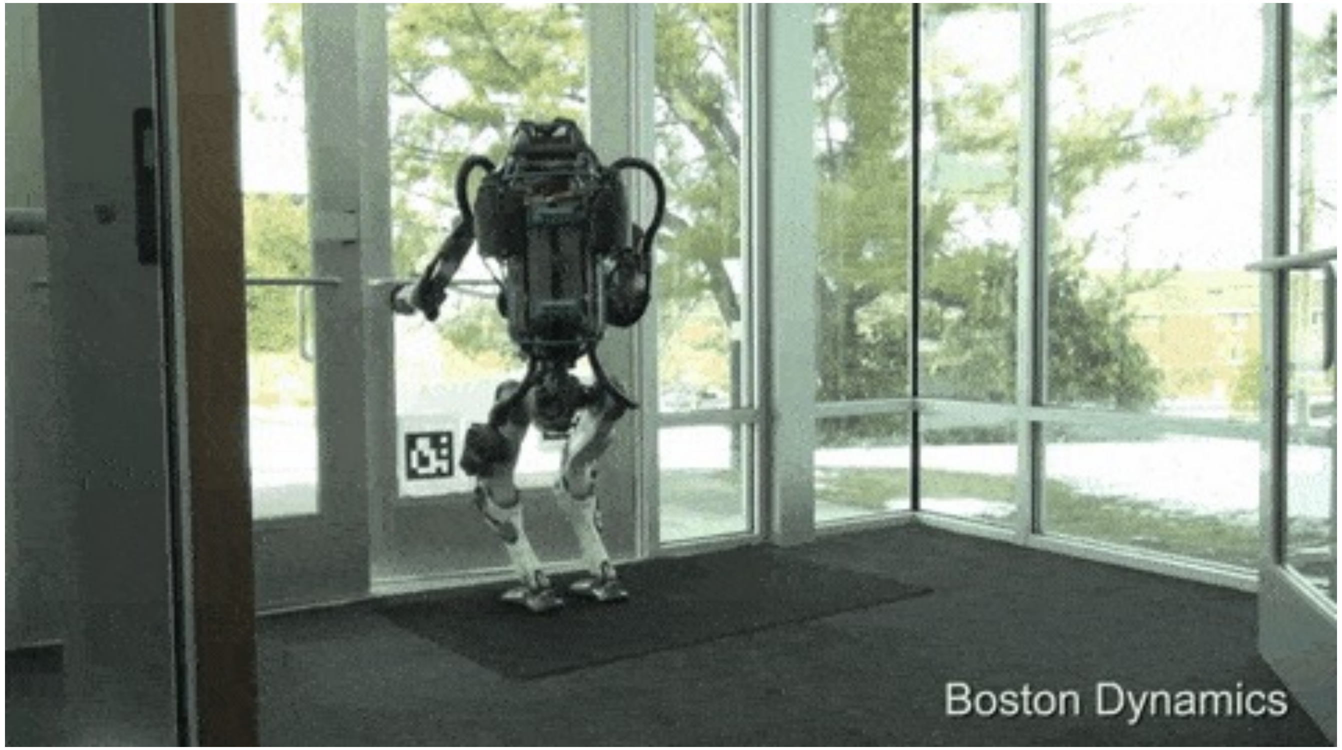
Recurrent neural networks in practice



Q: what is the action?

Image credit: Boston dynamics

Recurrent neural networks in practice

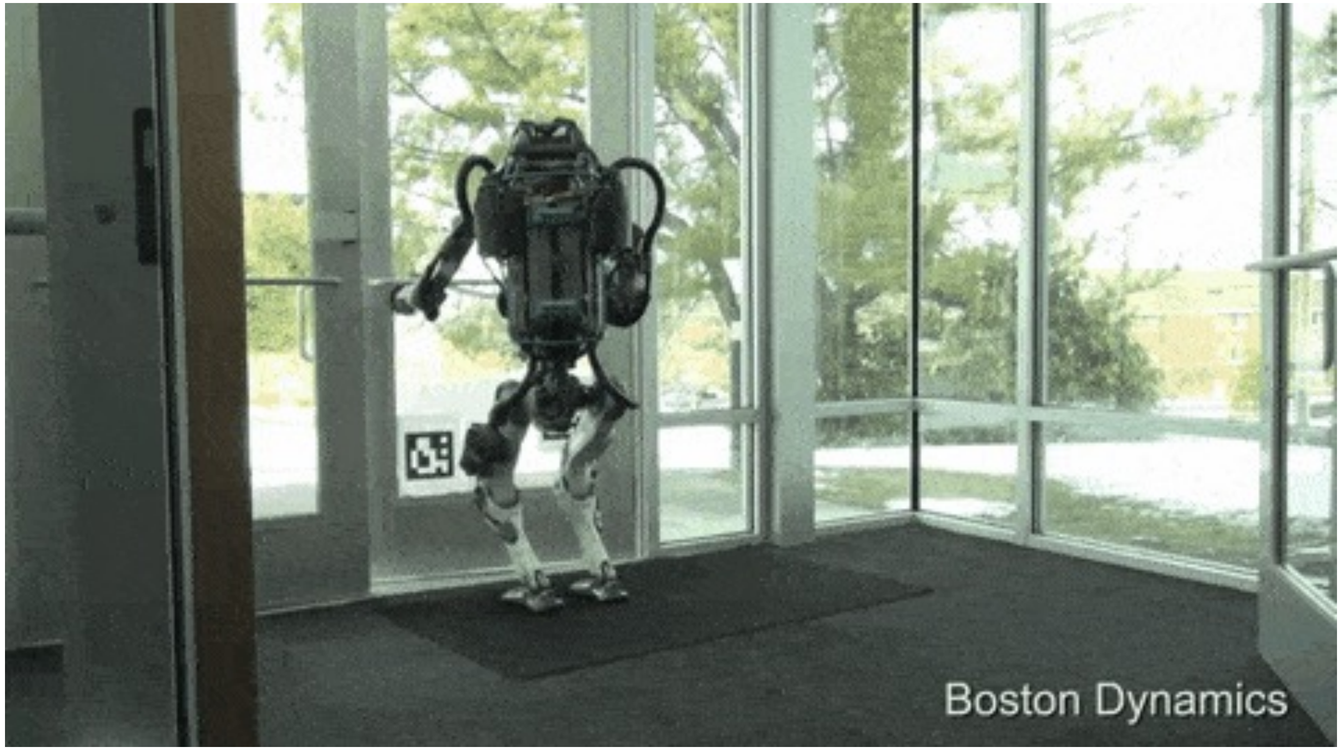


Q: what is the action?

Running or opening a door?

Image credit: Boston dynamics

Recurrent neural networks in practice



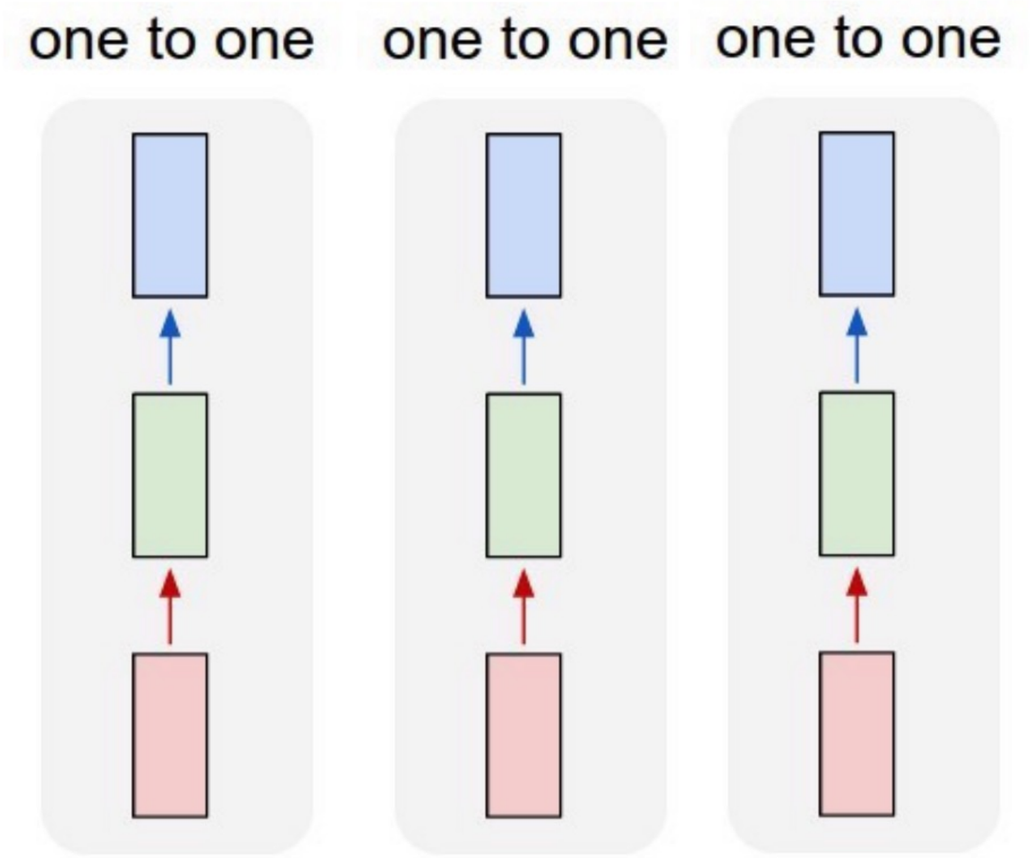
Action recognition:
predict a label from given multiple frames

Q: what is the action?

Running or opening a door?

Image credit: Boston dynamics

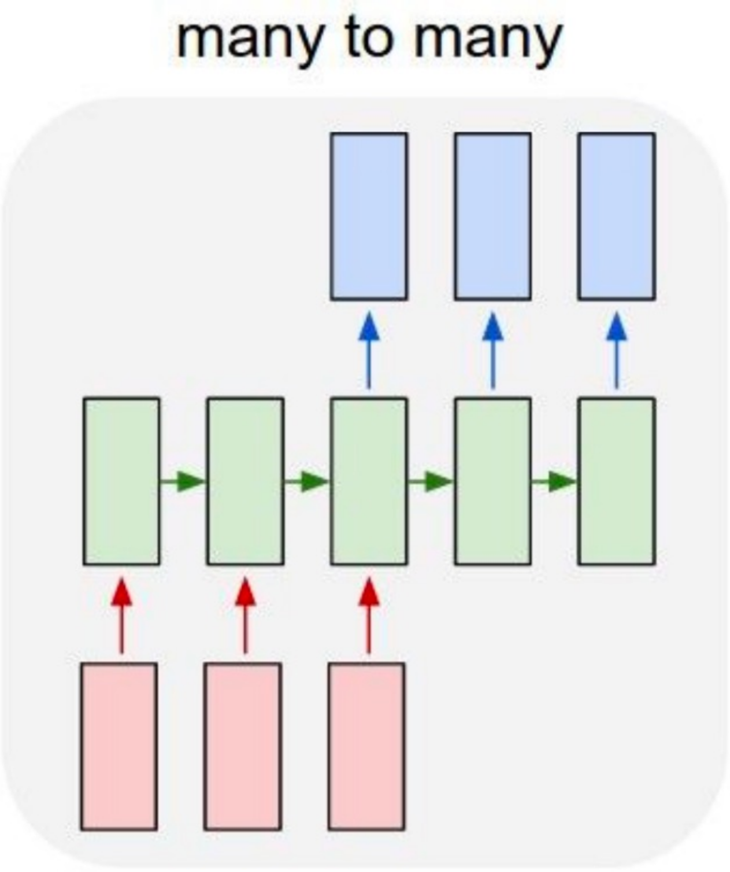
Recurrent neural networks



Vanilla
CNNs

Video data: multiple frames per second

Action recognition

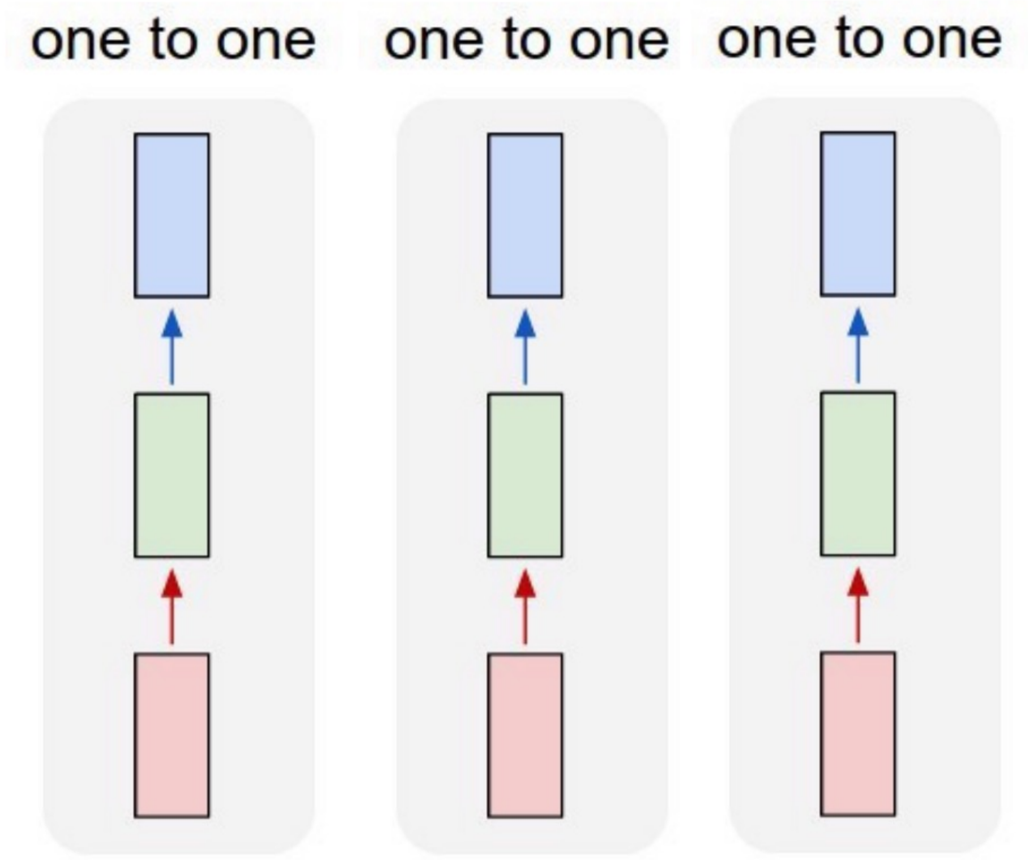


Video data: multiple
frames per second

Recurrent neural networks

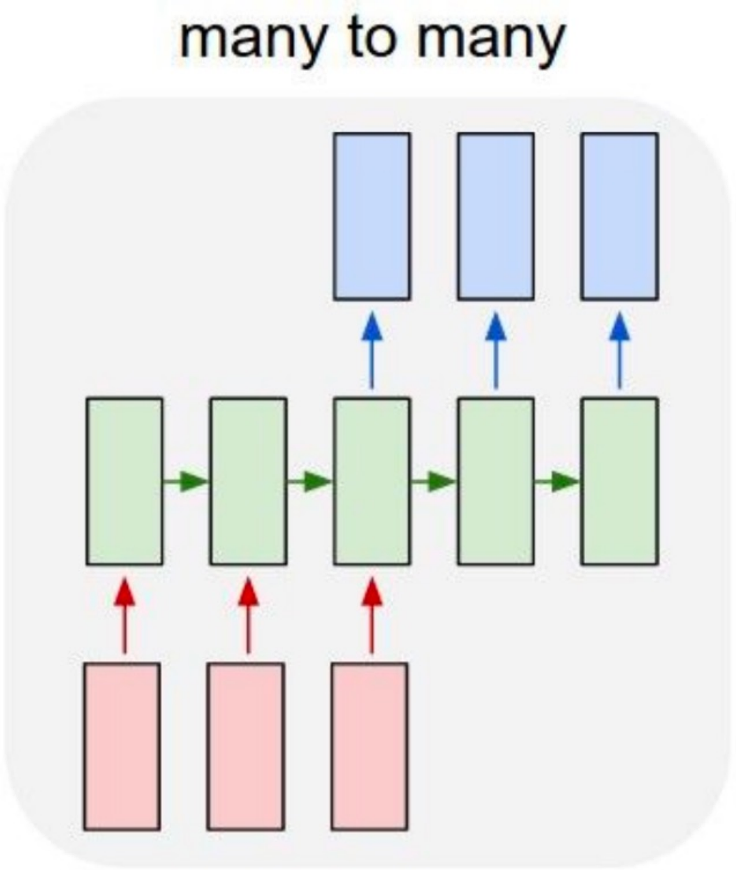
Q: what application?

Vanilla CNNs



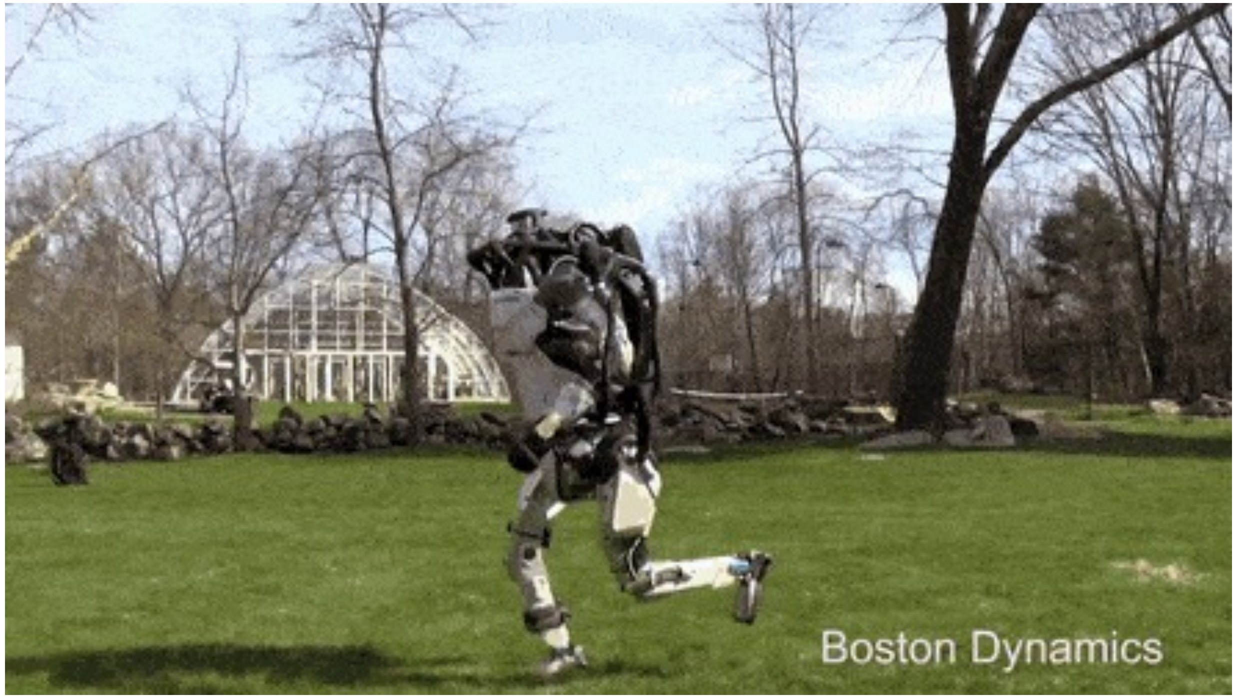
Video data: multiple frames per second

Action recognition



Video data: multiple frames per second

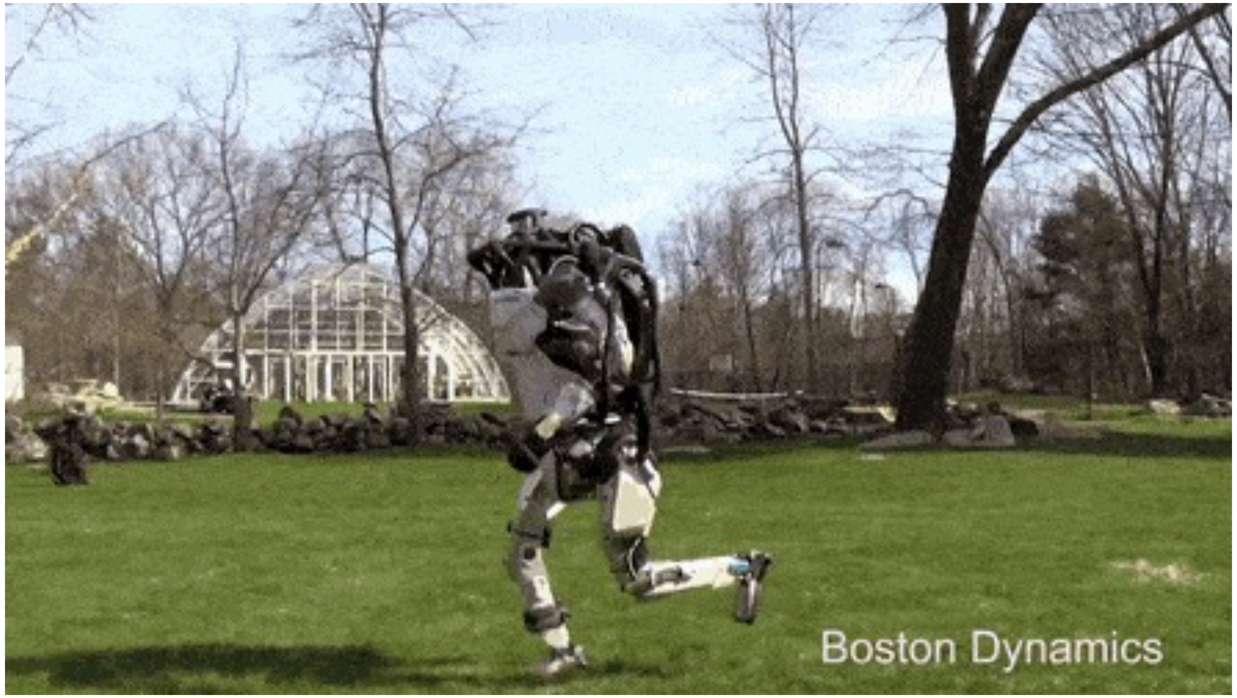
Recurrent neural networks in practice



Q: what is the action?

Image credit: Boston dynamics

Recurrent neural networks in practice

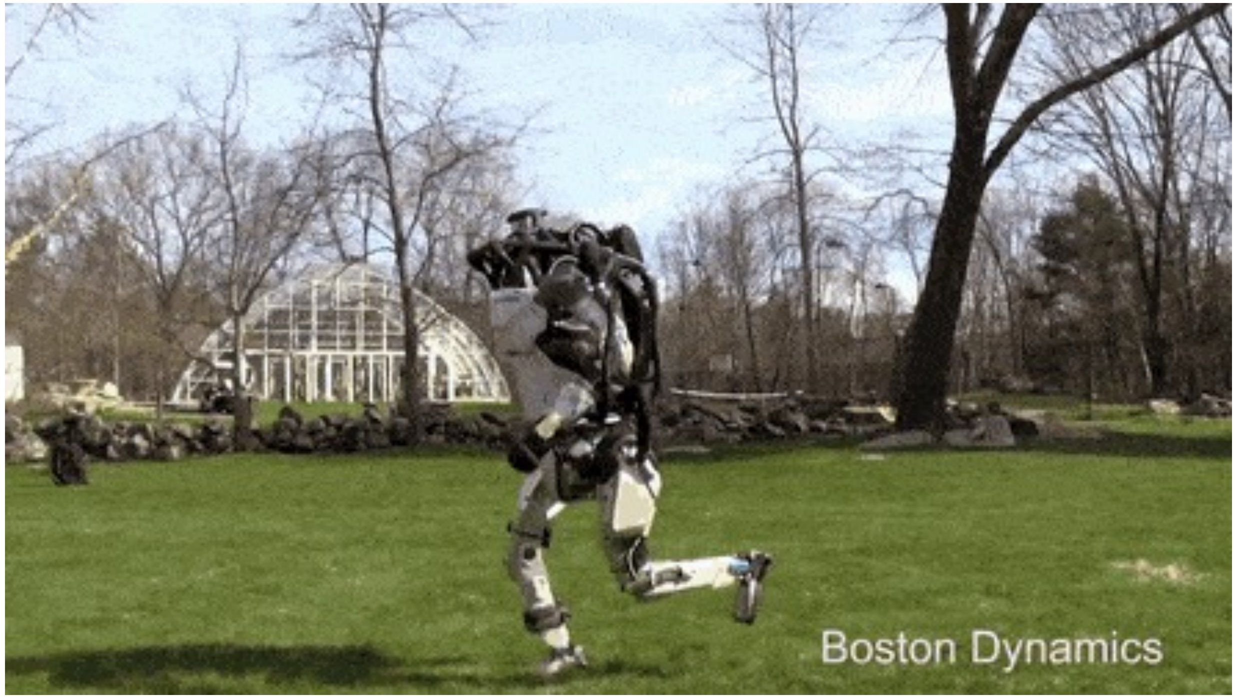


Q: what is the action?

R

Image credit: Boston dynamics

Recurrent neural networks in practice

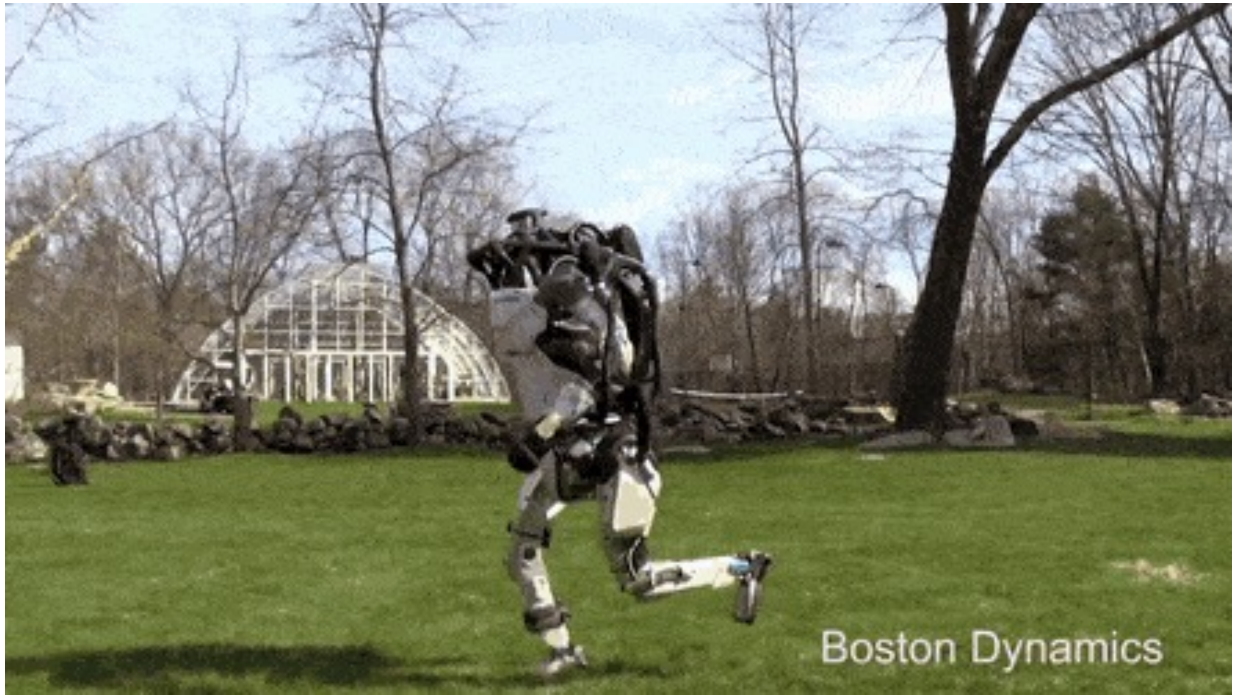


Q: what is the action?

Ru

Image credit: Boston dynamics

Recurrent neural networks in practice

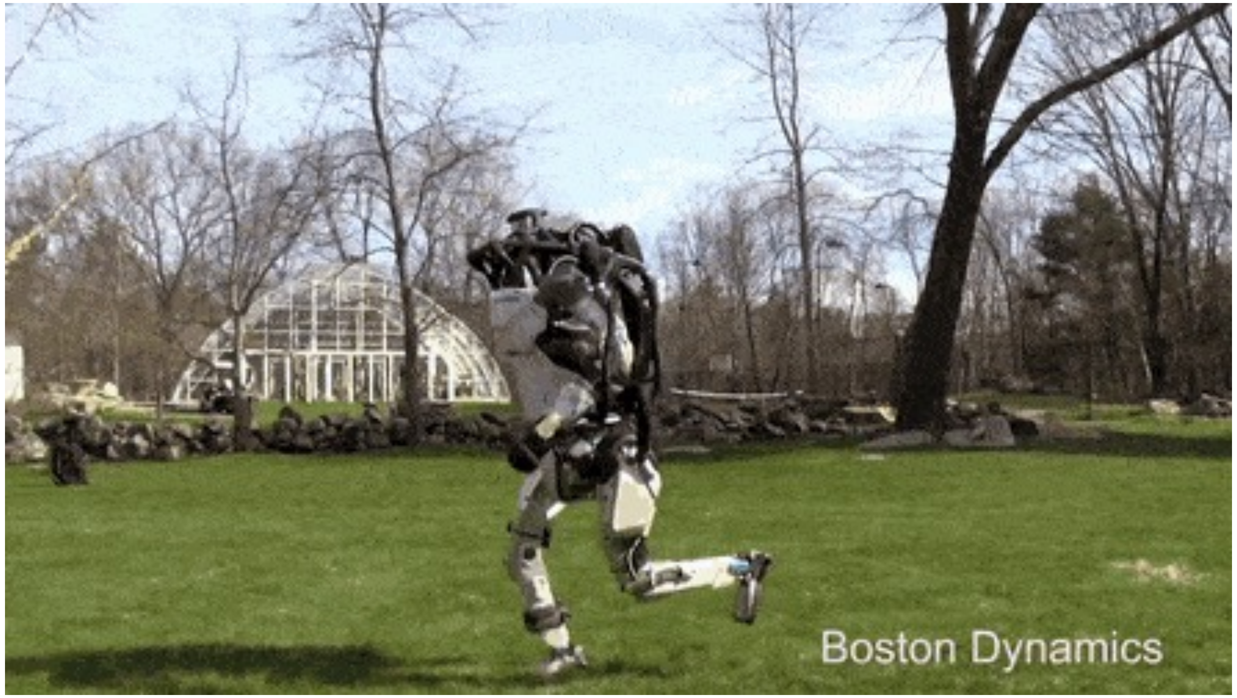


Q: what is the action?

Run

Image credit: Boston dynamics

Recurrent neural networks in practice

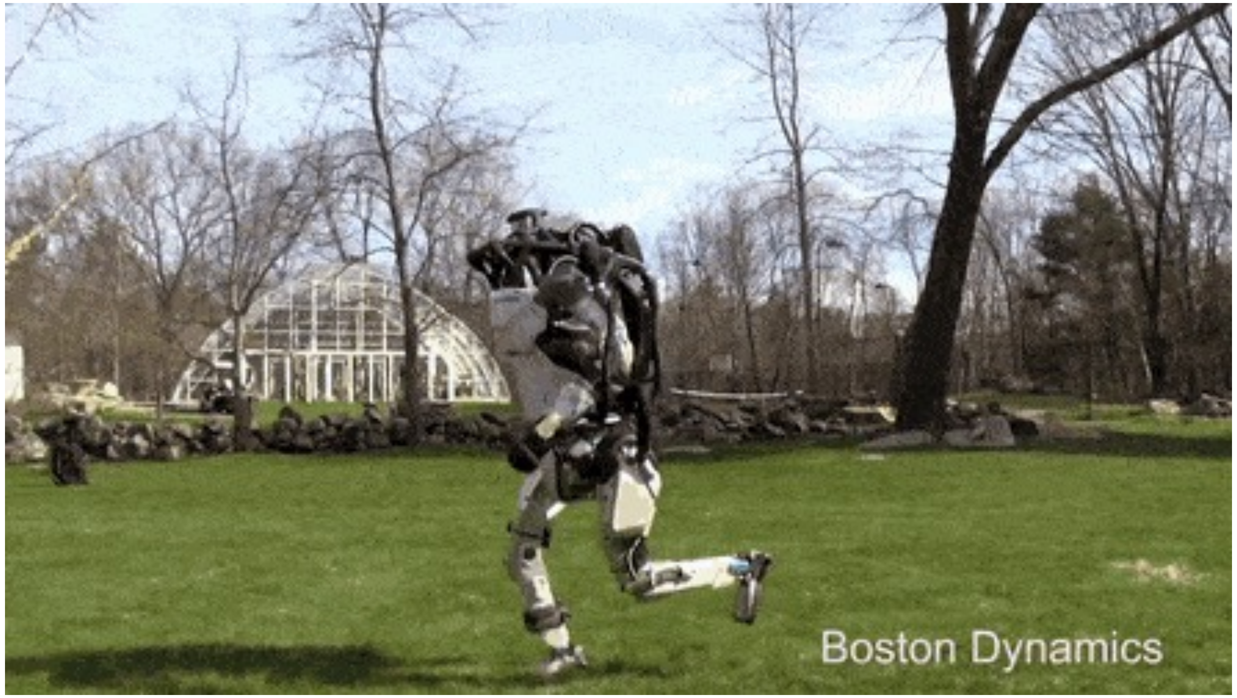


Q: what is the action?

Runn

Image credit: Boston dynamics

Recurrent neural networks in practice

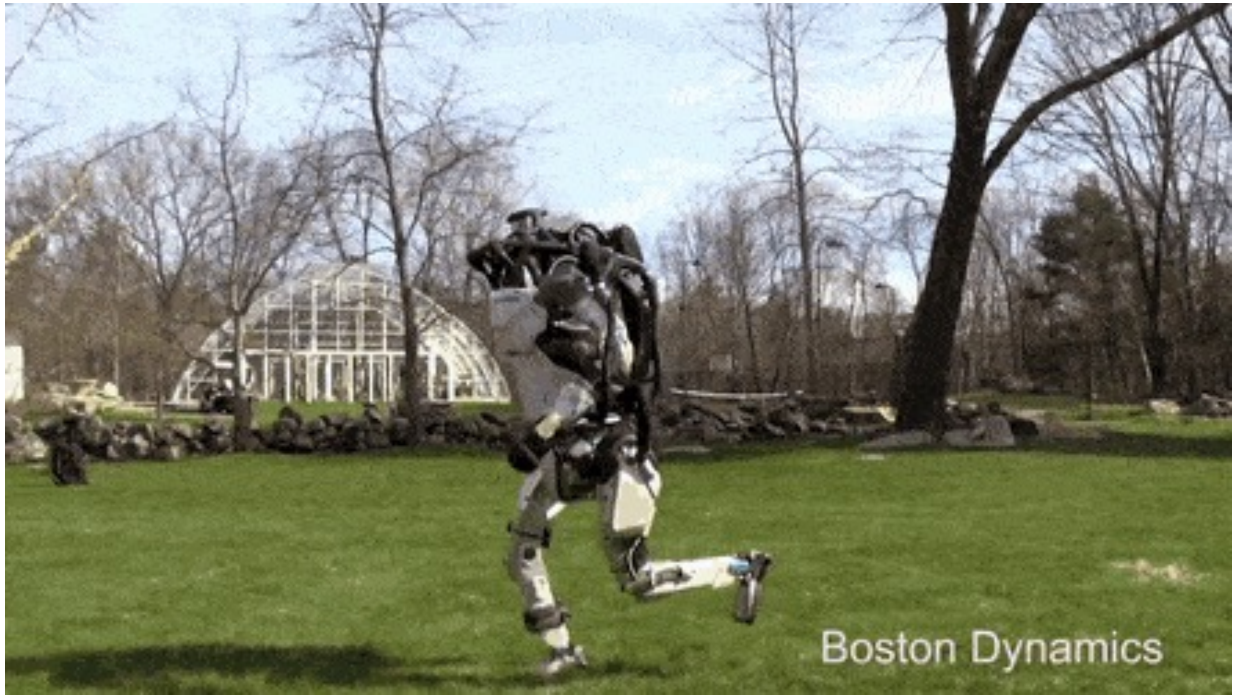


Q: what is the action?

Runni

Image credit: Boston dynamics

Recurrent neural networks in practice

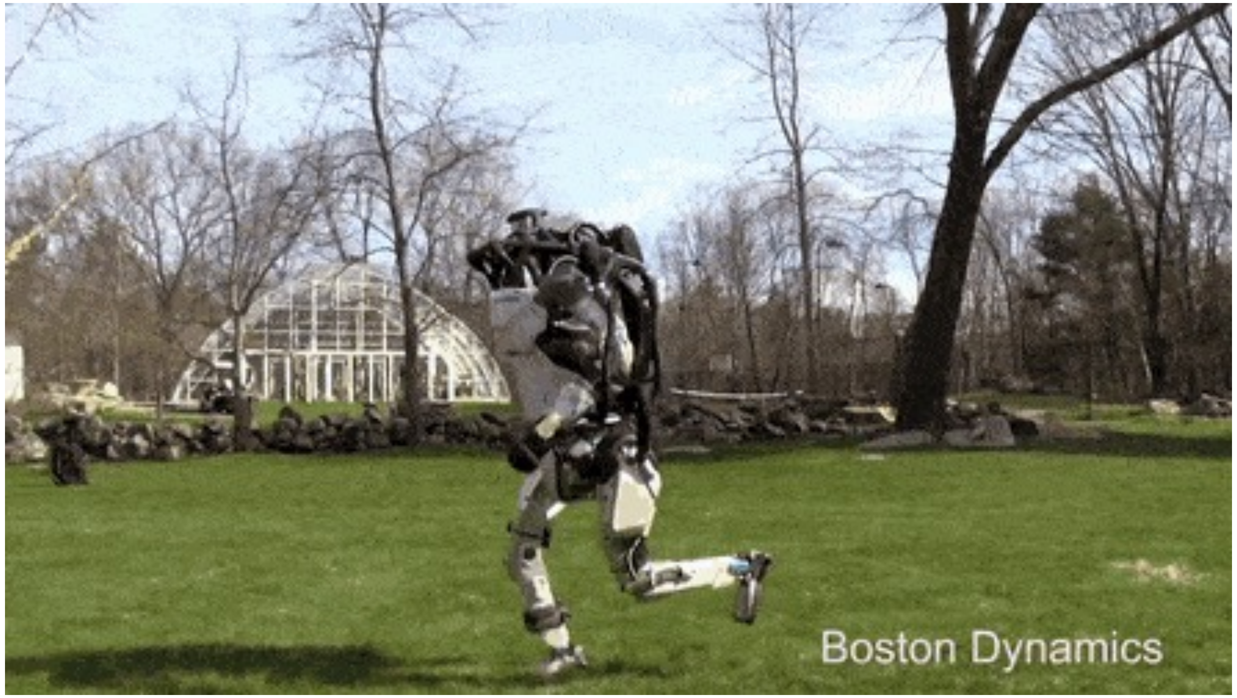


Q: what is the action?

Runnin

Image credit: Boston dynamics

Recurrent neural networks in practice

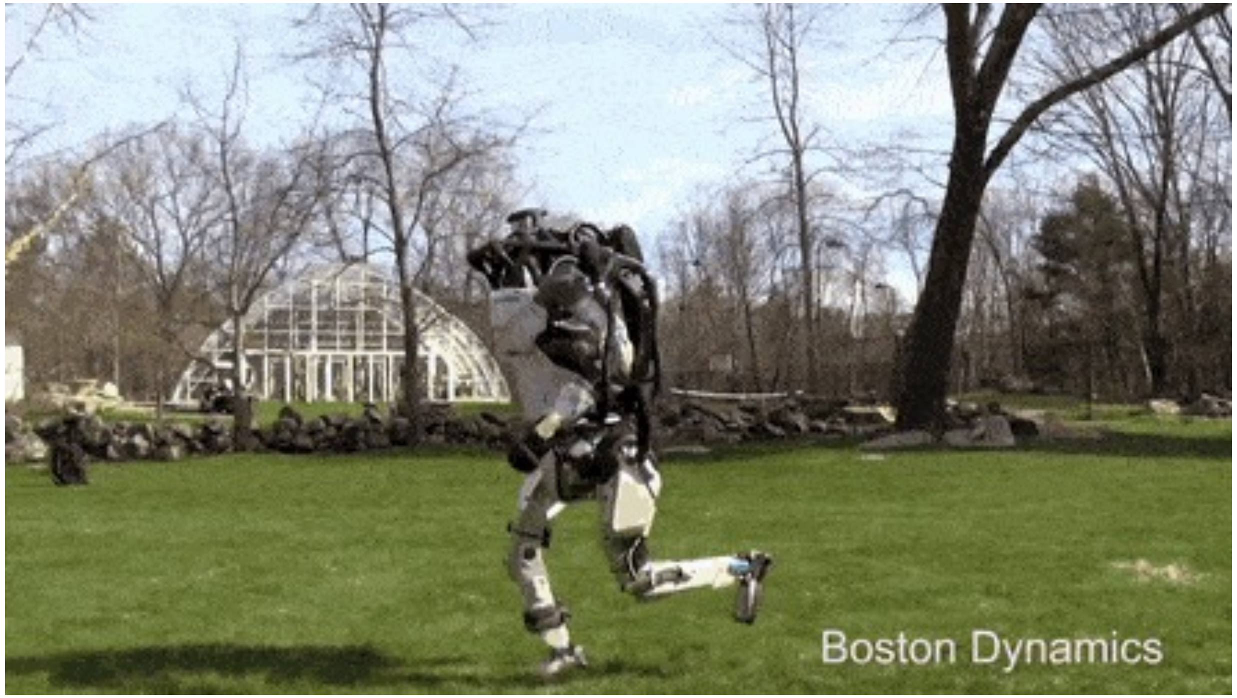


Q: what is the action?

Running

Image credit: Boston dynamics

Recurrent neural networks in practice



Q: what is the action?

Running ← Sequence data

Image credit: Boston dynamics

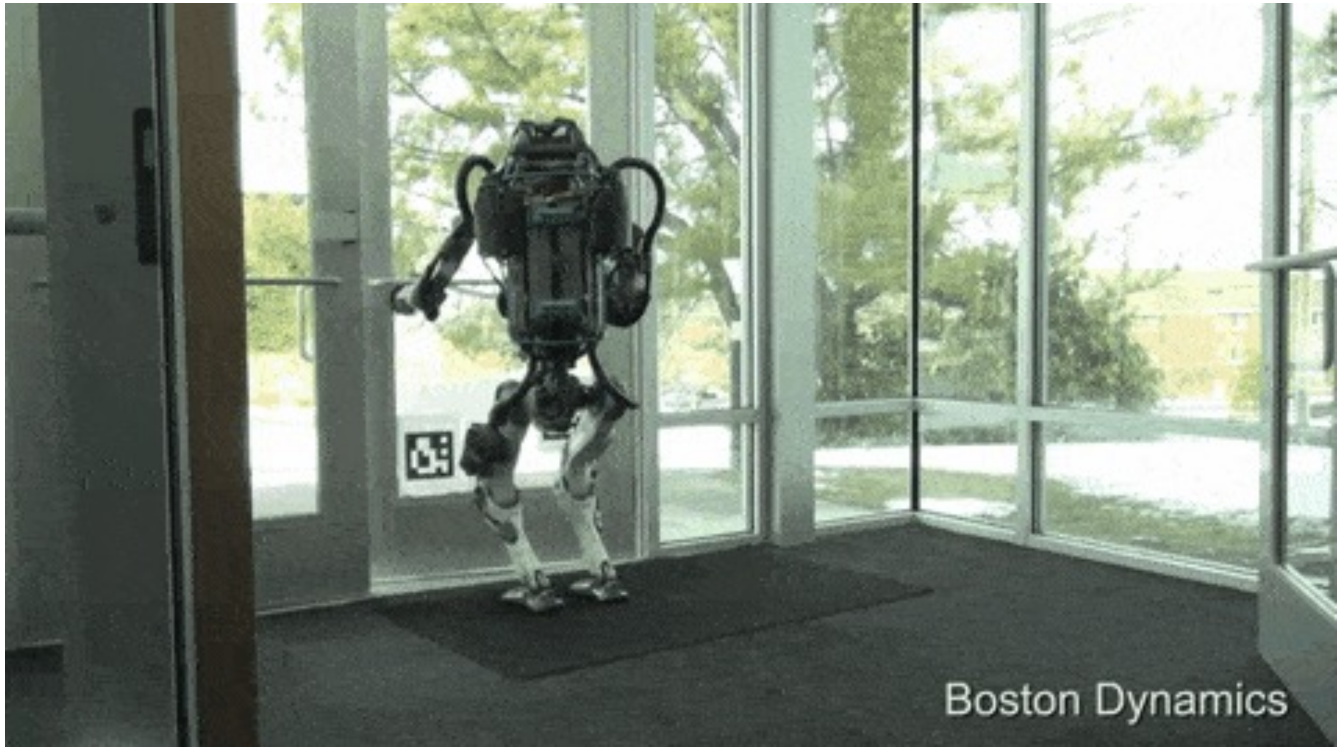
Recurrent neural networks in practice

Q: what is the action?

Opening a door

Image credit: Boston dynamics

Recurrent neural networks in practice



Q: what is the action?

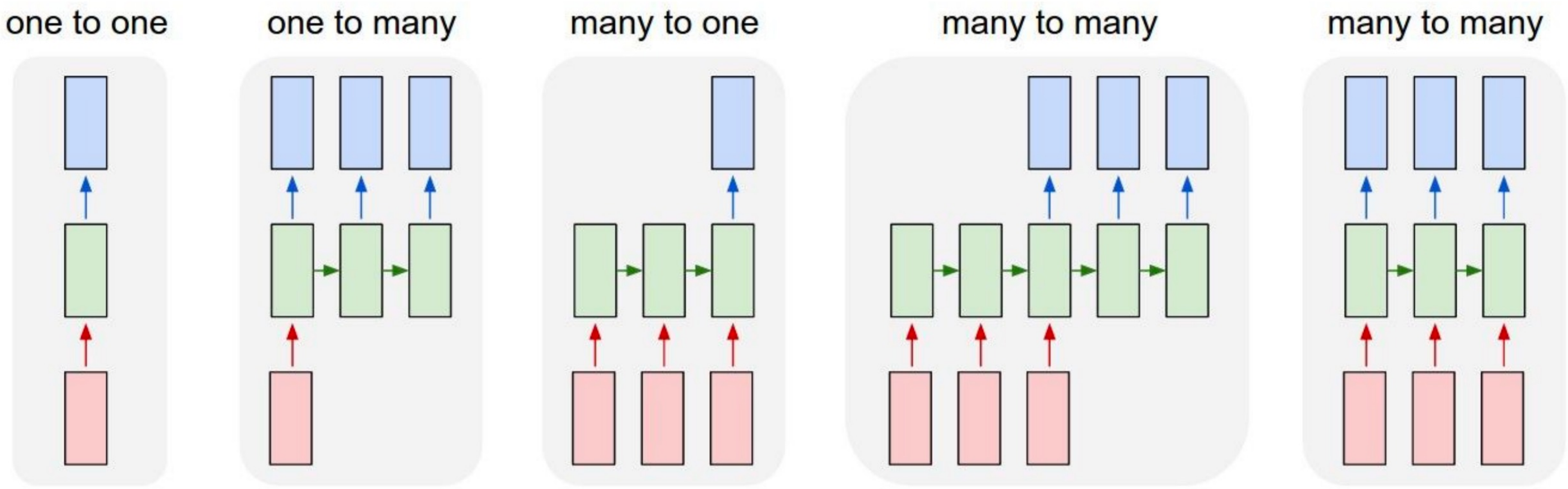
Opening a door

Video captioning:
Generate captions

Image credit: Boston dynamics

Recurrent neural networks

What real applications?



Recurrent neural networks

What real applications?

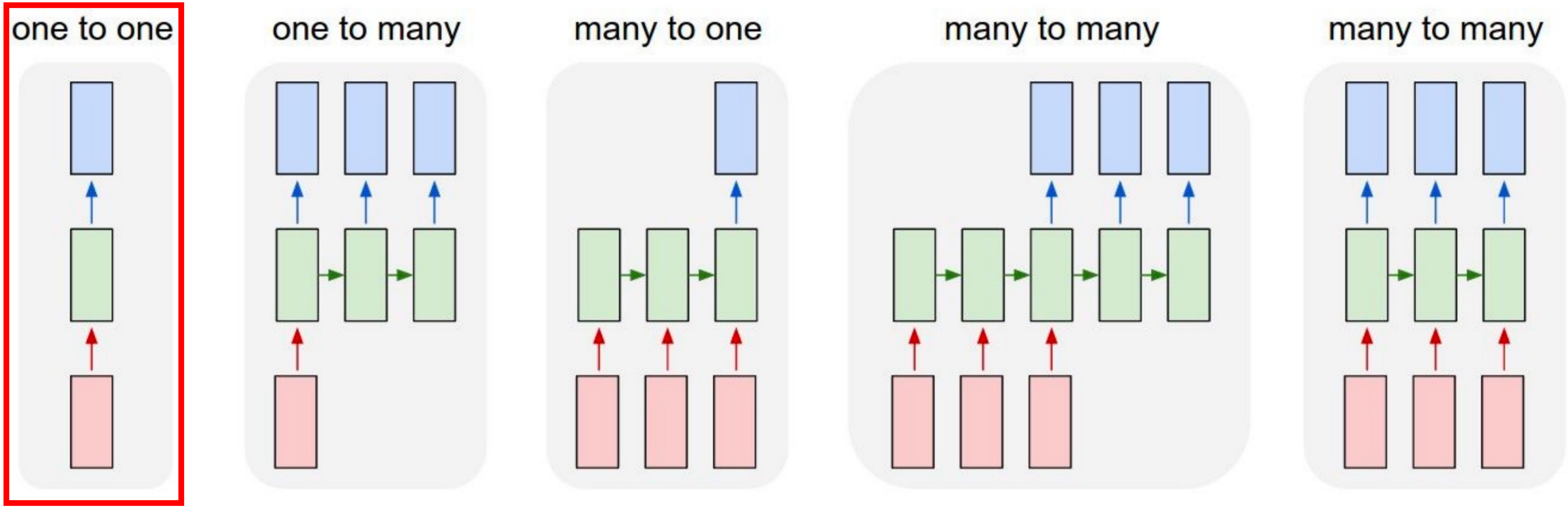
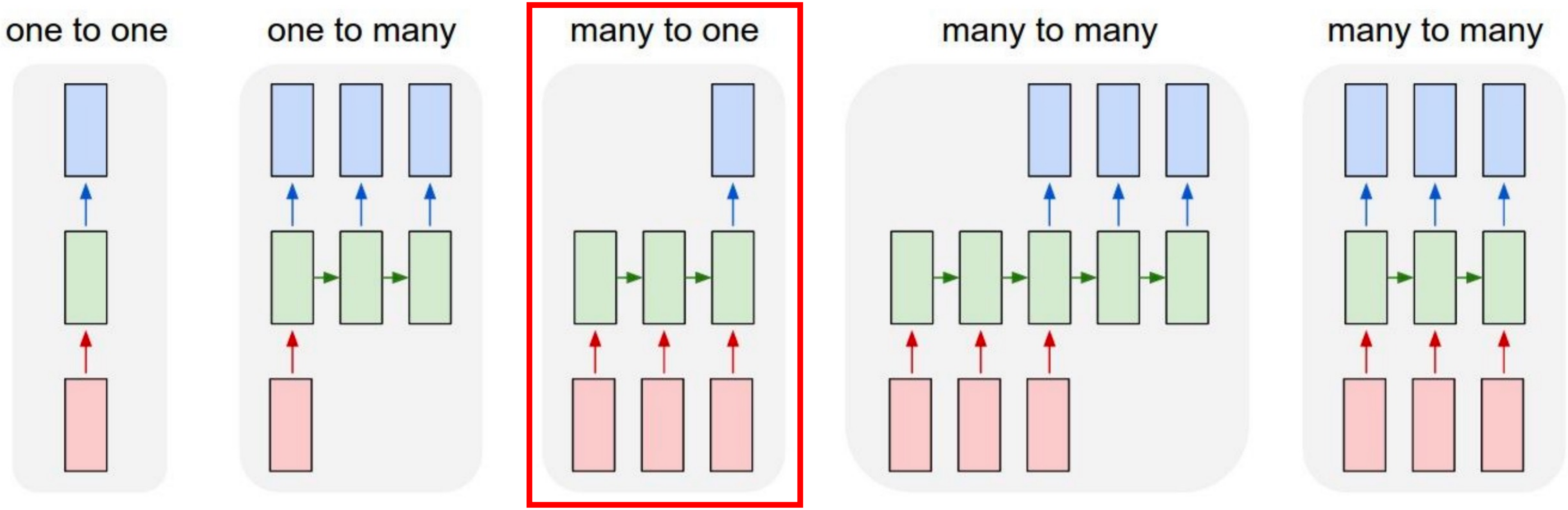


Image classification

Recurrent neural networks

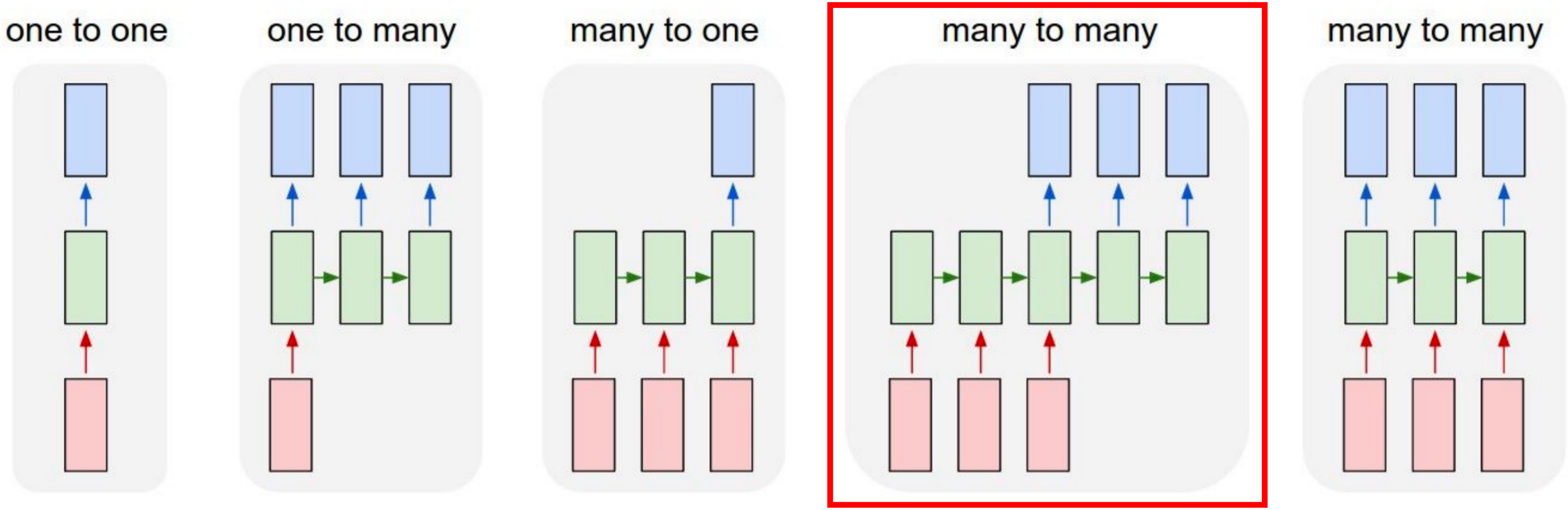
What real applications?



Action recognition

Recurrent neural networks

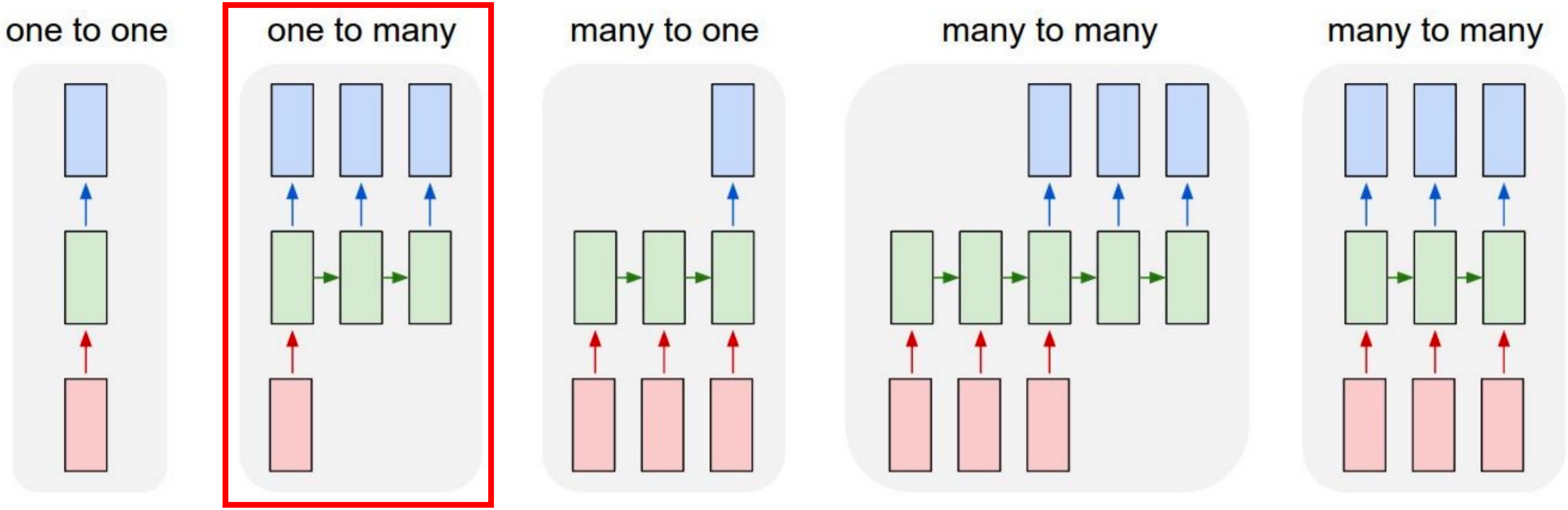
What real applications?



Video captioning

Recurrent neural networks

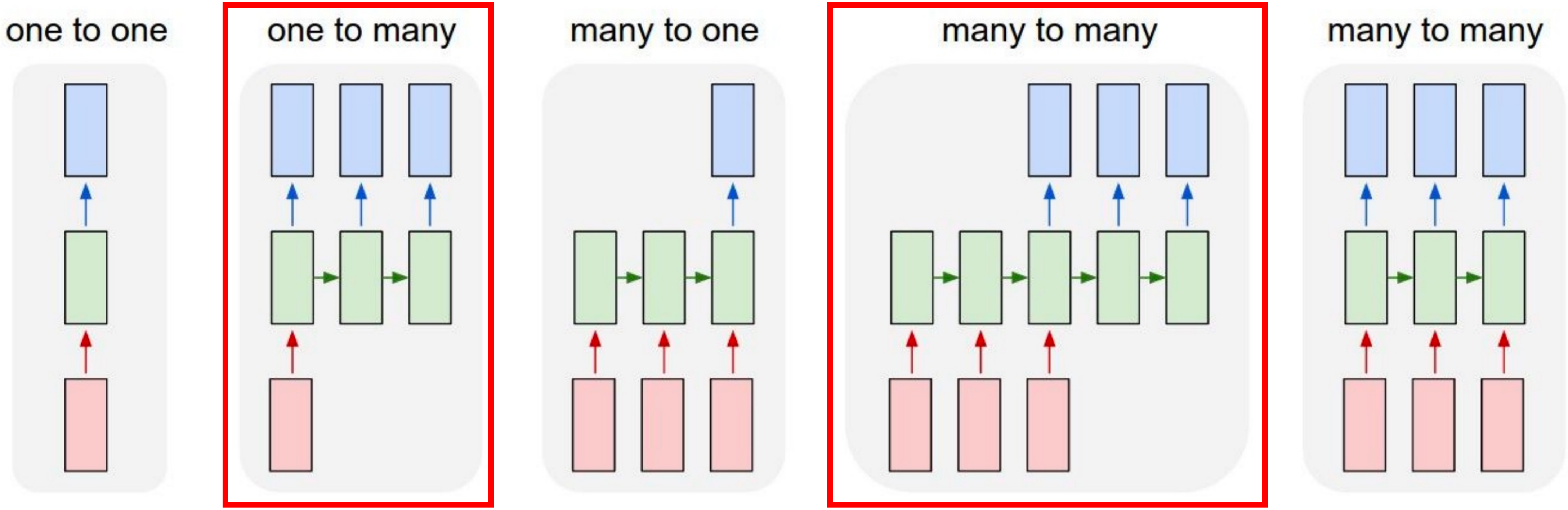
What real applications?



Q: what application?

Recurrent neural networks

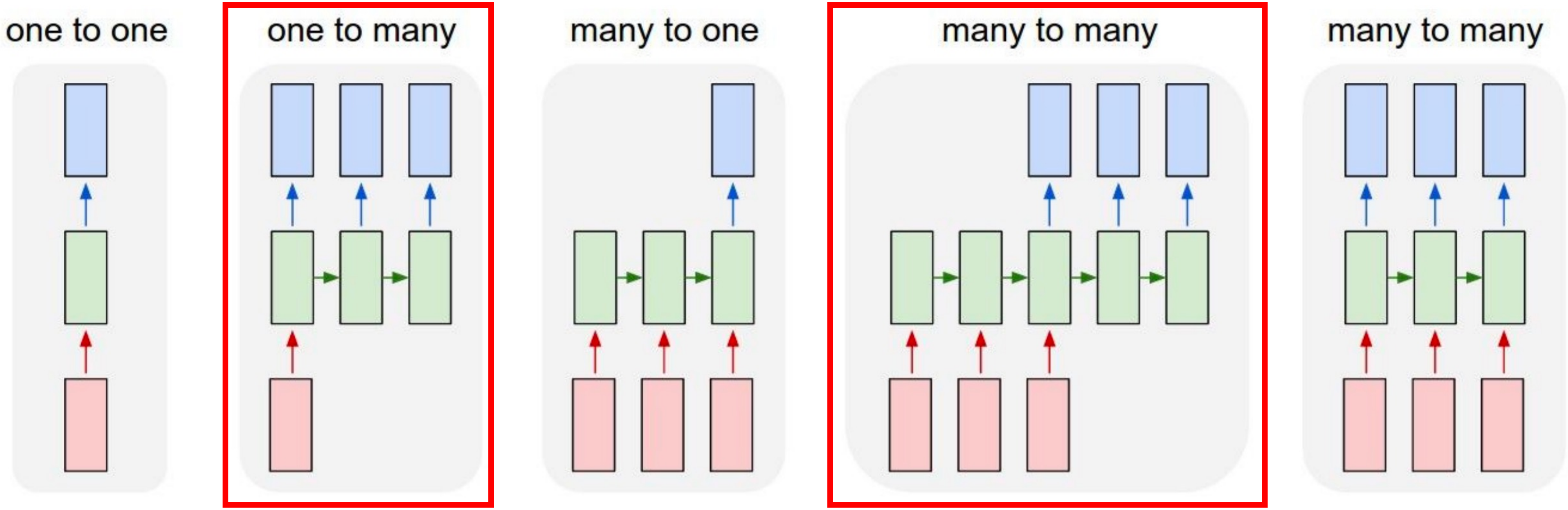
What real applications?



Q: what application?

Recurrent neural networks

What real applications?

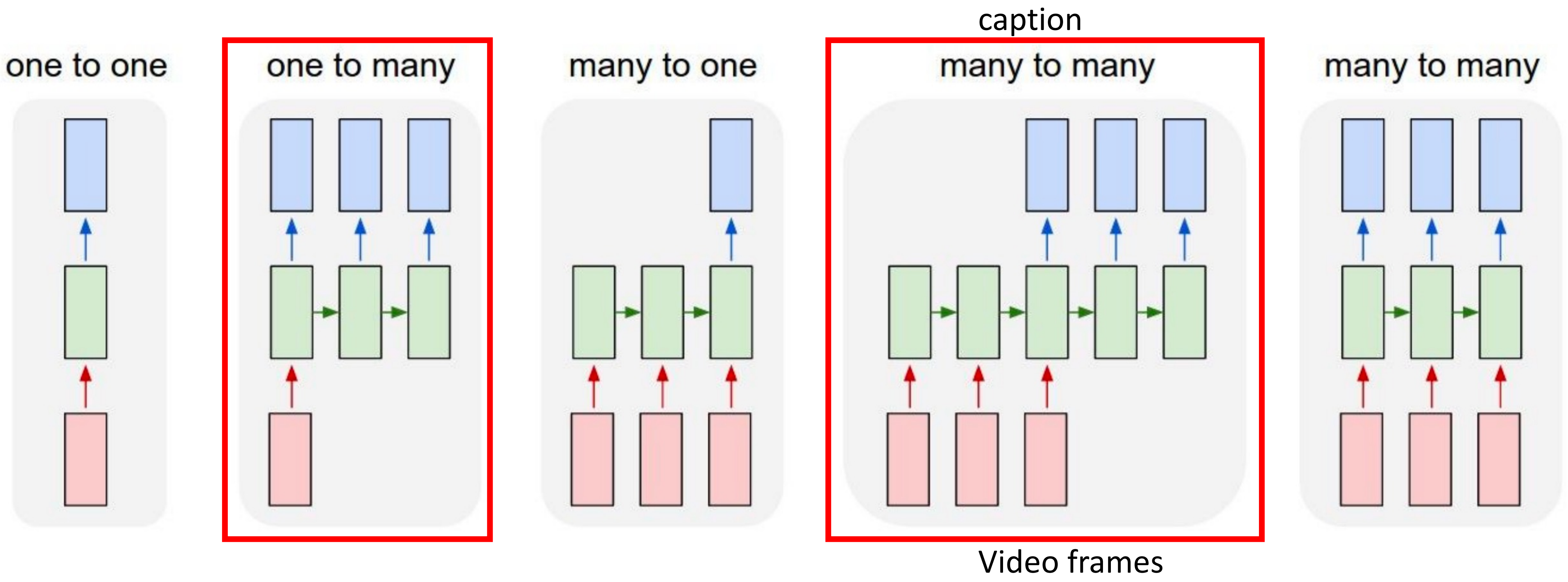


Video frames

Q: what application?

Recurrent neural networks

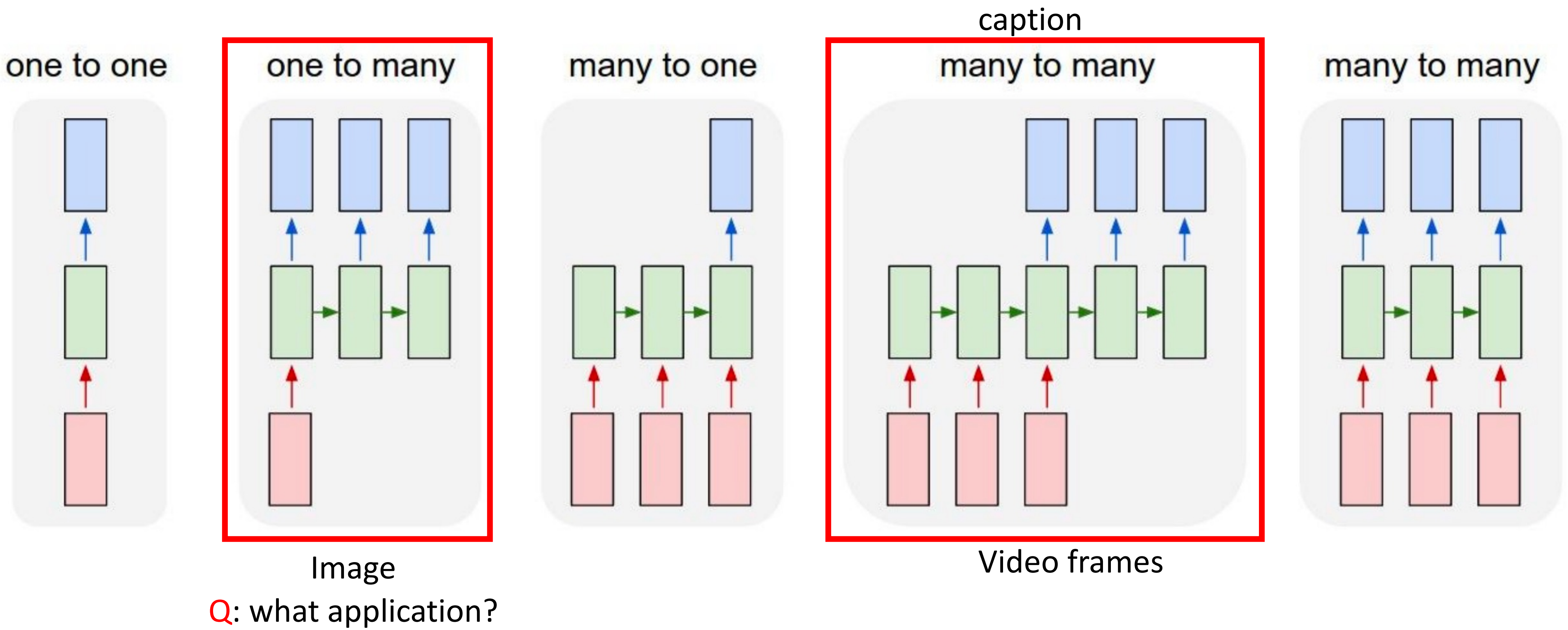
What real applications?



Q: what application?

Recurrent neural networks

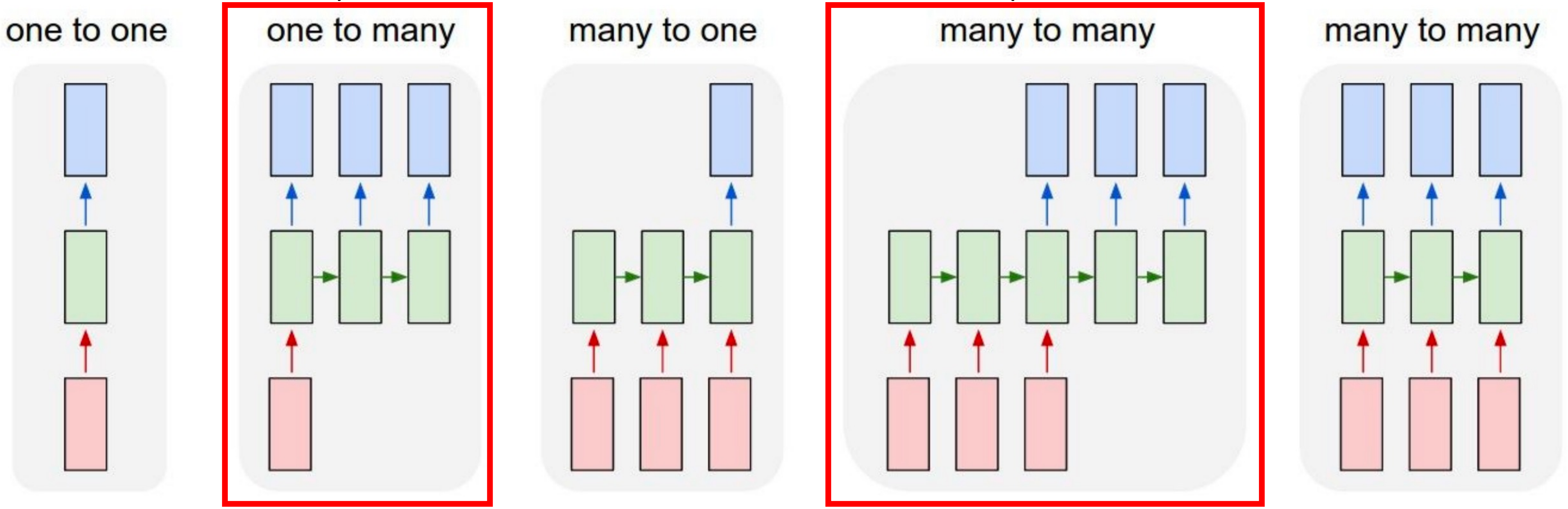
What real applications?



Recurrent neural networks

What real applications?

caption | caption



Image

Q: what application?

Video frames

Image captioning

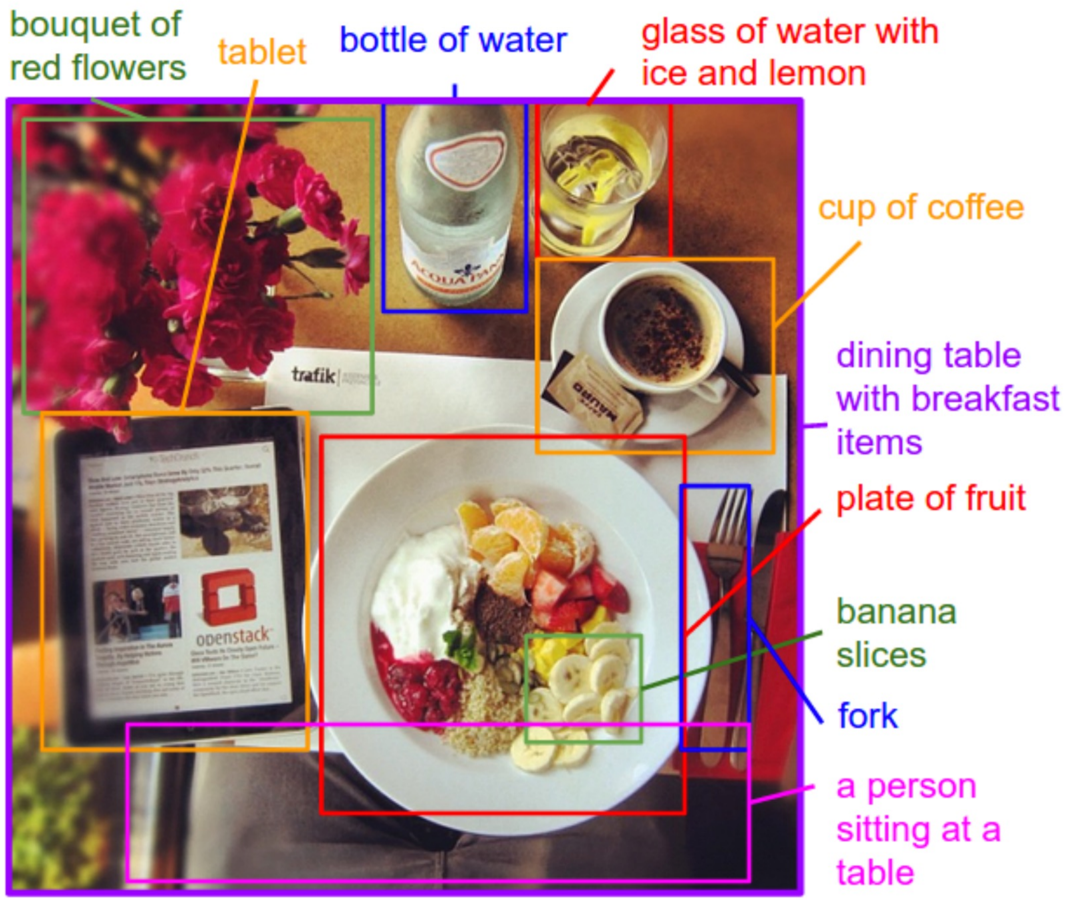


Figure from Karpathy, Andrej, and Li Fei-Fei. "Deep visual-semantic alignments for generating image descriptions." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3128-3137. 2015.

Image captioning

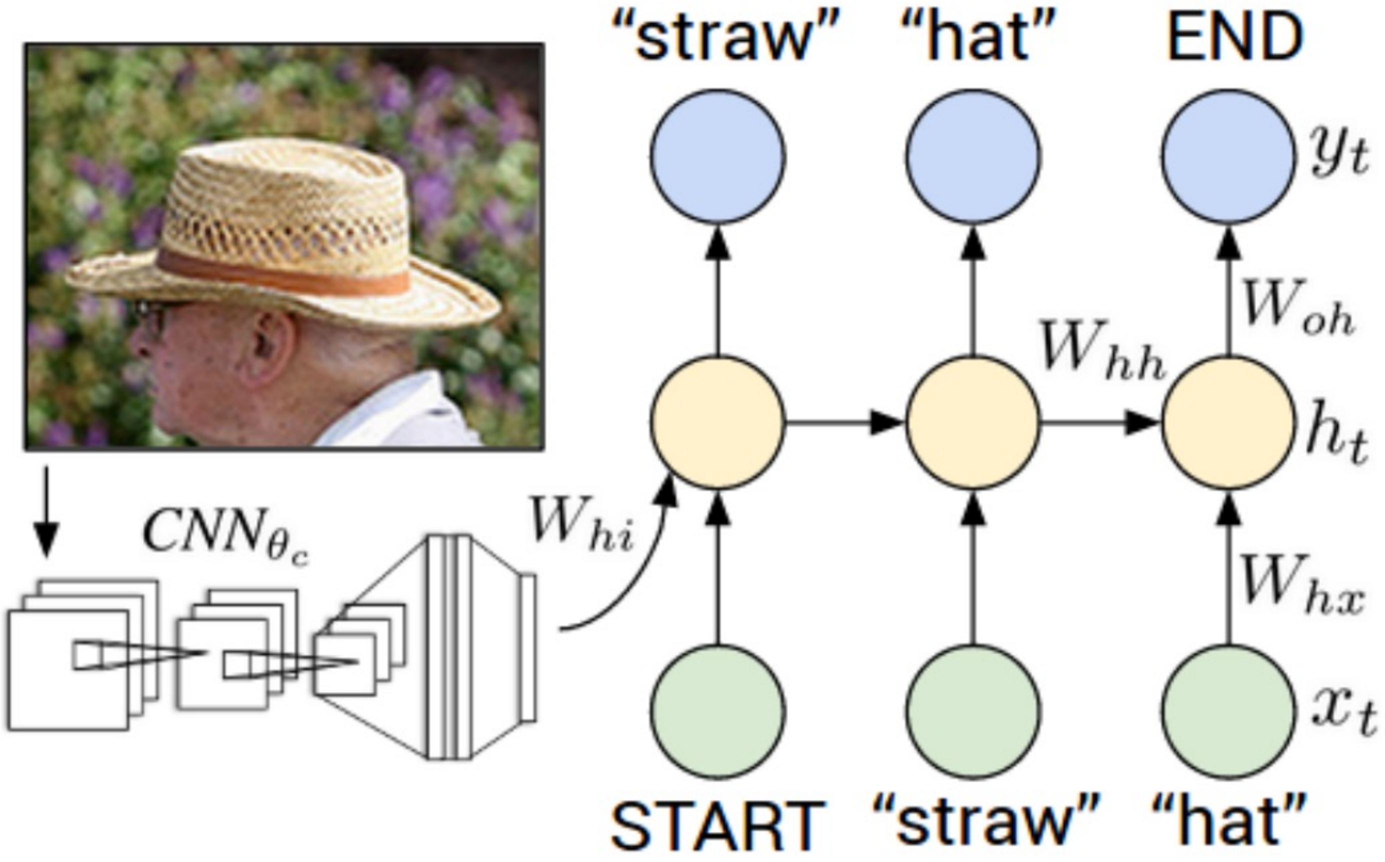
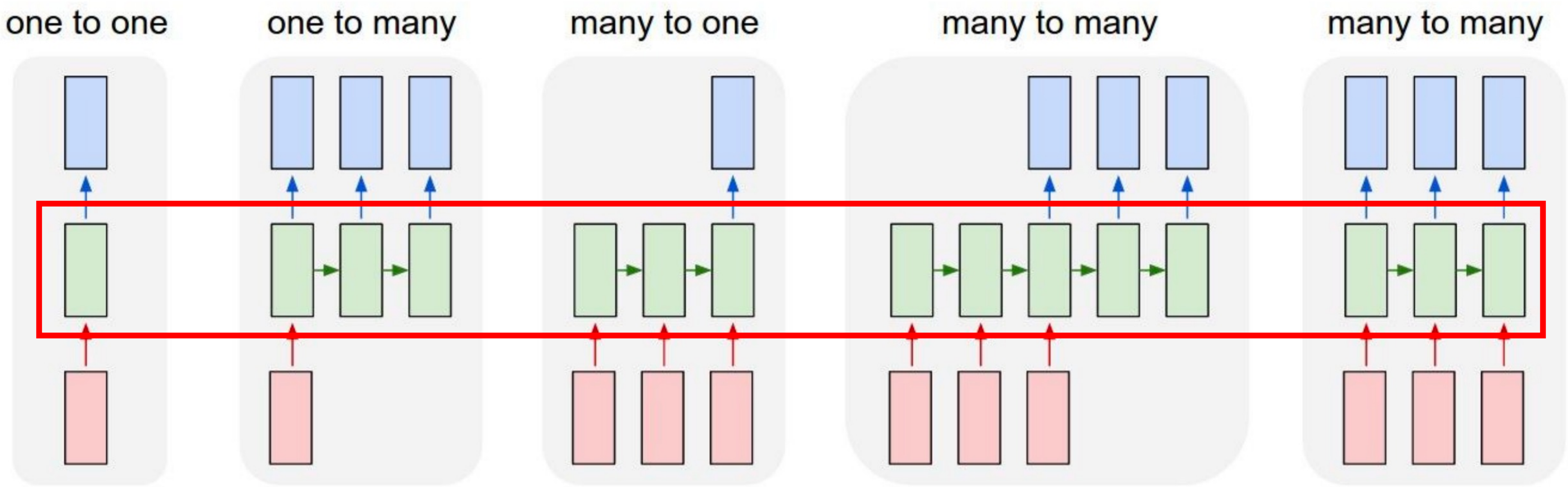


Figure from Karpathy, Andrej, and Li Fei-Fei. "Deep visual-semantic alignments for generating image descriptions." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3128-3137. 2015.

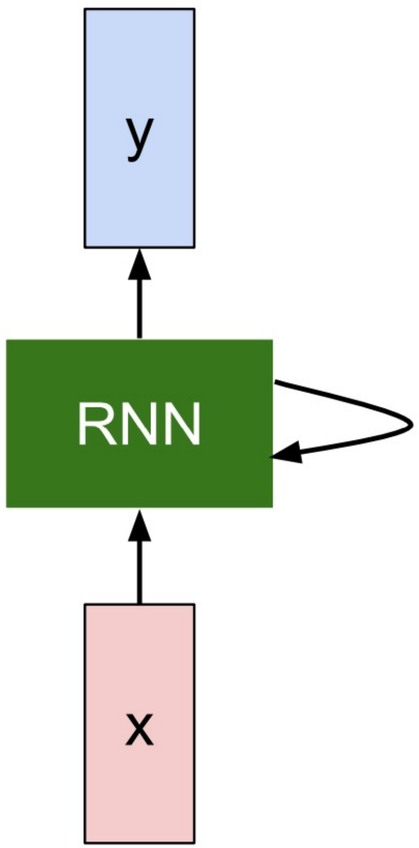
Recurrent neural networks

What’s the key?

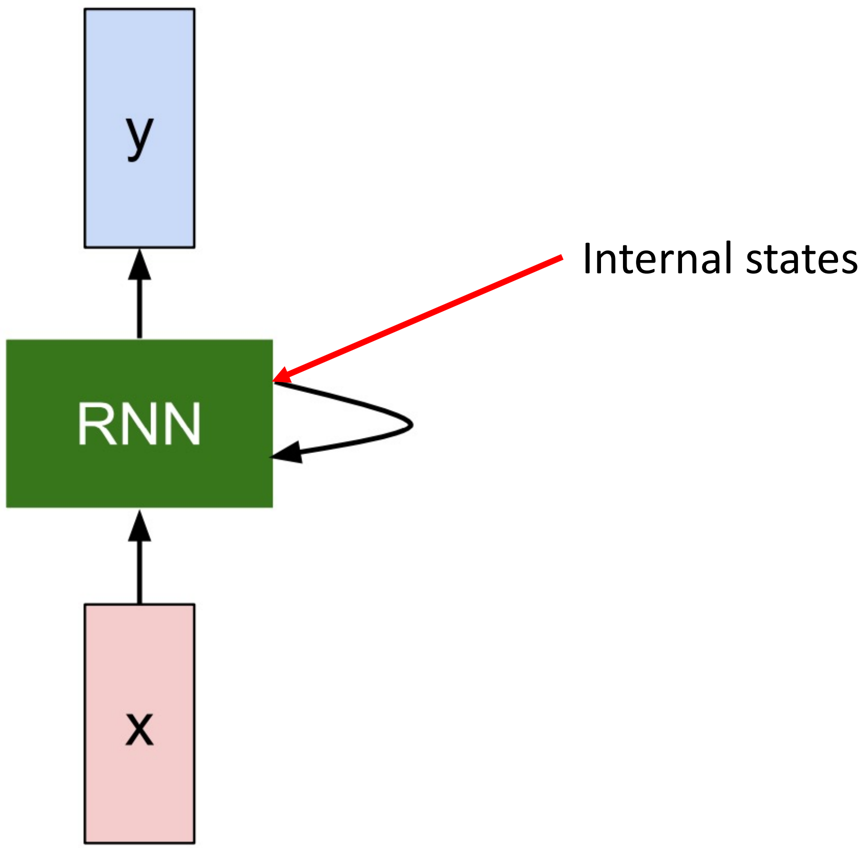


Q: what application?

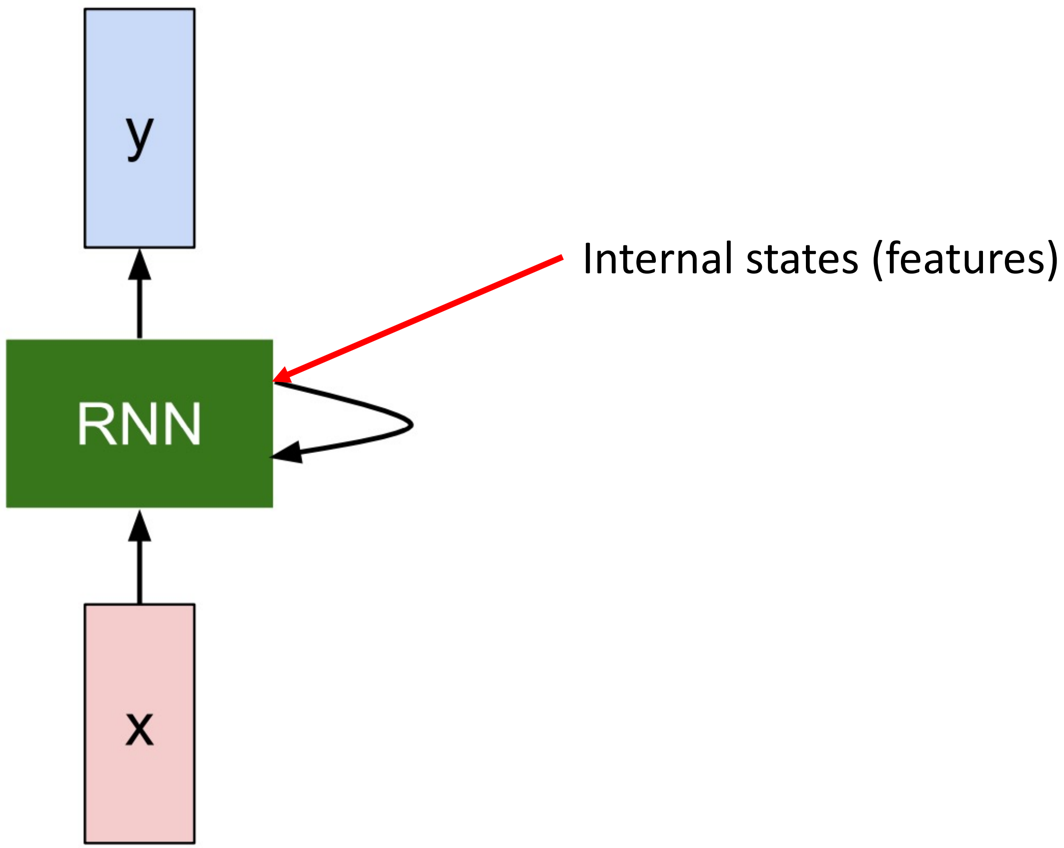
Recurrent neural networks



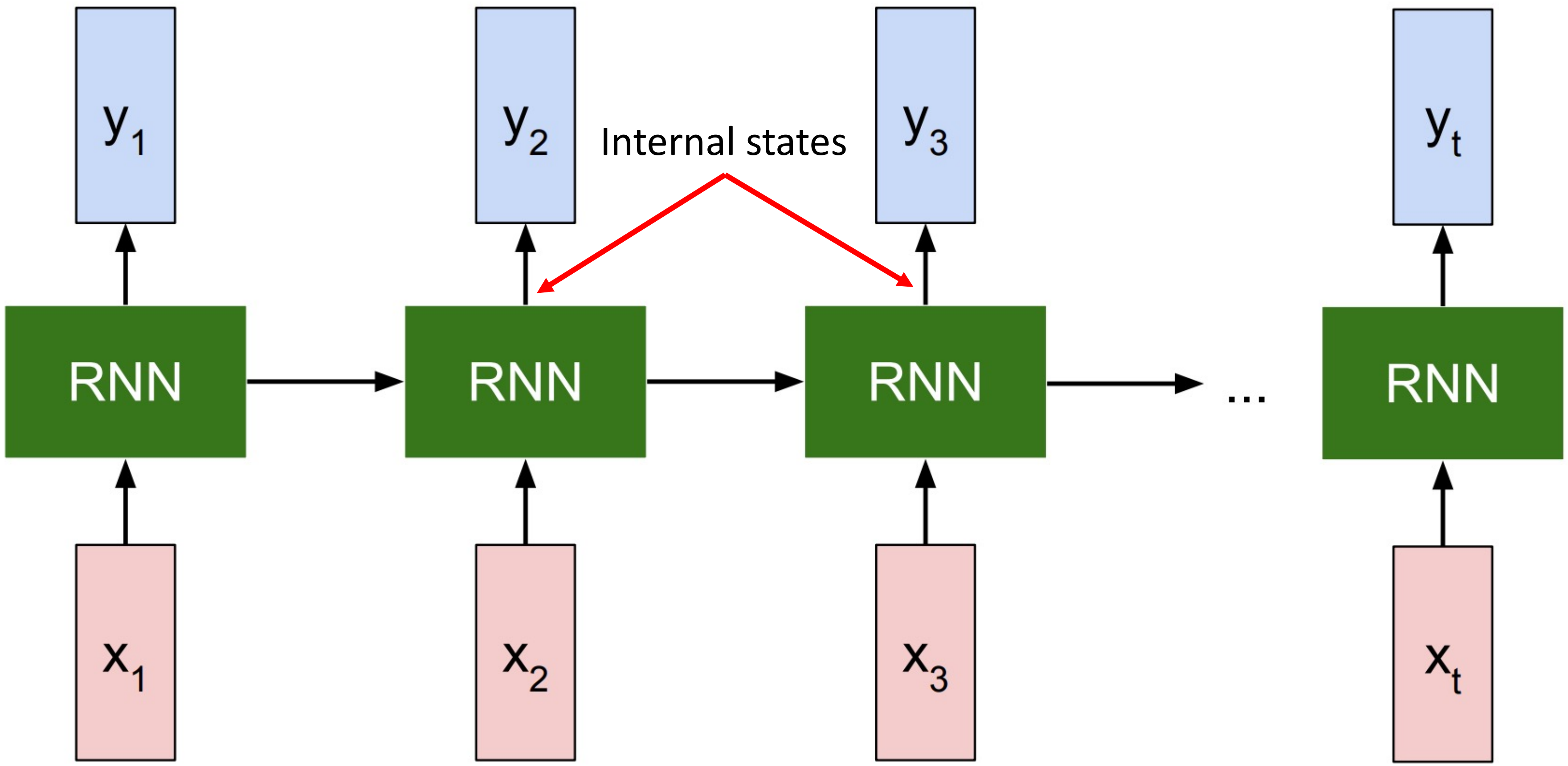
Recurrent neural networks



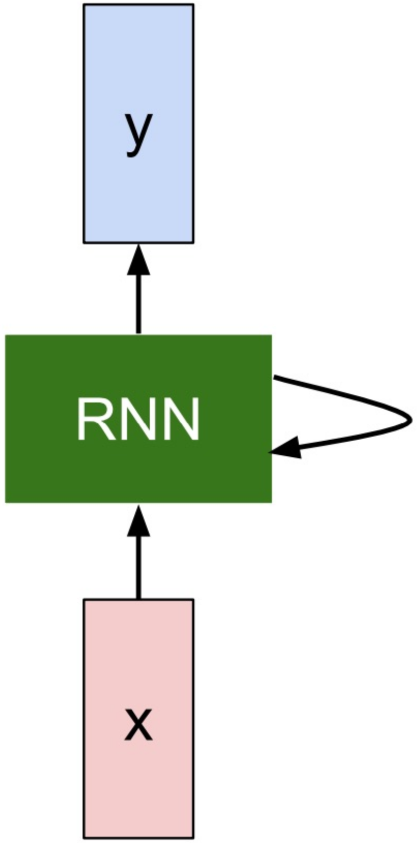
Recurrent neural networks



Recurrent neural networks



Recurrent neural networks



$$\boxed{h_t} = \boxed{f_W}(\boxed{h_{t-1}}, \boxed{x_t})$$

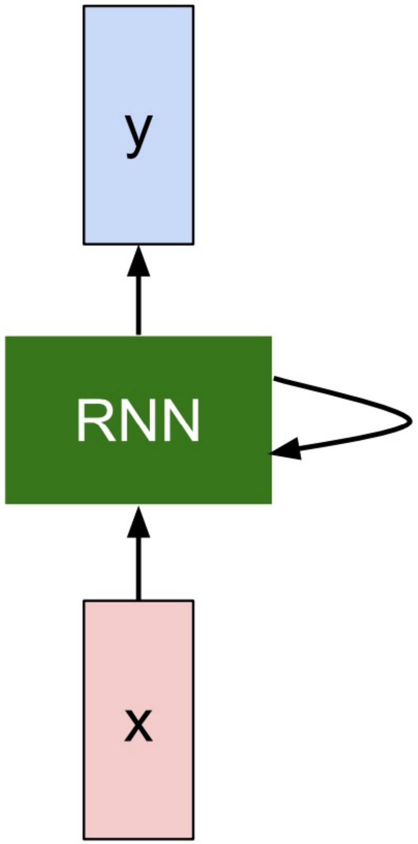
new state

old state

input vector at
some time step

some function
with parameters W

Recurrent neural networks



$$\begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}$$

$$\boxed{h_t} = \boxed{f_W}(\boxed{h_{t-1}}, \boxed{x_t})$$

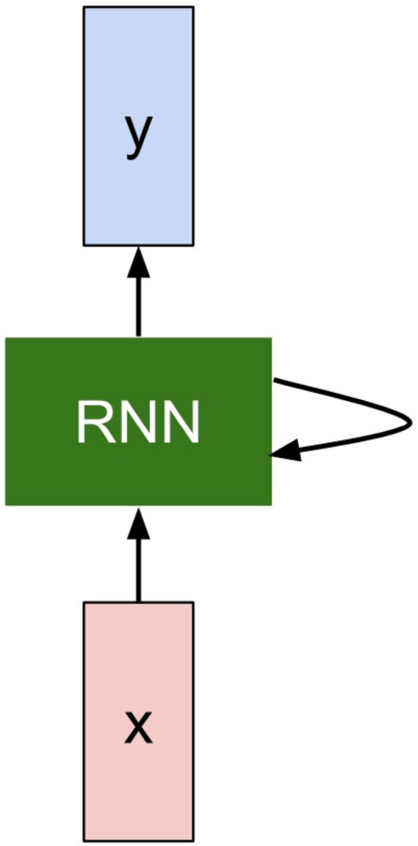
new state

some function
with parameters W

old state

input vector at
some time step

Recurrent neural networks



$$\left[W * \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix} \right]$$

$$\boxed{h_t} = \boxed{f_W}(\boxed{h_{t-1}}, \boxed{x_t})$$

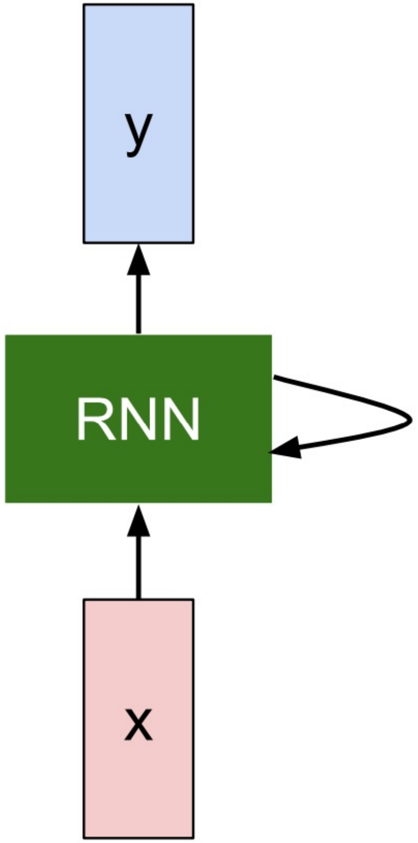
new state

old state

input vector at
some time step

some function
with parameters W

Recurrent neural networks



$$h_t = f\left[\mathrm{W} * \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}\right]$$

$$\boxed{h_t} = \boxed{f_W}(\boxed{h_{t-1}}, \boxed{x_t})$$

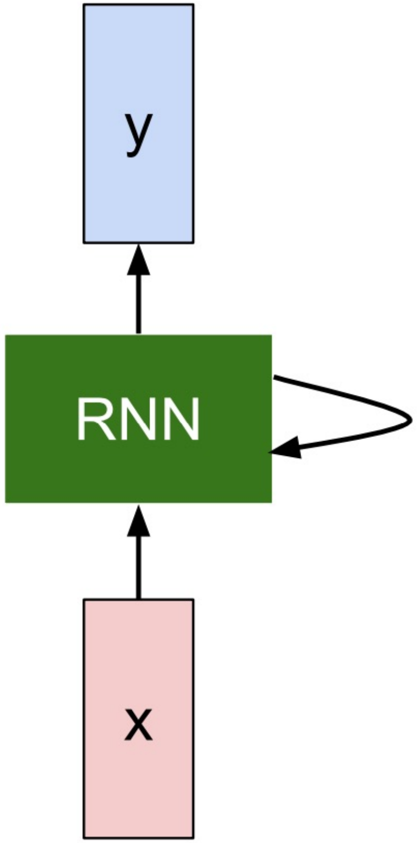
new state

old state

input vector at some time step

some function with parameters W

Recurrent neural networks



$$h_t = f\left[\mathrm{W} * \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix}\right]$$

$$f = \tanh(\cdot)$$

$$\boxed{h_t} = \boxed{f_W}(\boxed{h_{t-1}}, \boxed{x_t})$$

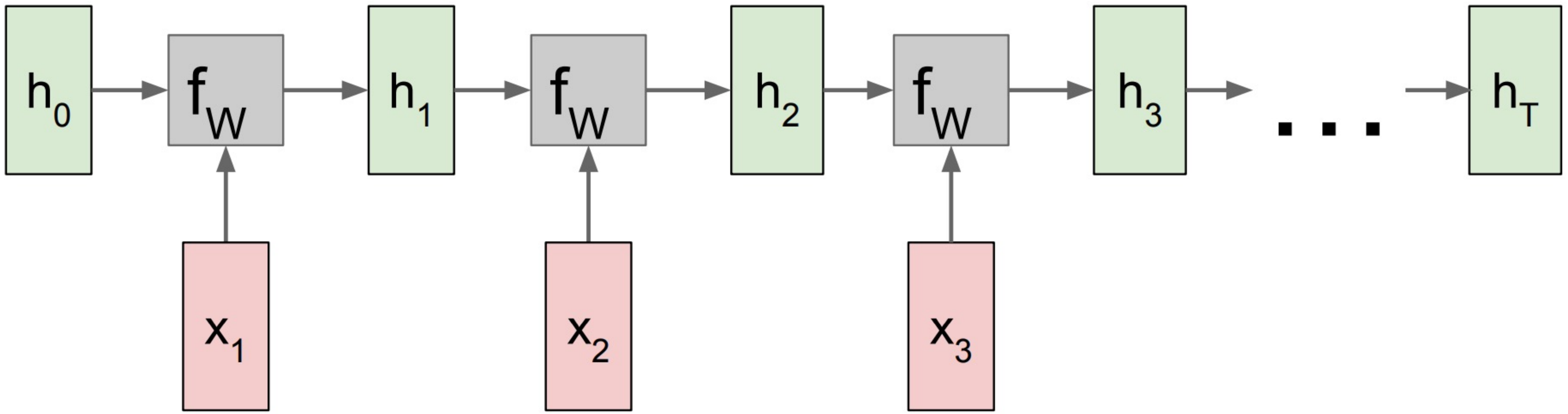
new state

old state

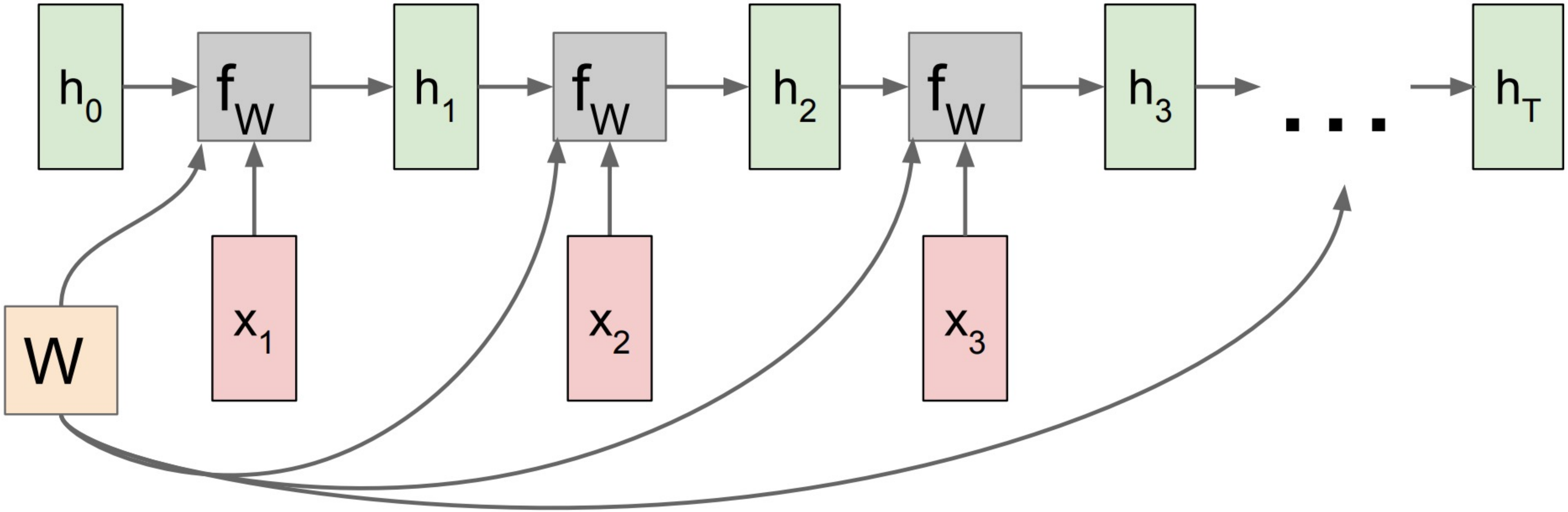
input vector at
some time step

some function
with parameters W

Recurrent neural networks

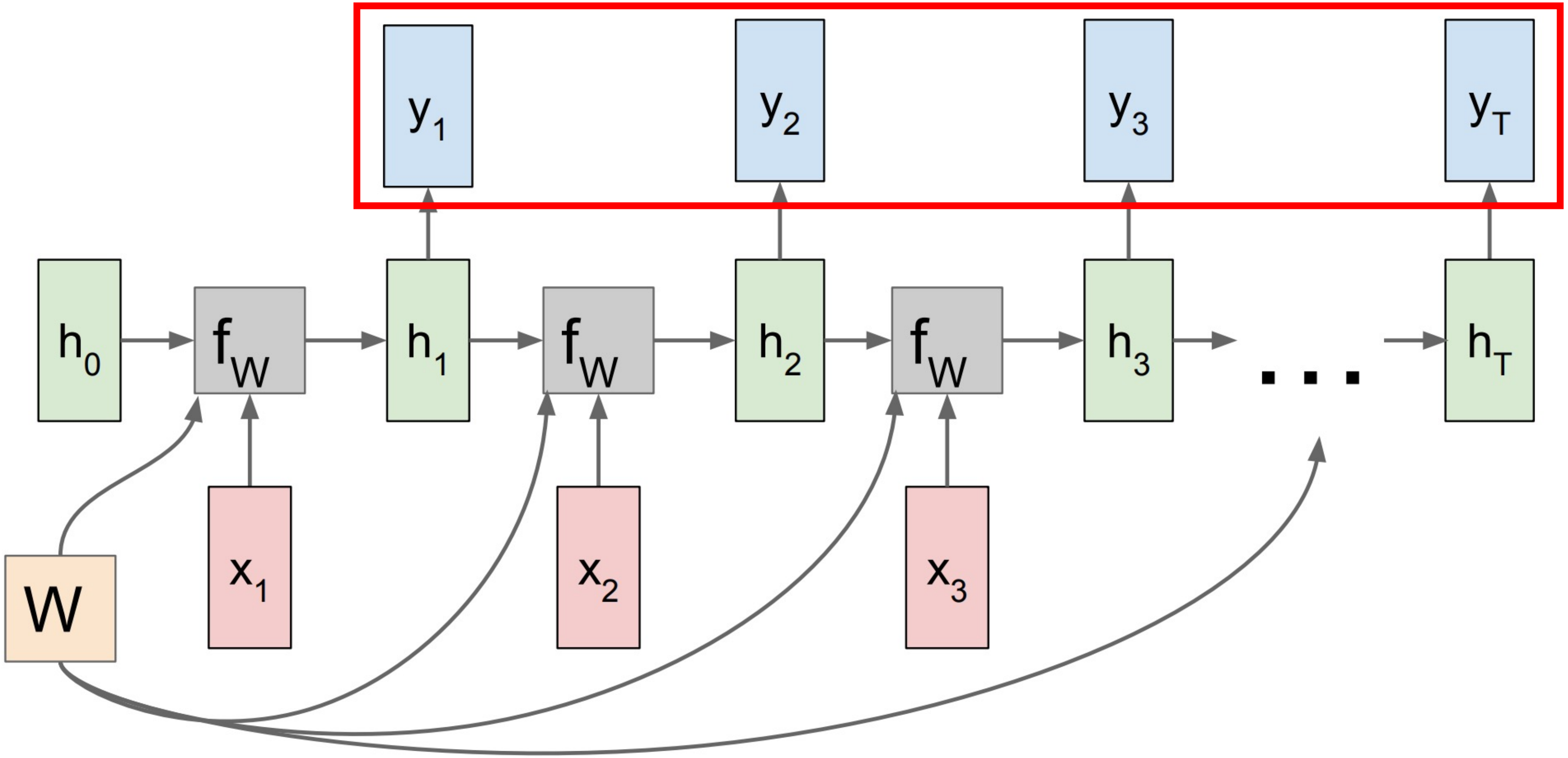


Recurrent neural networks



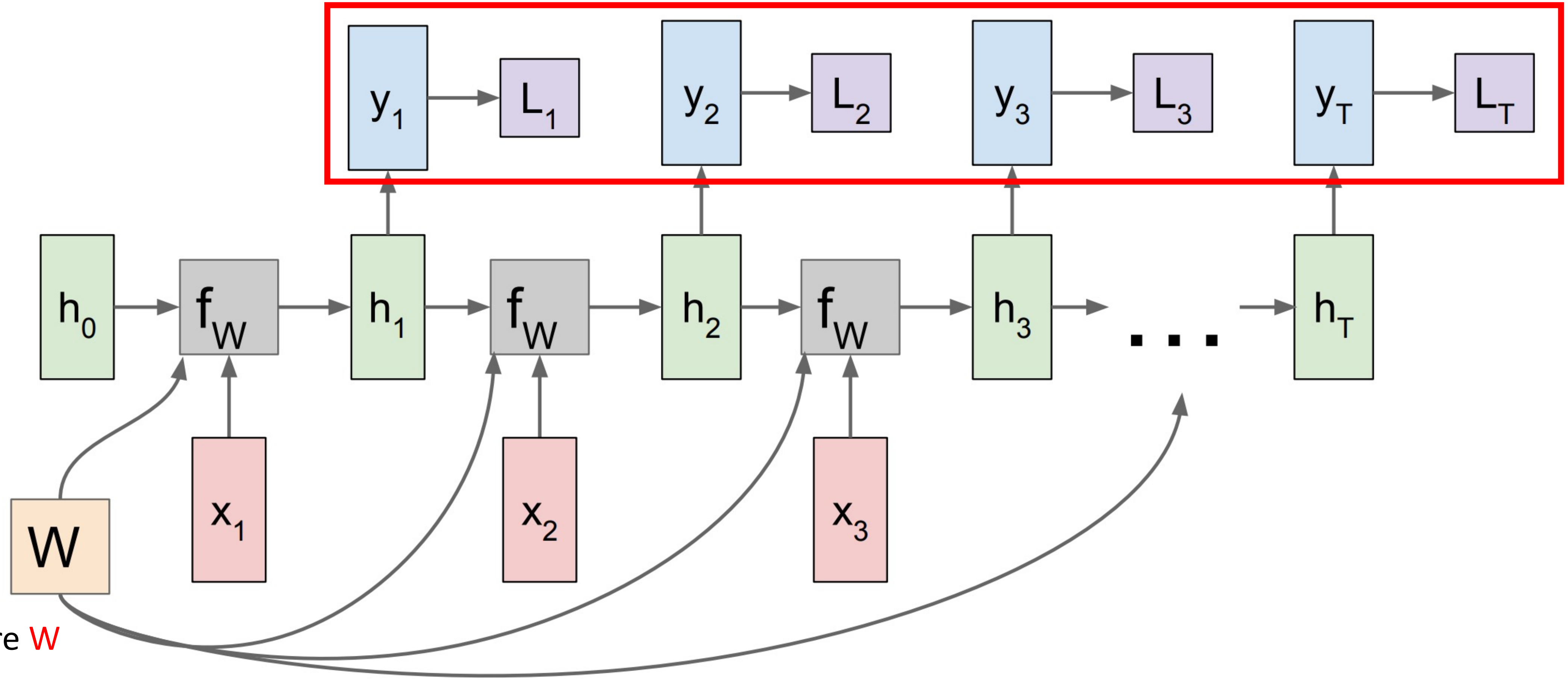
Share W

Recurrent neural networks



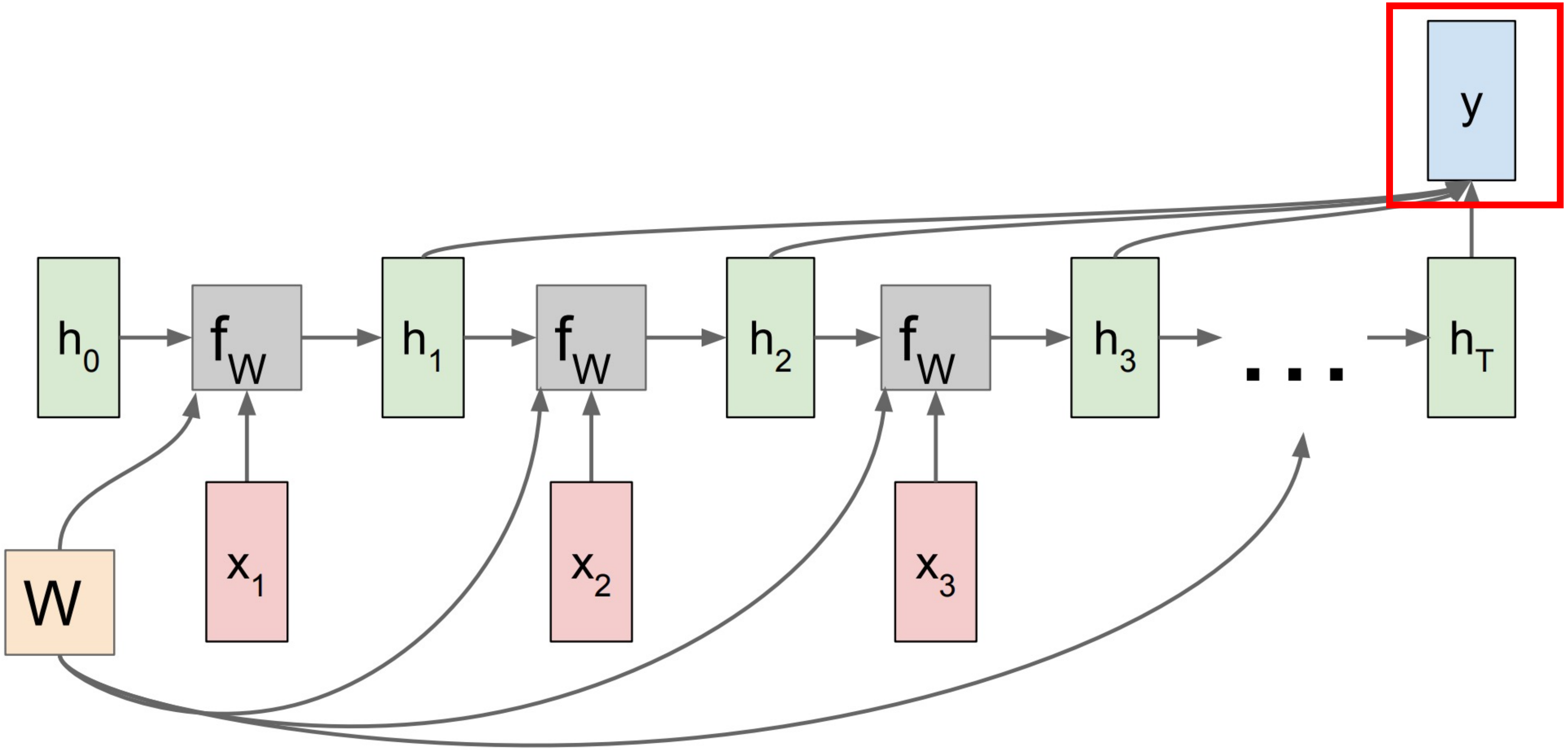
Share W

Recurrent neural networks



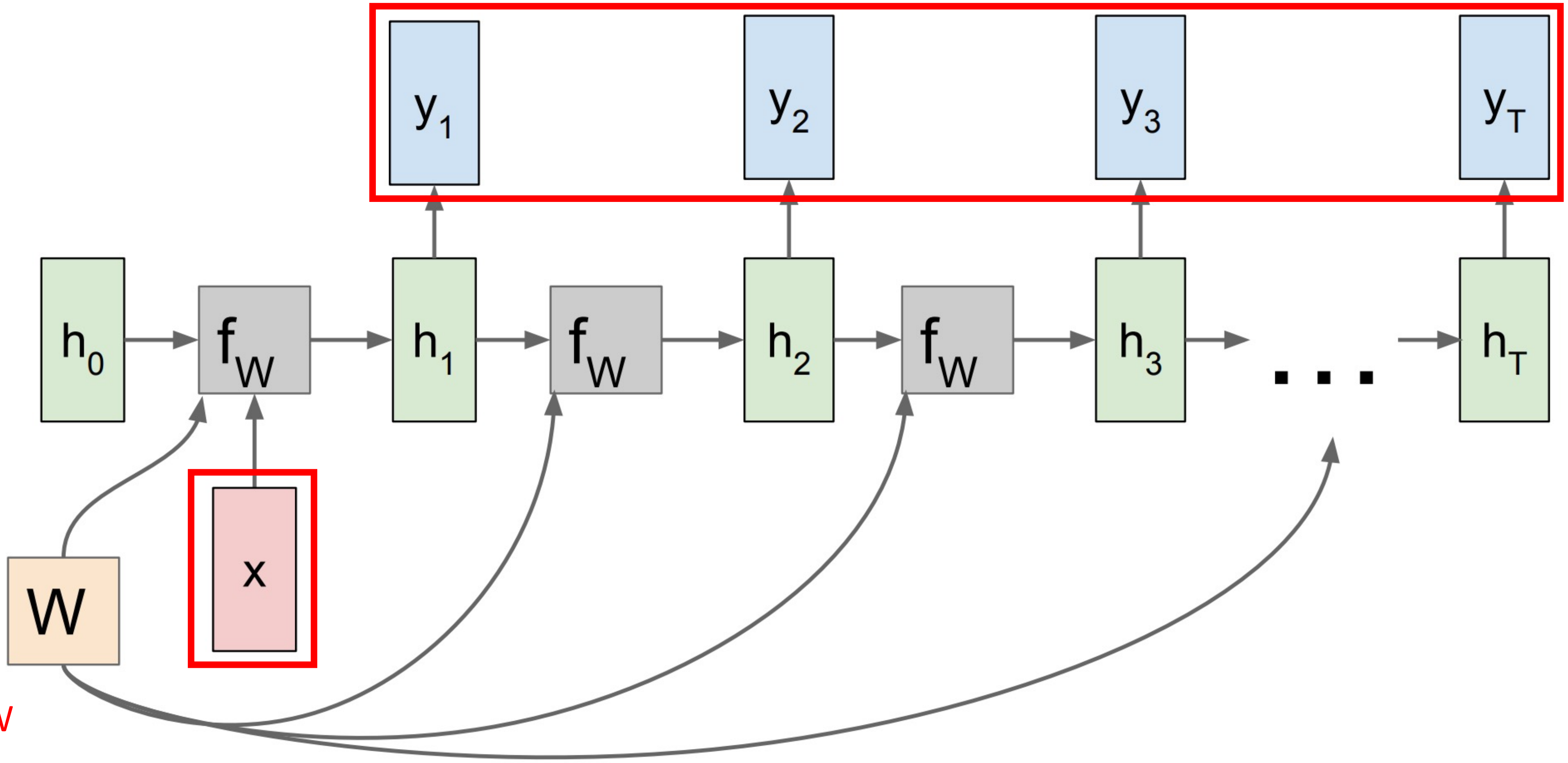
Share W

Recurrent neural networks



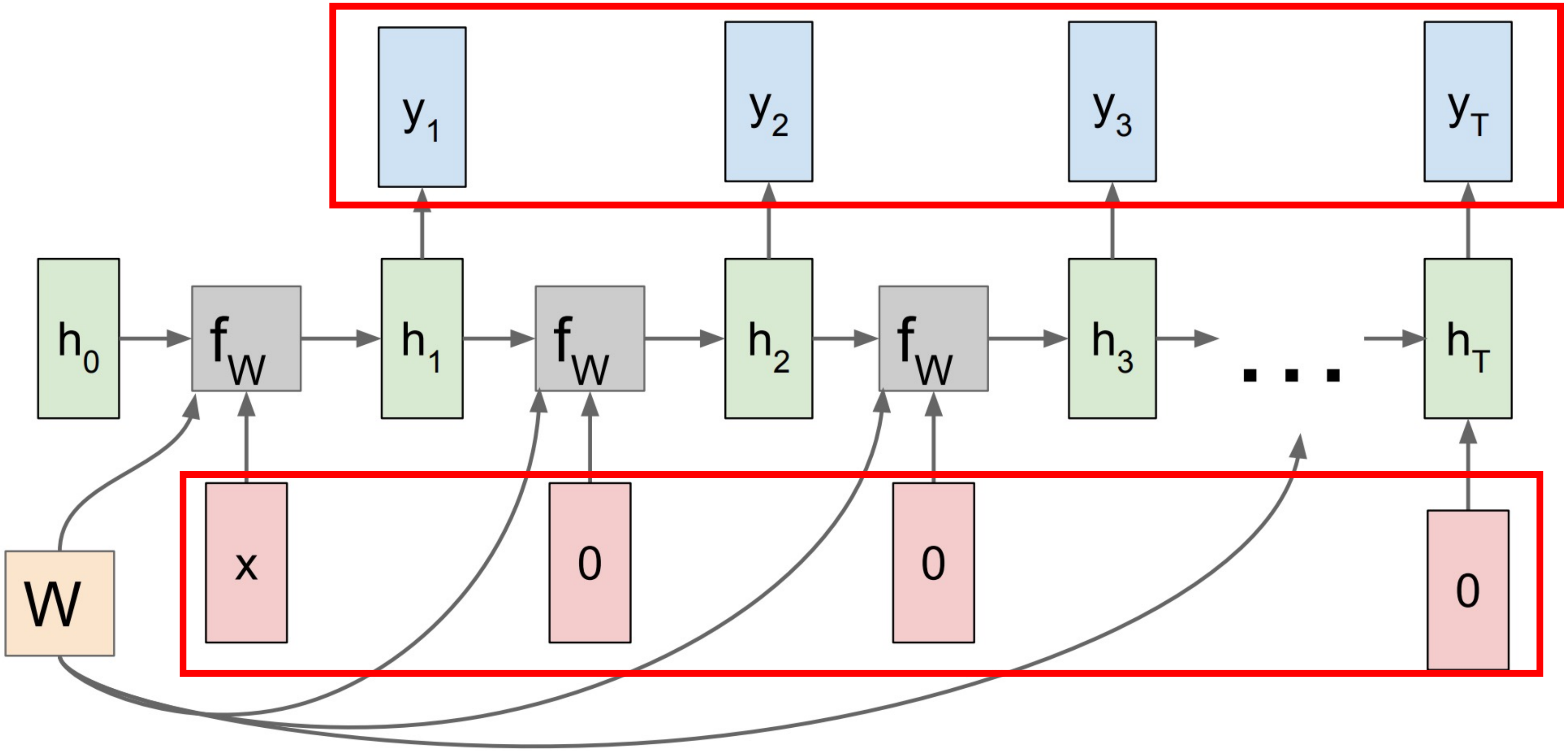
Share W

Recurrent neural networks



Share W

Recurrent neural networks



Share W

RNNs for language model

- Guess the word:
  - h

RNNs for language model

- Guess the word:
    - he

RNNs for language model

- Guess the word:
  - hel

RNNs for language model

- Guess the word:
  - hell

RNNs for language model

- Guess the word:
  - hello

RNNs for language model

- Guess the word:
  - hello
  - net

RNNs for language model

- Guess the word:
  - hello
  - netw

RNNs for language model

- Guess the word:
  - hello
  - netwo

RNNs for language model

- Guess the word:
  - hello
  - network

RNNs for language model

- Guess the word:
    - hello
    - network
    - l

RNNs for language model

- Guess the word:
  - hello
  - network
  - lan

RNNs for language model

- Guess the word:
  - hello
  - network
  - langu

RNNs for language model

- Guess the word:
  - hello
  - network
  - languag

RNNs for language model

- Guess the word:
  - hello
  - network
  - language

RNNs for language model

- Guess the word:
  - hello
  - network
  - language
- Sequence data: predict the next value

RNNs for language model

- Guess the word:
  - hello
  - network
  - language
- Sequence data: predict the next value
  - n
  - n

RNNs for language model

- Guess the word:
    - hello
    - network
    - language
- Sequence data: predict the next value
    - ne
    - ne

RNNs for language model

- Guess the word:
  - hello
  - network
  - language
- Sequence data: predict the next value
  - neu
  - net

RNNs for language model

- Guess the word:
  - hello
  - network
  - language
- Sequence data: predict the next value
  - neur
  - netw

RNNs for language model

- Guess the word:
  - hello
  - network
  - language
- Sequence data: predict the next value
  - neura
  - netwo

RNNs for language model

- Guess the word:
  - hello
  - network
  - language
- Sequence data: predict the next value
  - neural
  - network

RNNs for language model

- Guess the word:
  - hello
  - network
  - language
- Sequence data: predict the next value
  - neural
  - network

RNNs for language model

- Guess the word:
  - hello
  - network
  - language
- Sequence data: predict the next value
  - neural
  - network

RNNs for language model

- Guess the word:
  - hello
  - network
  - language
- Sequence data: predict the next value
  - neural
  - network

Information flow

Character-level language model

- Vocabulary: {a, b, …, z}

Character-level language model

- Vocabulary: {a, b, …, z}
- Given a sequence of character:

Character-level language model

- Vocabulary: {a, b, …, z}
- Given a sequence of character:
  - hellx
  - mornixx
  - languaxx
  - neurxx
  - netwxxx
  - …

Character-level language model

- Vocabulary: {a, b, …, z}
- Given a sequence of character:
  - hellx → hello
  - mornixx → morning
  - languaxx → language
  - neurxx → neural
  - netwxxx → network
  - …

Character-level language model

- Vocabulary: {h, e, l, o}

Character-level language model

- Vocabulary: {h, e, l, o}

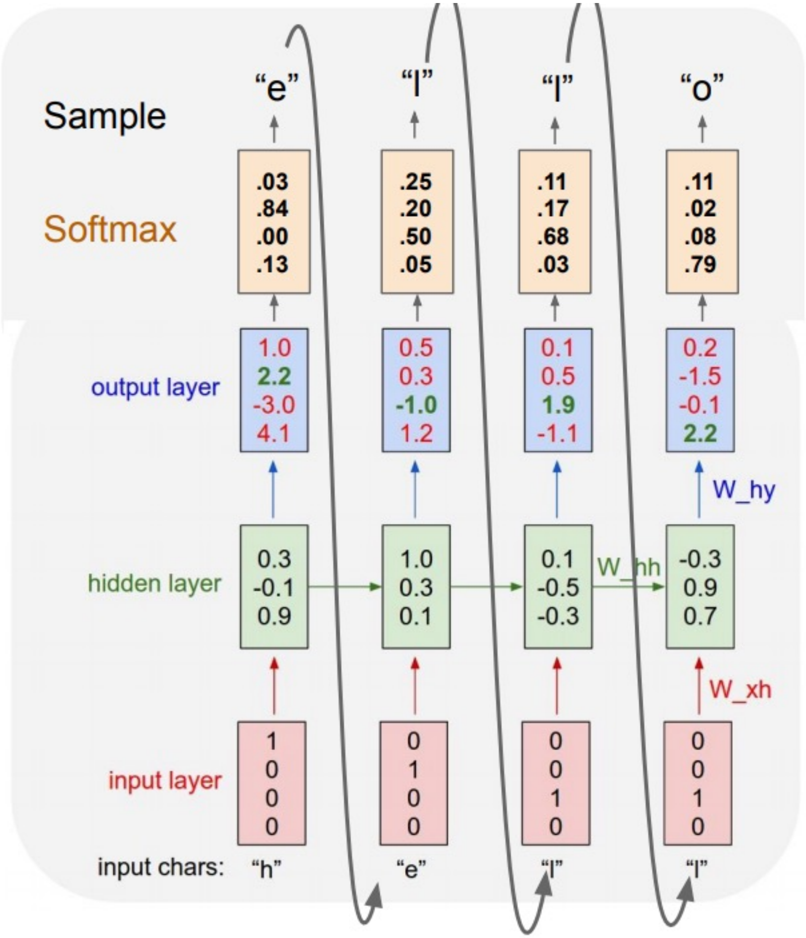
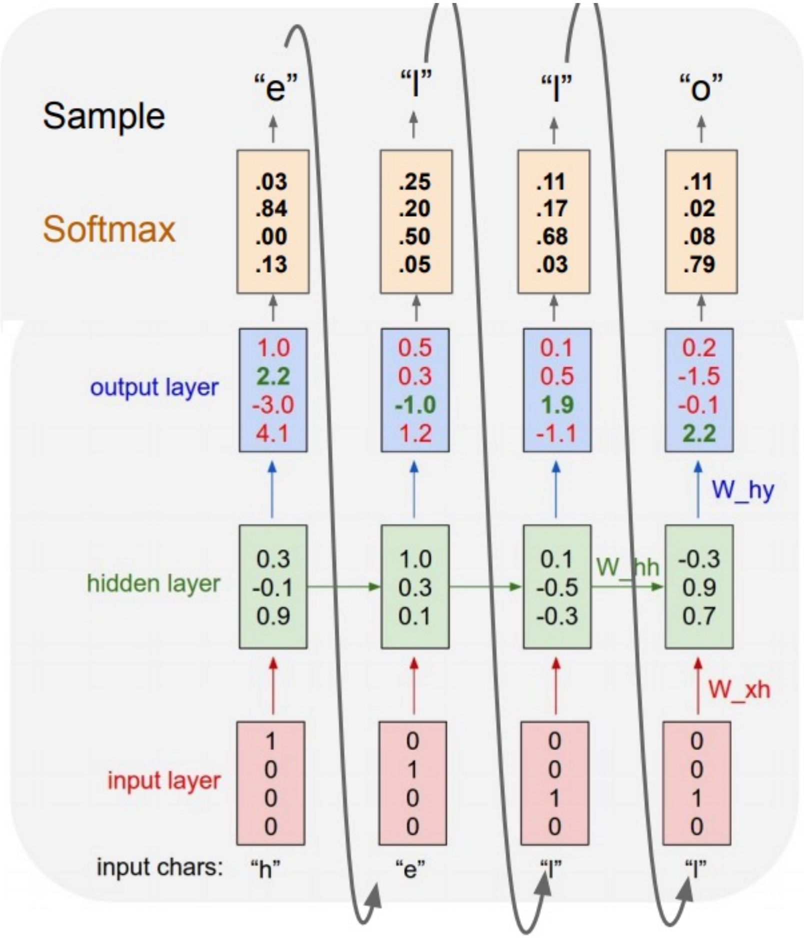


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}



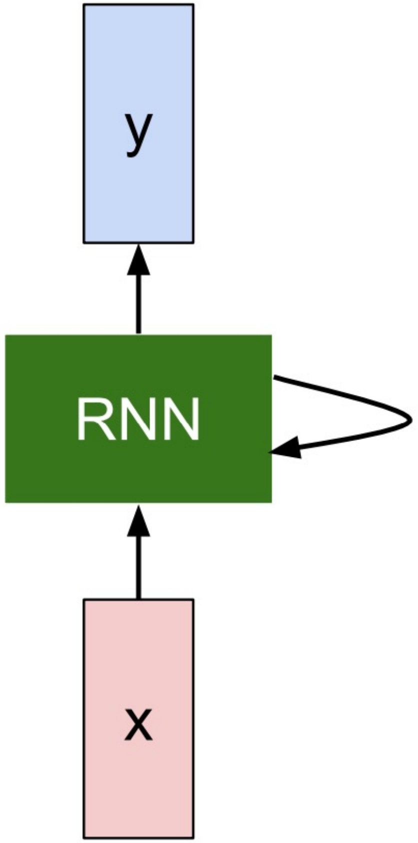
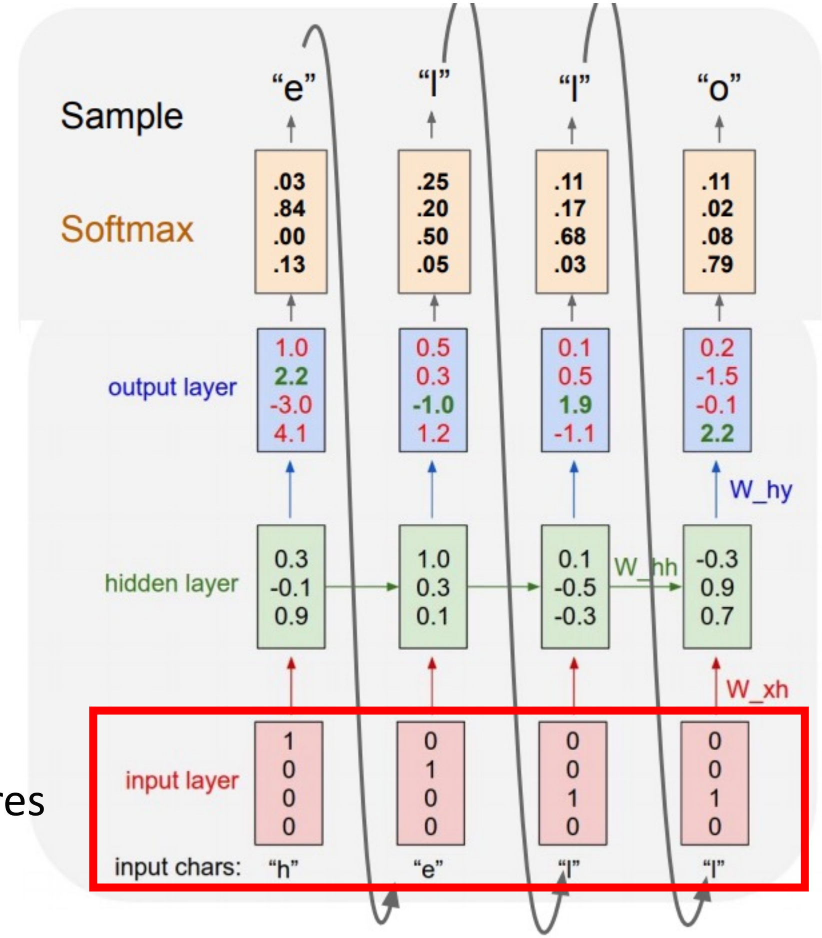


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}



character features

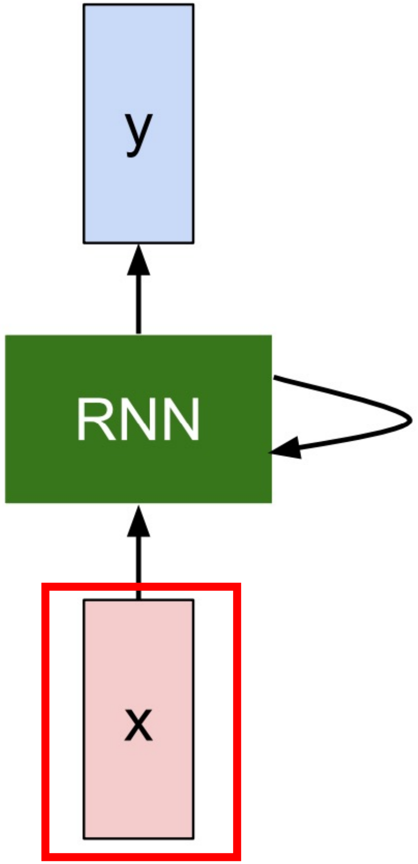
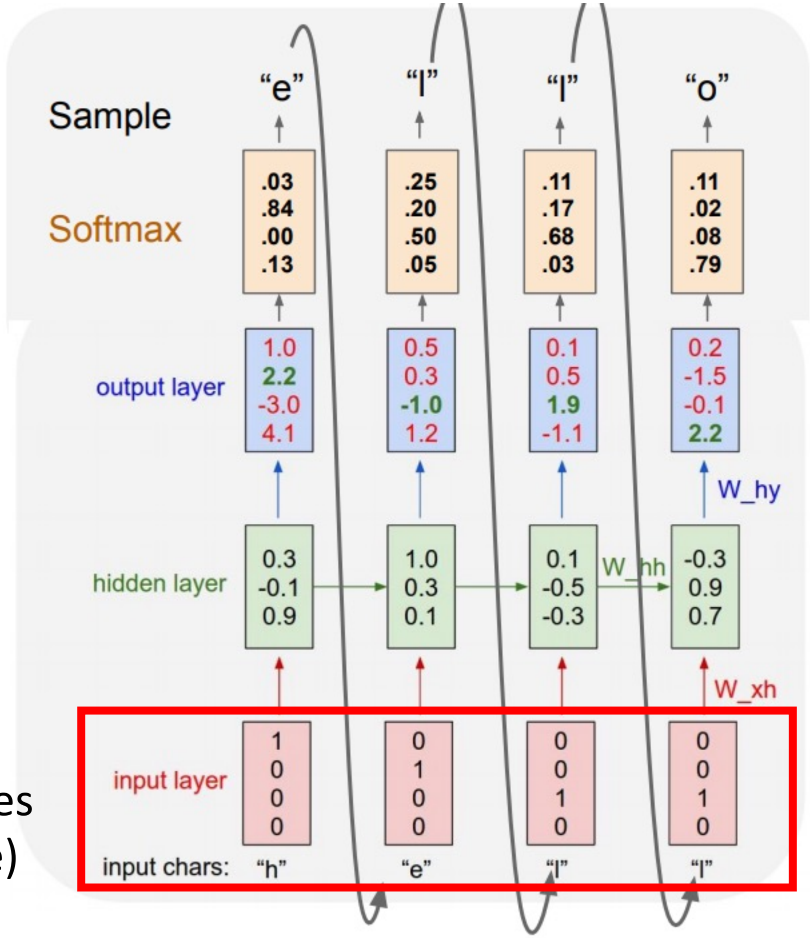


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}



character features
(one-hot encode)

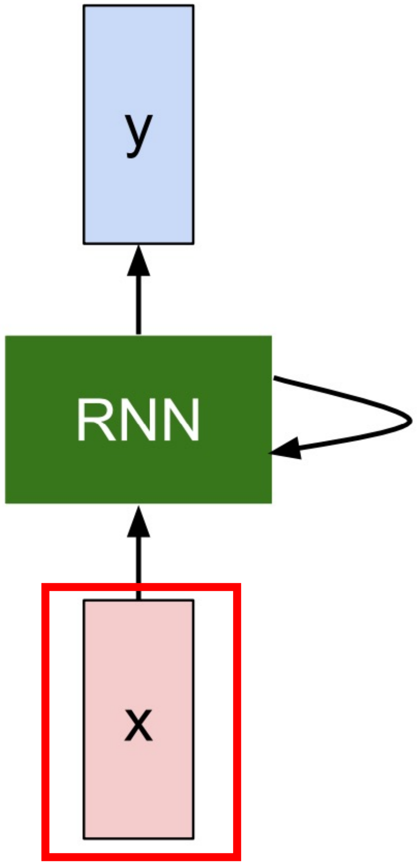
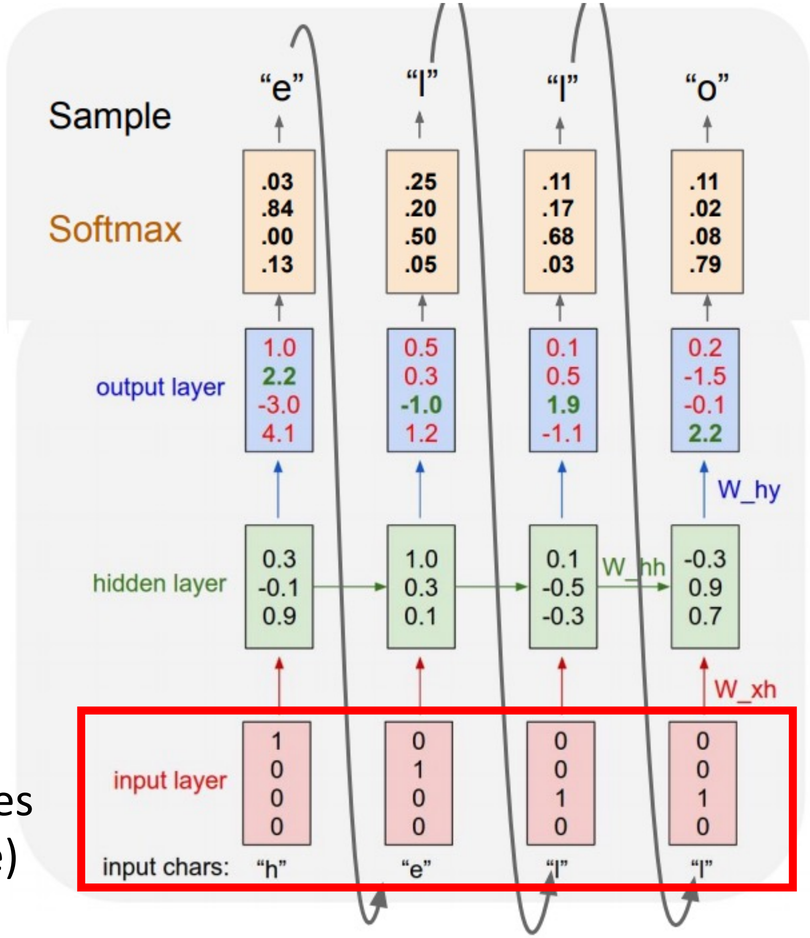


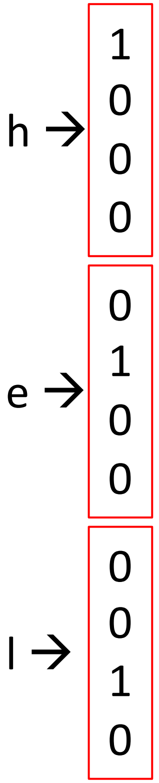
Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}



character features
(one-hot encode)



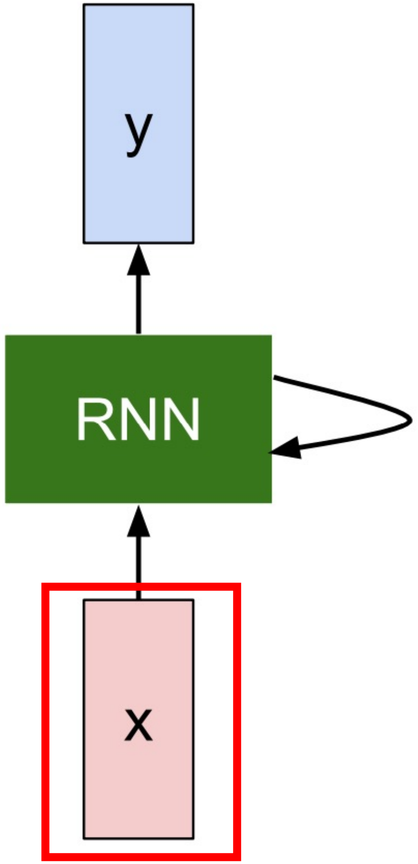
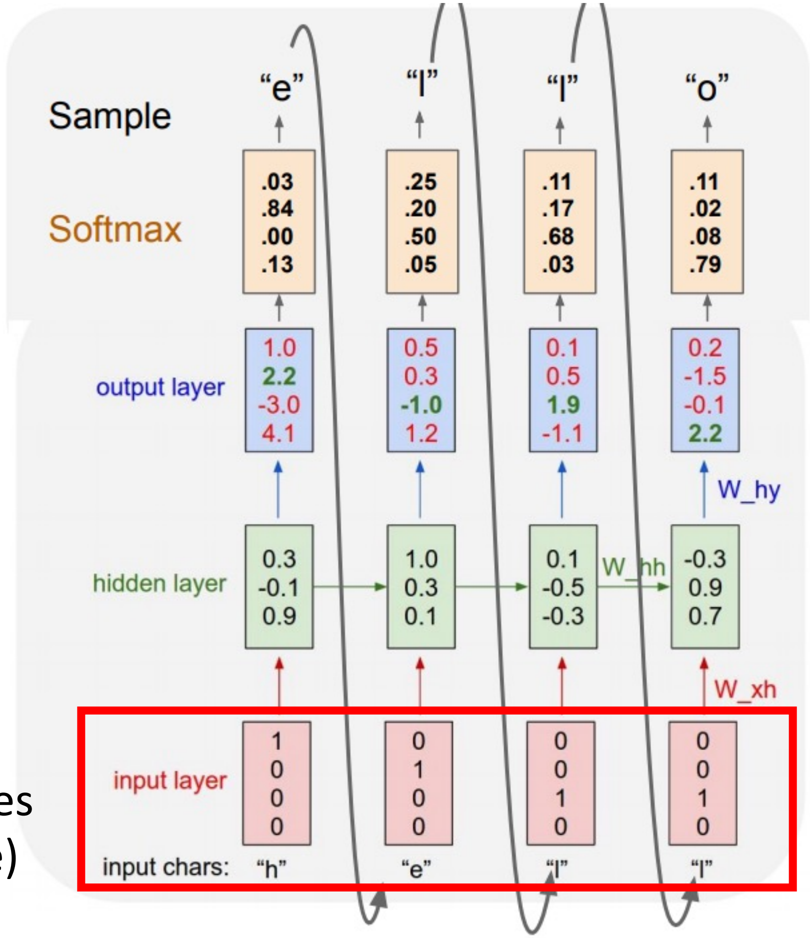


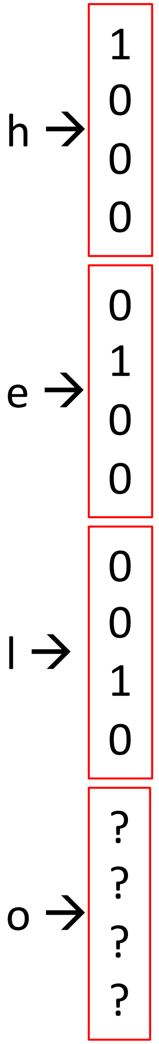
Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}



character features
(one-hot encode)



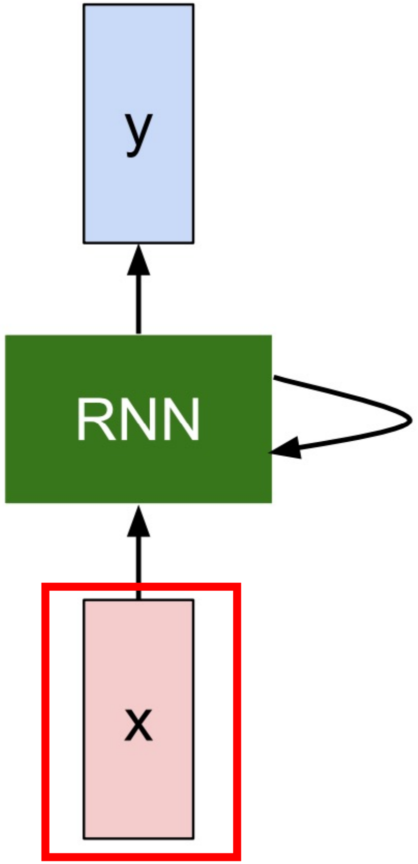
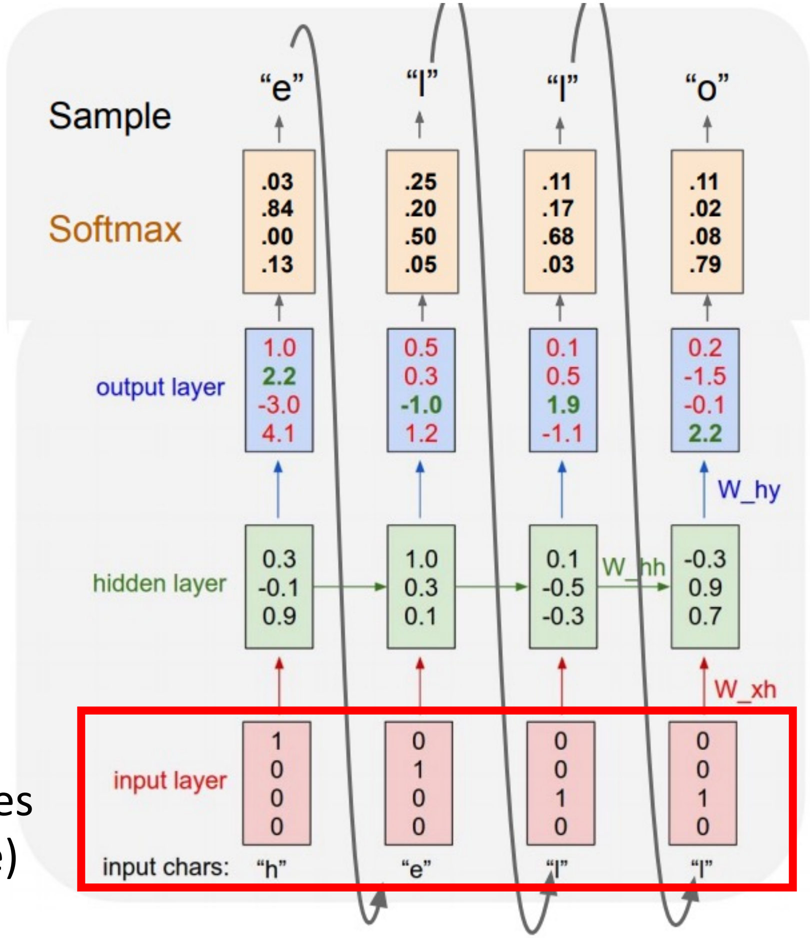


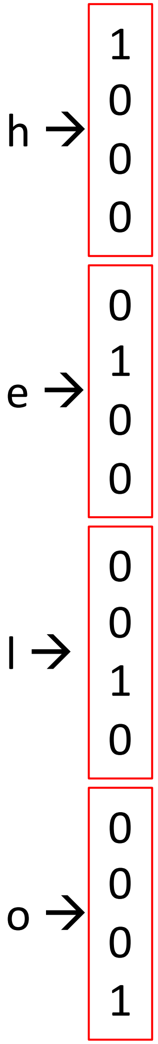
Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}



character features
(one-hot encode)



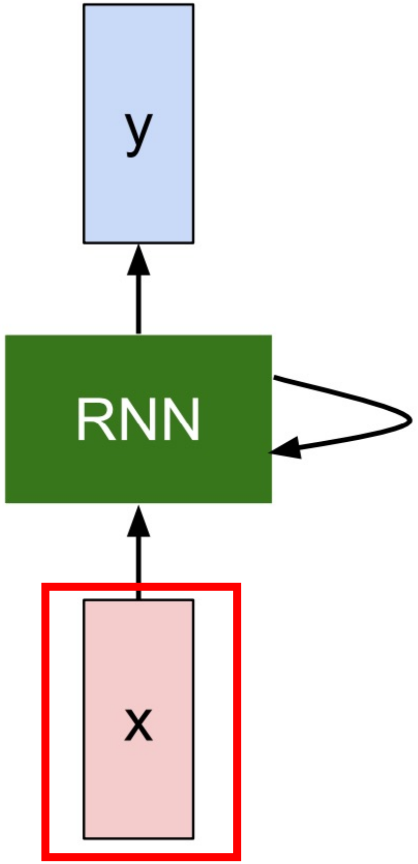
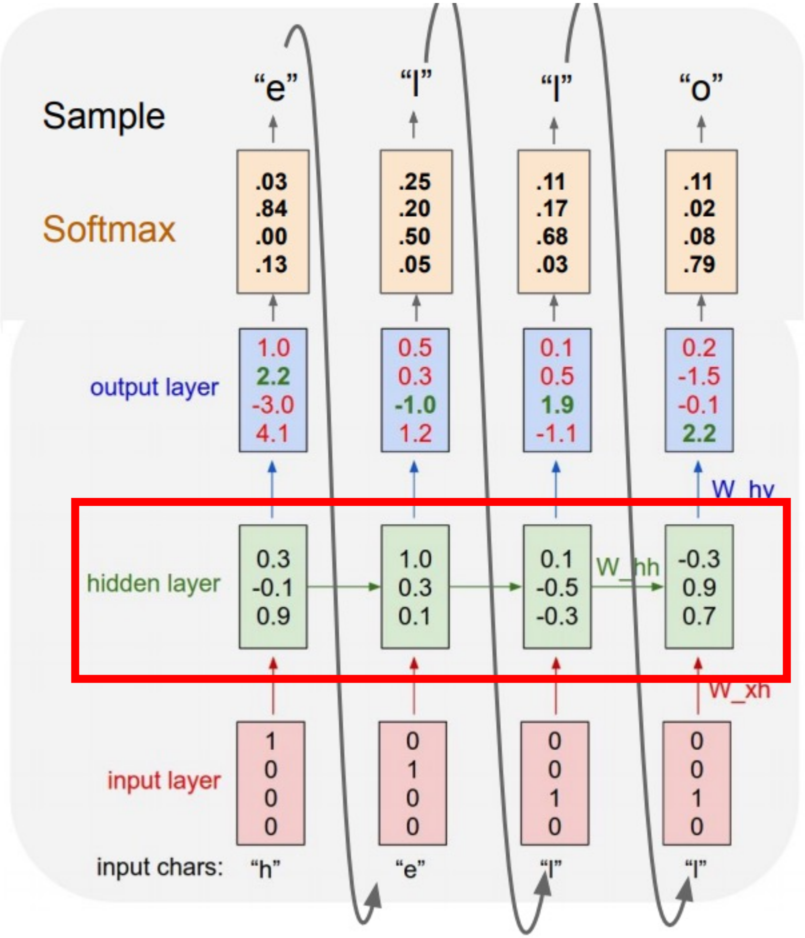


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}



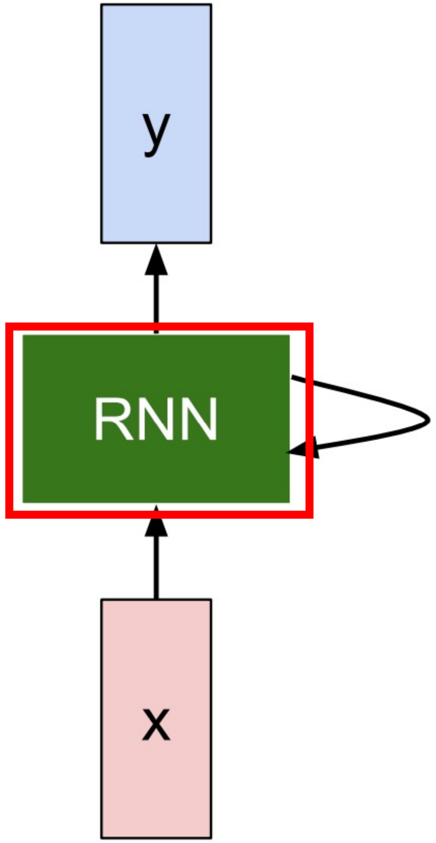
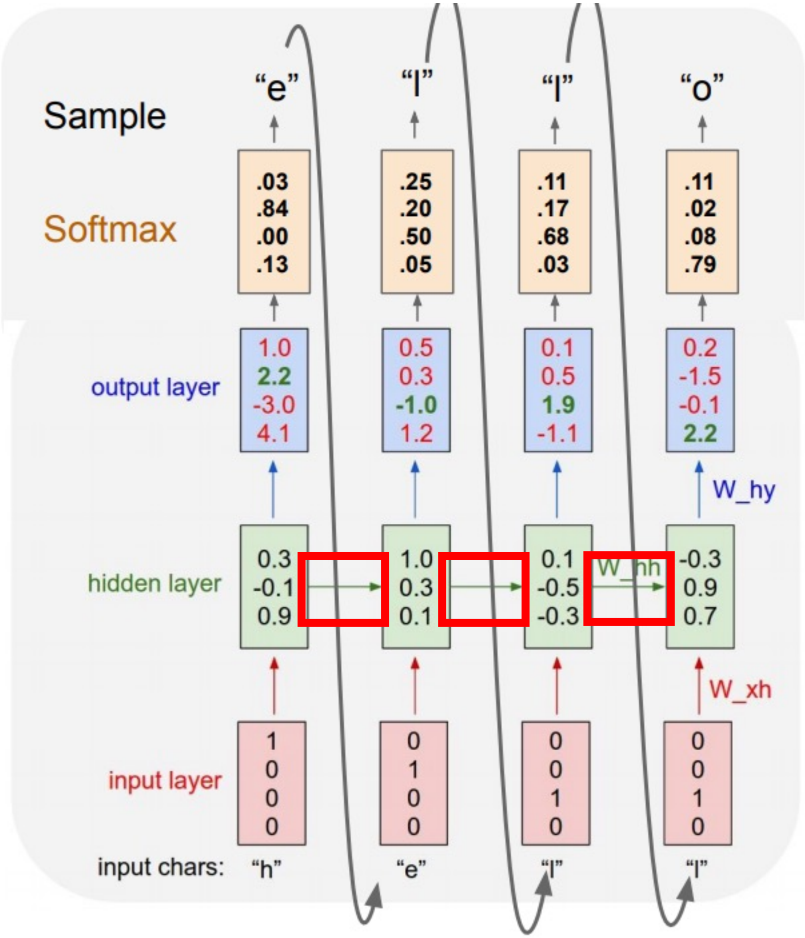


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}



$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t)$$

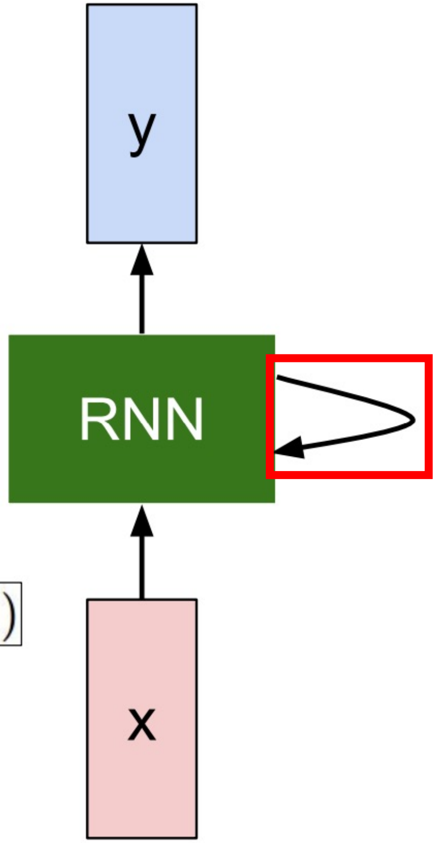
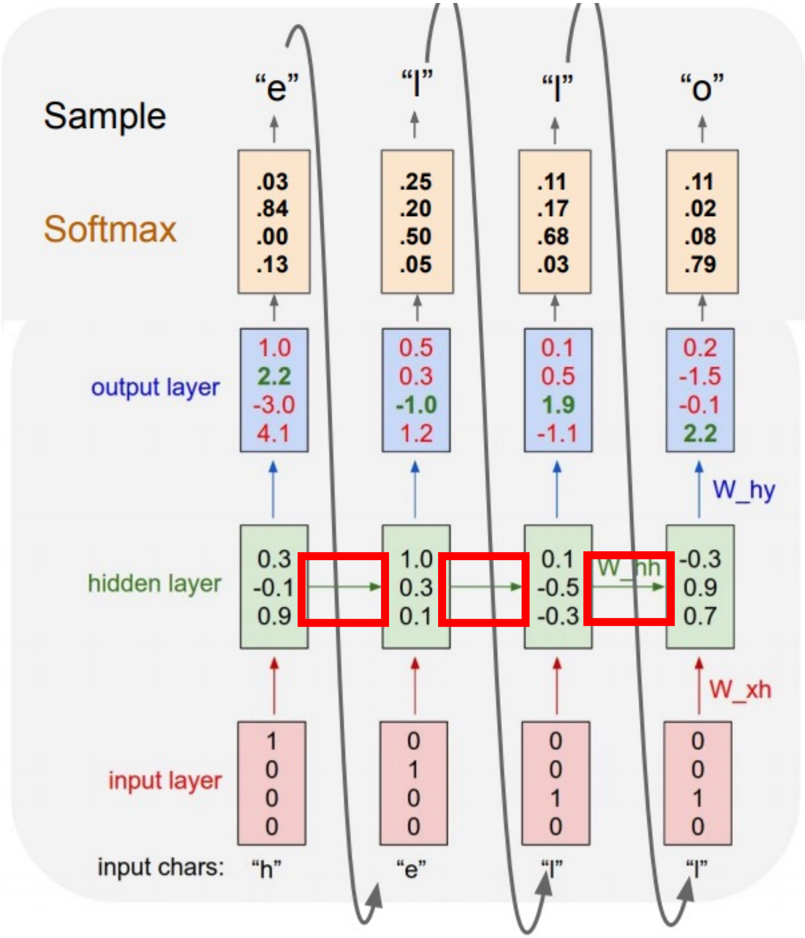


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}



$$h_t = \tanh(\boxed{W_{hh} h_{t-1}} + W_{xh} x_t)$$

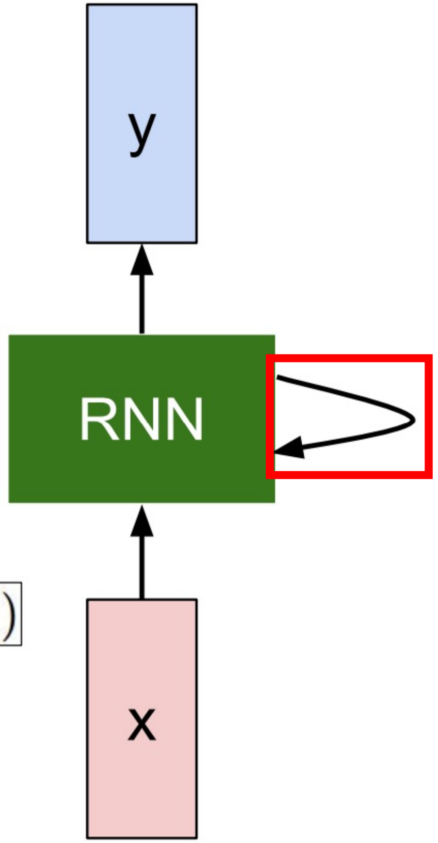
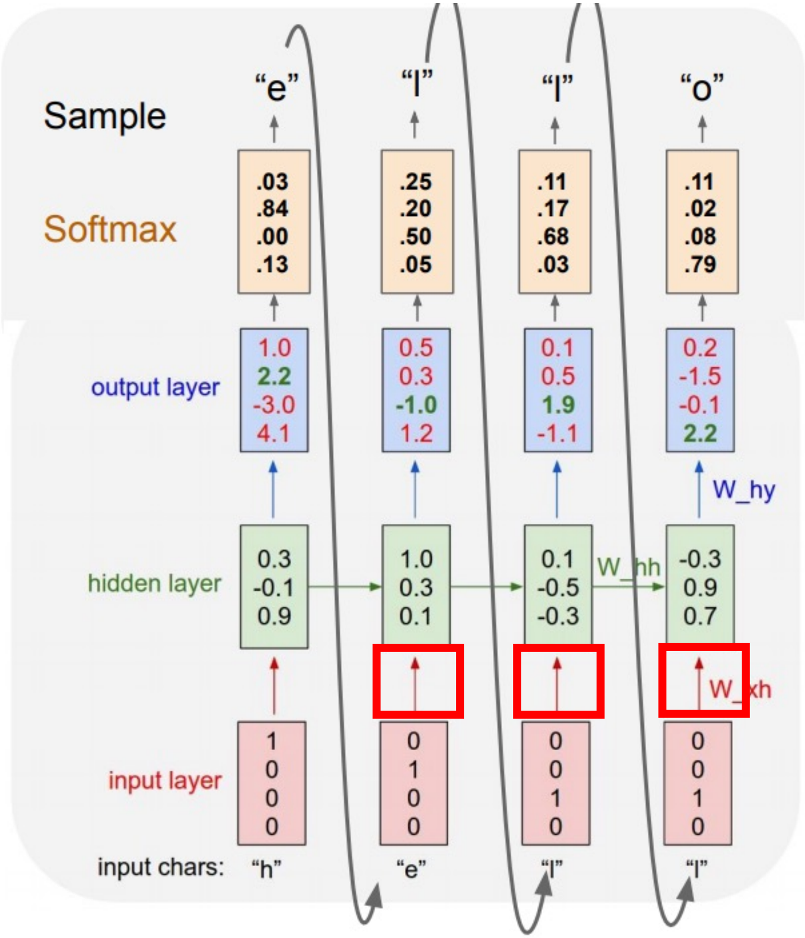


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}



$$h_t = \tanh(W_{hh}h_{t-1} + \boxed{W_{xh}x_t})$$

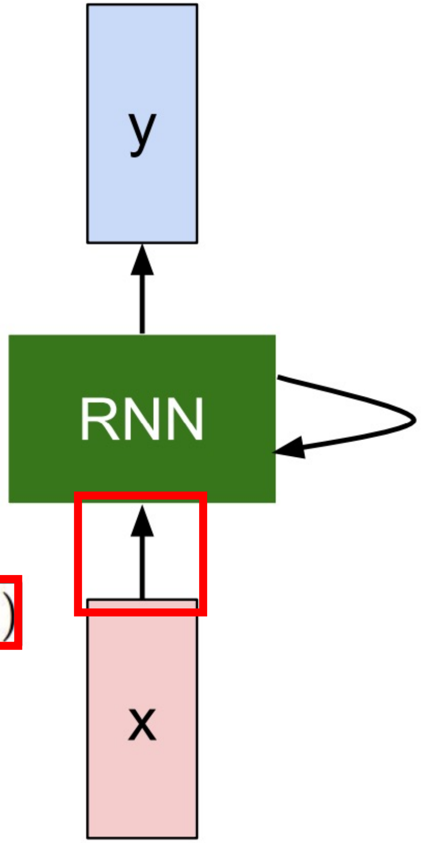
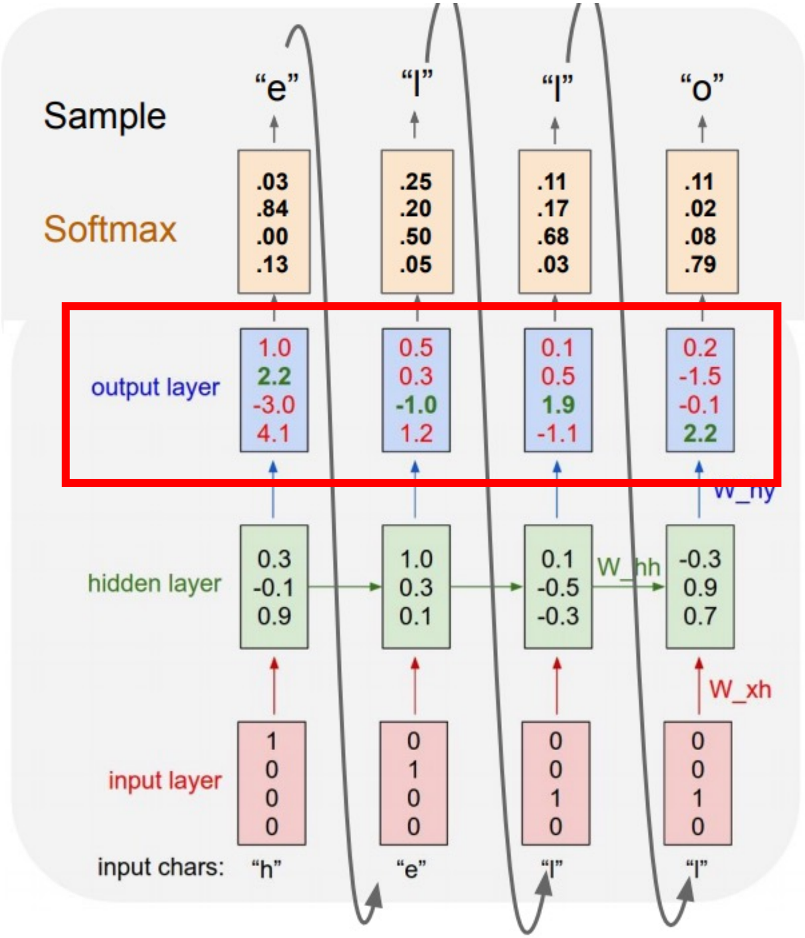


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}



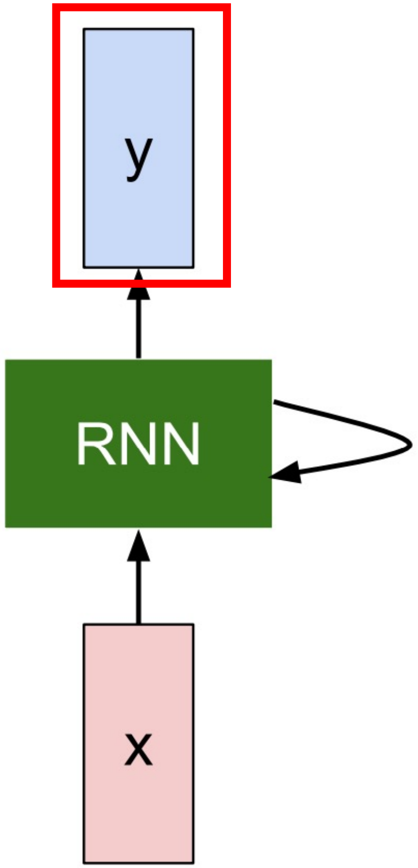
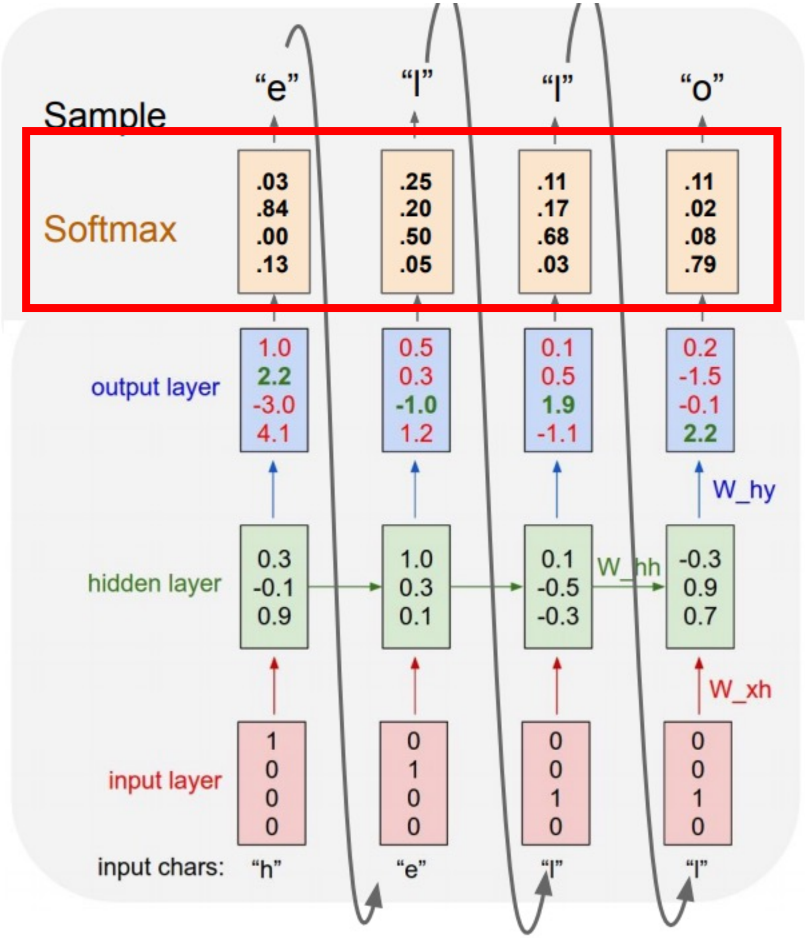


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}



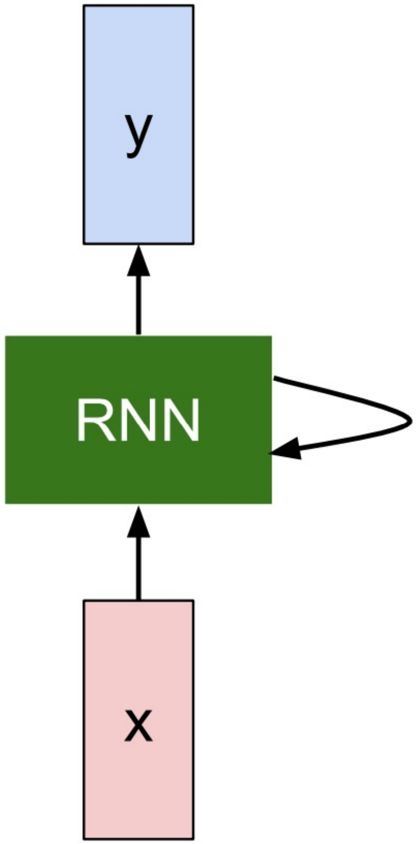
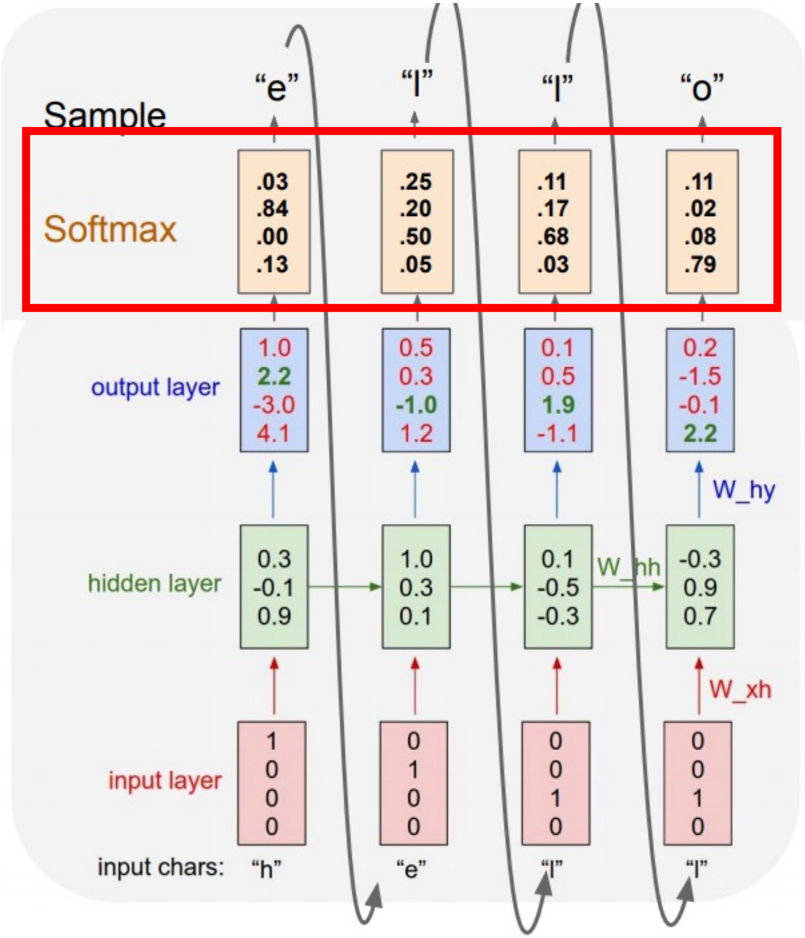


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}

probability



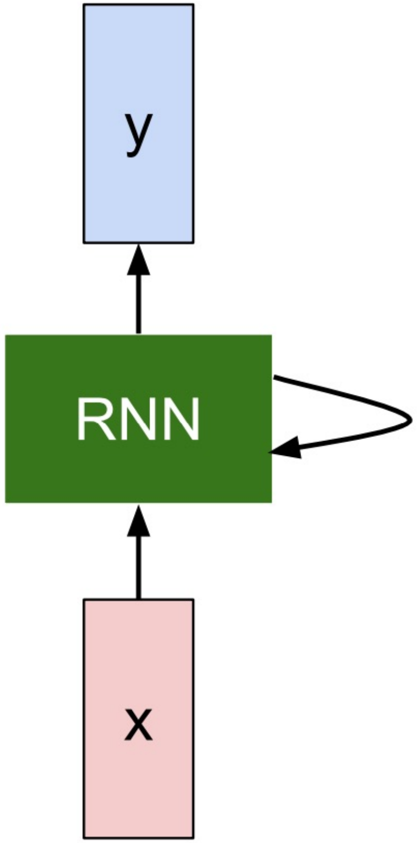
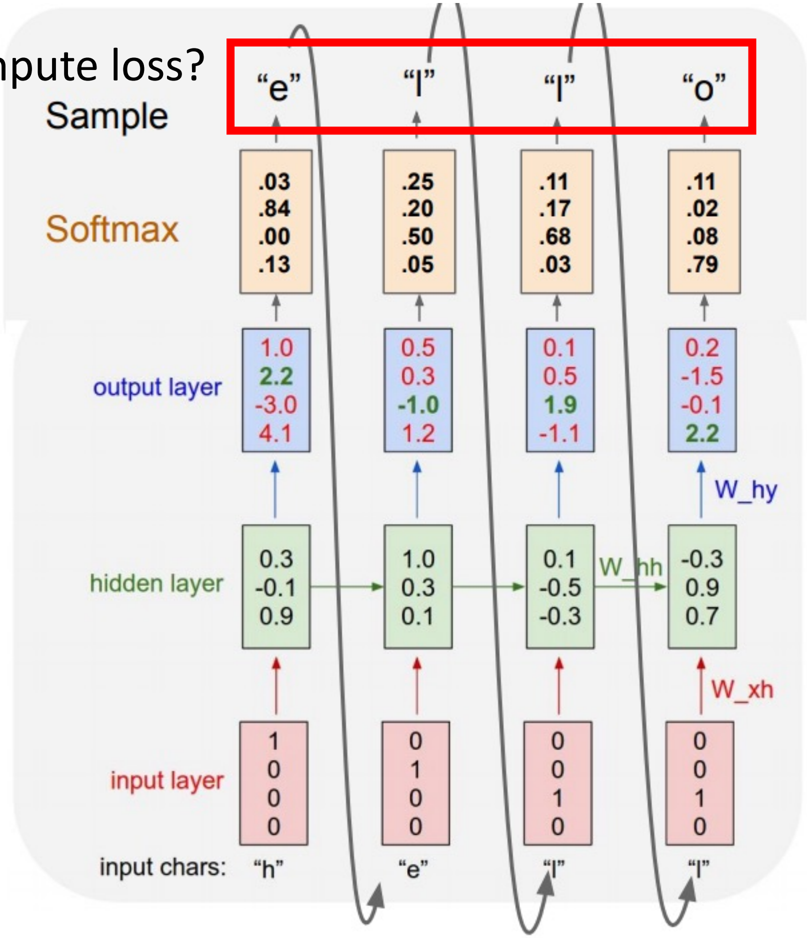


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}

Q: How to compute loss?



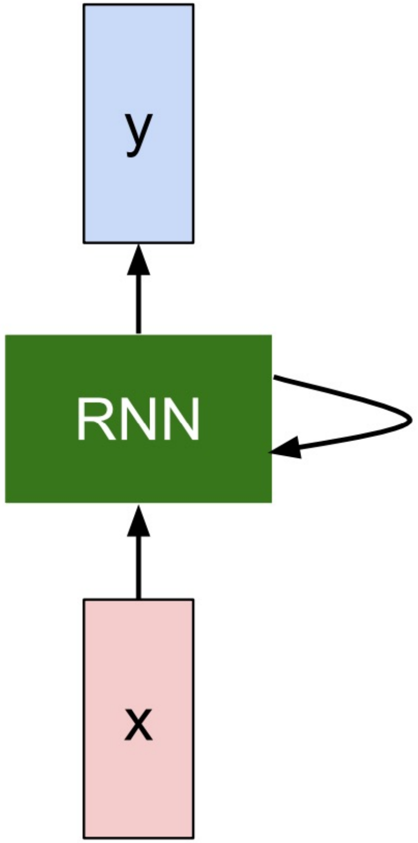
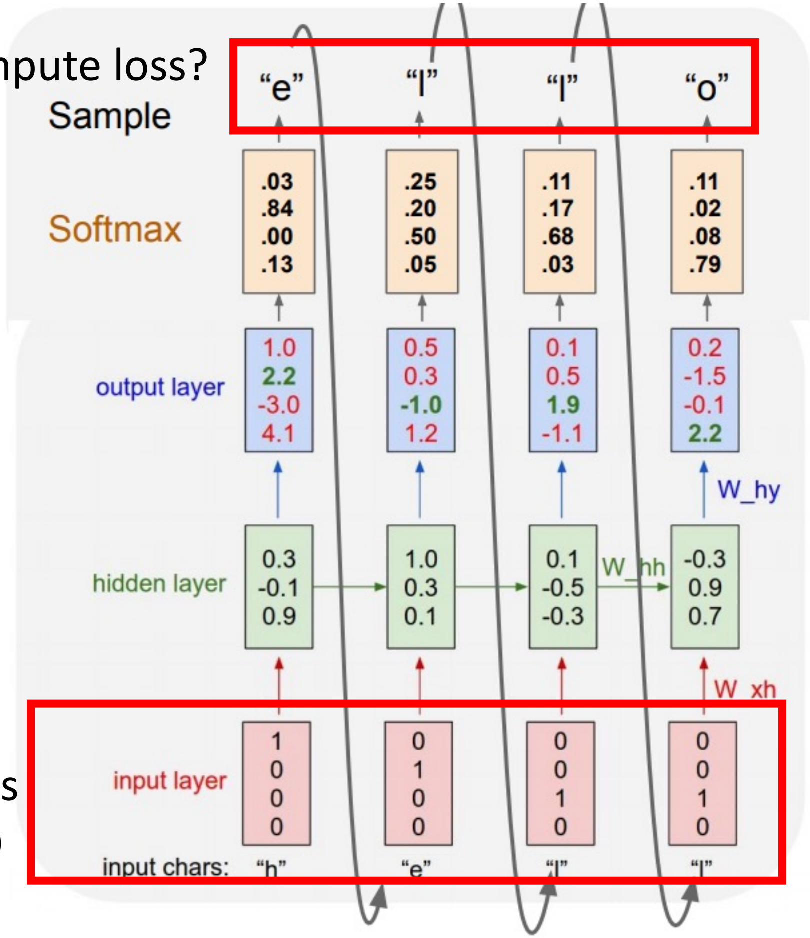


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}

Q: How to compute loss?



character features (one-hot encode)

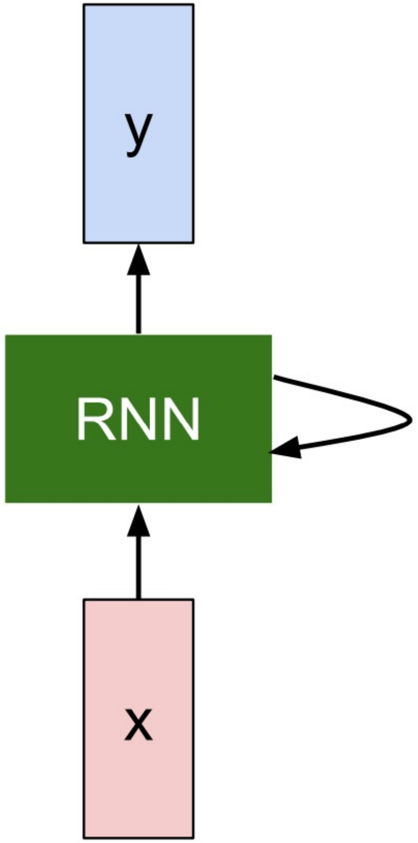
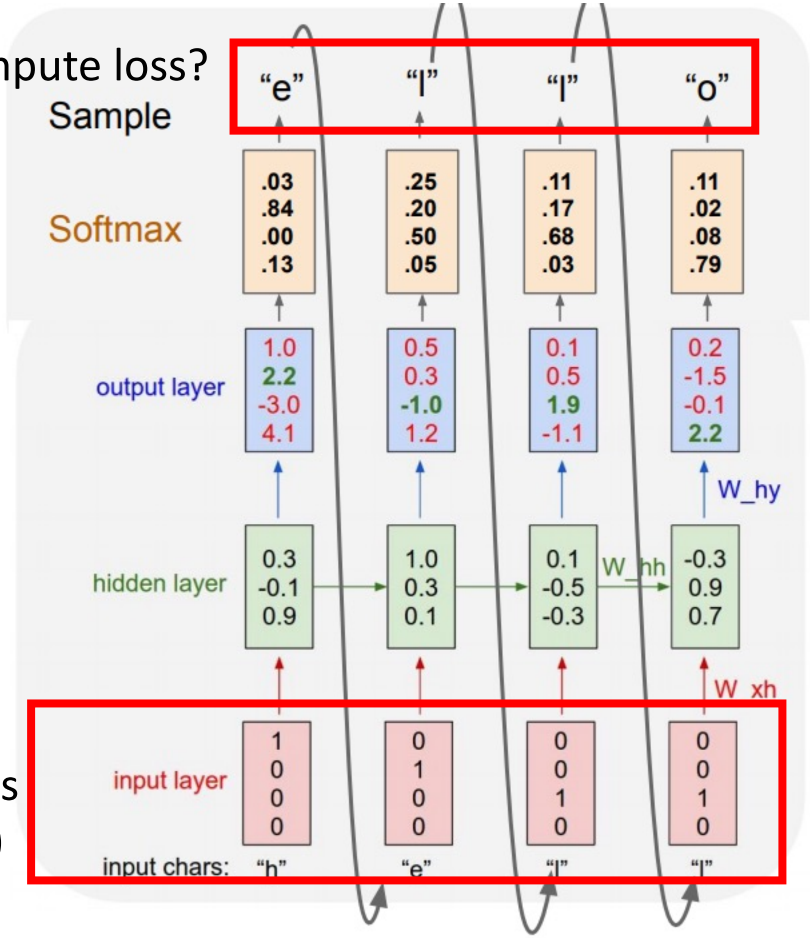


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

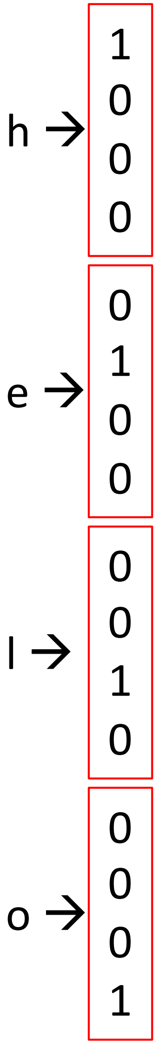
Character-level language model

- Vocabulary: {h, e, l, o}

Q: How to compute loss?



character features
(one-hot encode)



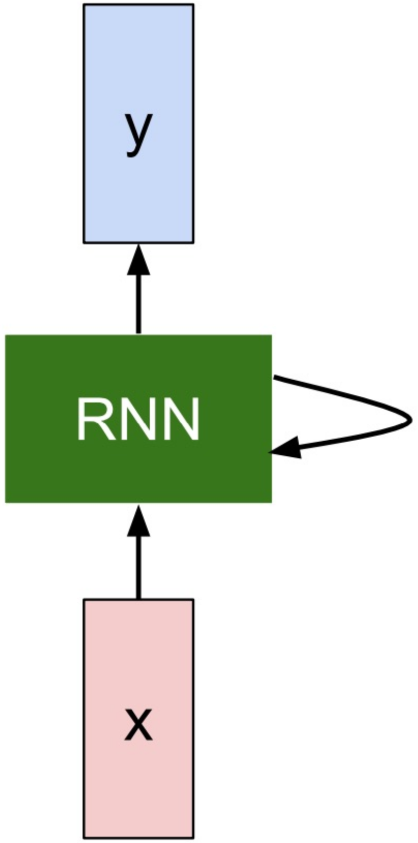
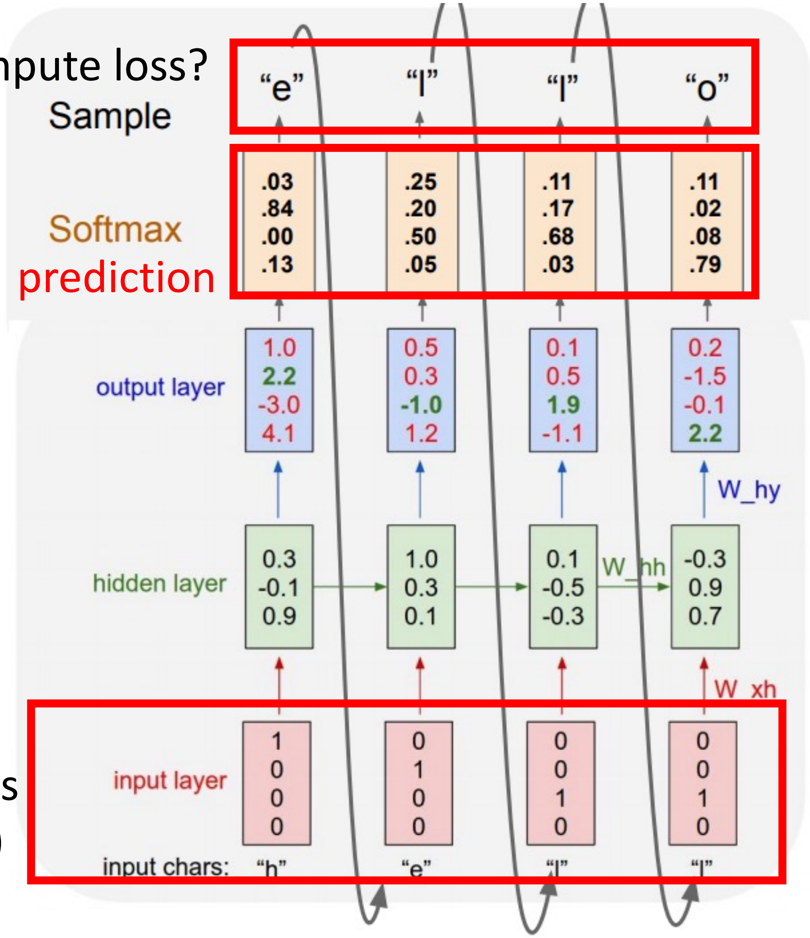


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}

Q: How to compute loss?



character features
(one-hot encode)

$$h \rightarrow \begin{bmatrix}1\\0\\0\\0\end{bmatrix} \quad e \rightarrow \begin{bmatrix}0\\1\\0\\0\end{bmatrix} \quad l \rightarrow \begin{bmatrix}0\\0\\1\\0\end{bmatrix} \quad o \rightarrow \begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

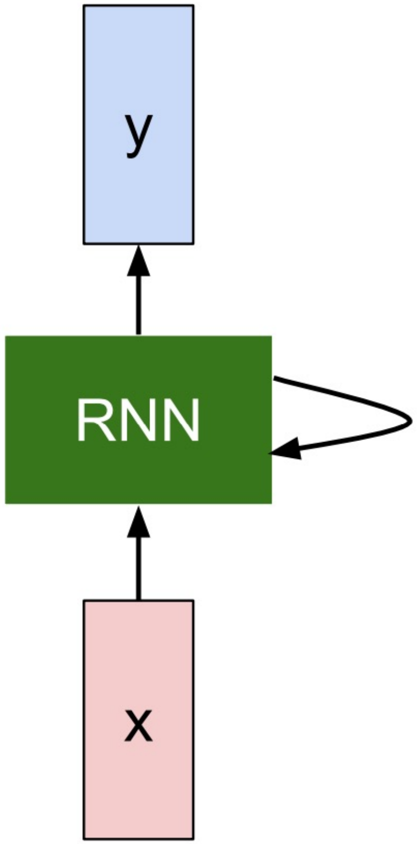


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Character-level language model

- Vocabulary: {h, e, l, o}

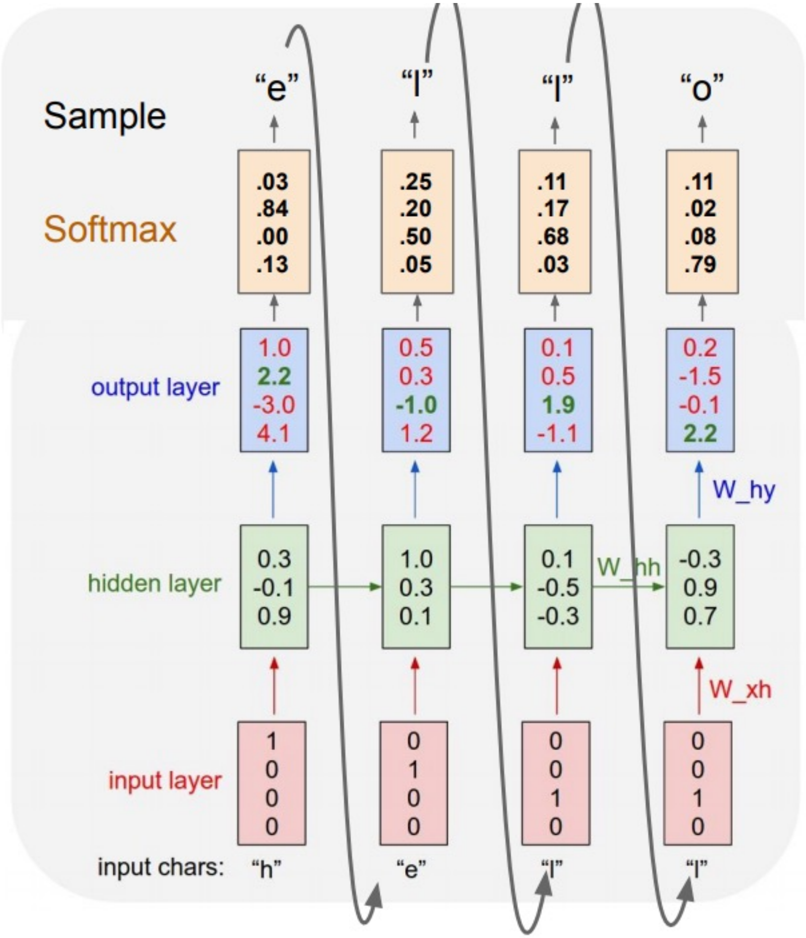


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Word-level language model

- Vocabulary: {h, e, l, o} → {ant, and, …, network, …, zoo}

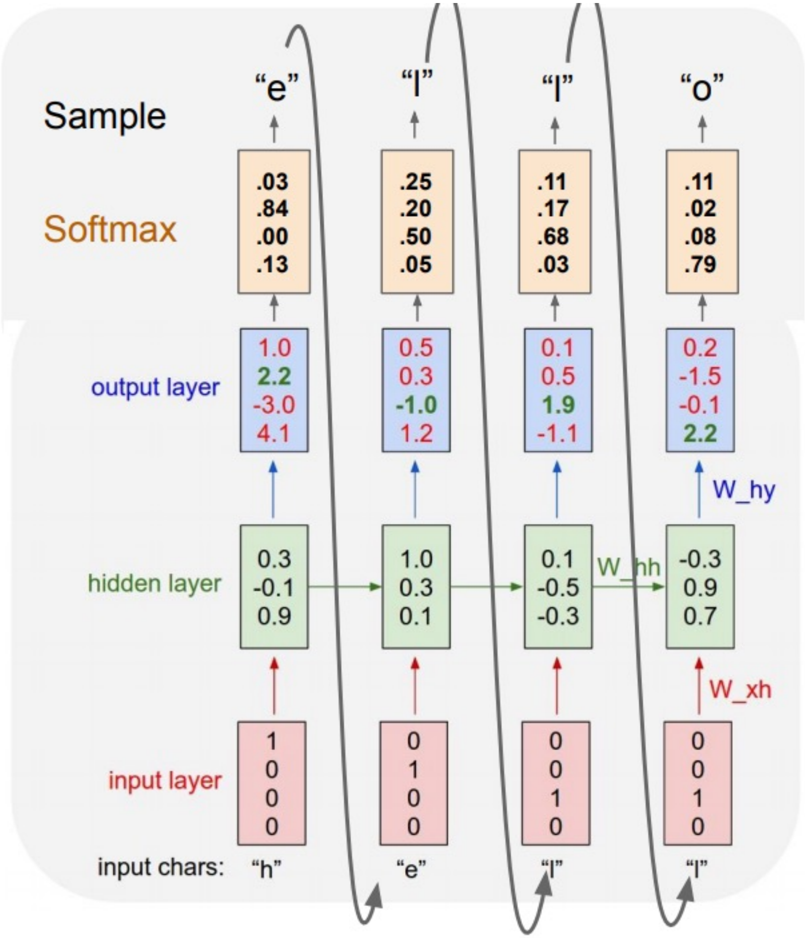
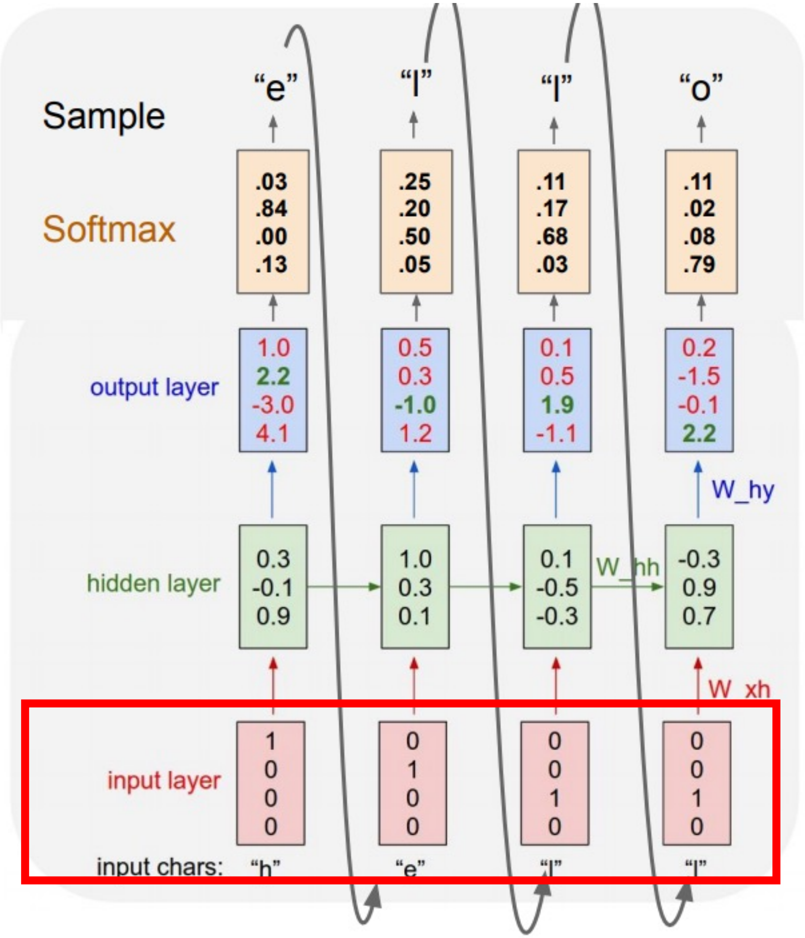


Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Word-level language model

- Vocabulary: {h, e, l, o} → {ant, and, …, network, …, zoo}

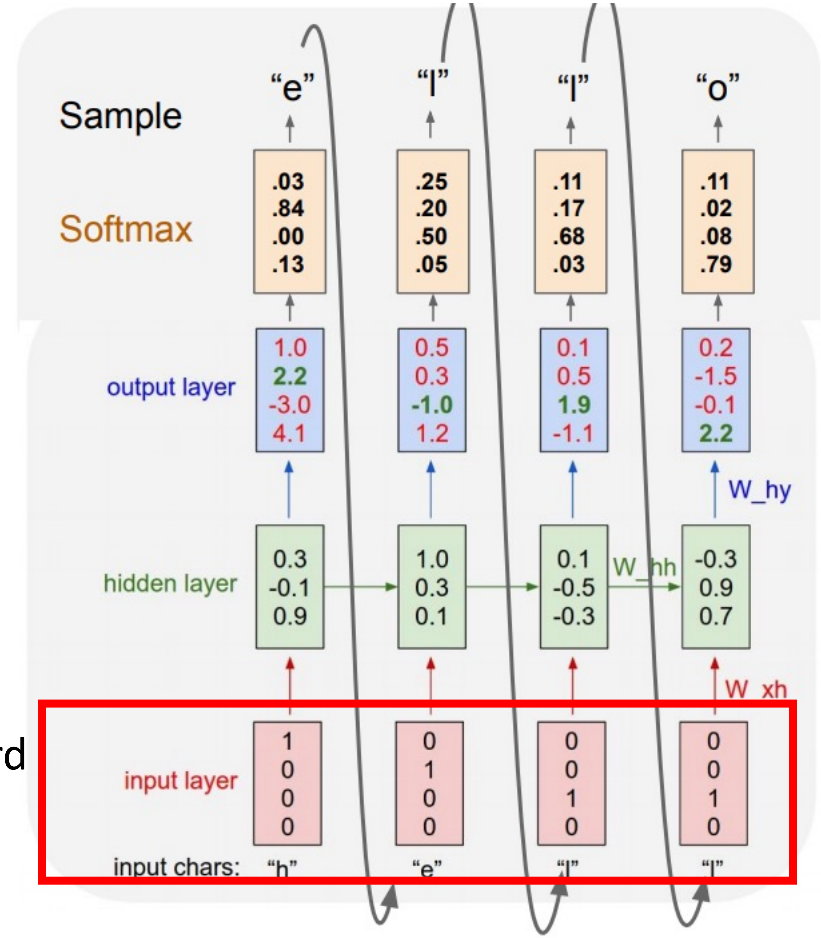


Character→Word

Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Word-level language model

- Vocabulary: {h, e, l, o} Change to → {ant, and, …, network, …, zoo}



Change to
Character → Word

Image from http://cs231n.stanford.edu/slides/2020/lecture_10.pdf

Image captioning

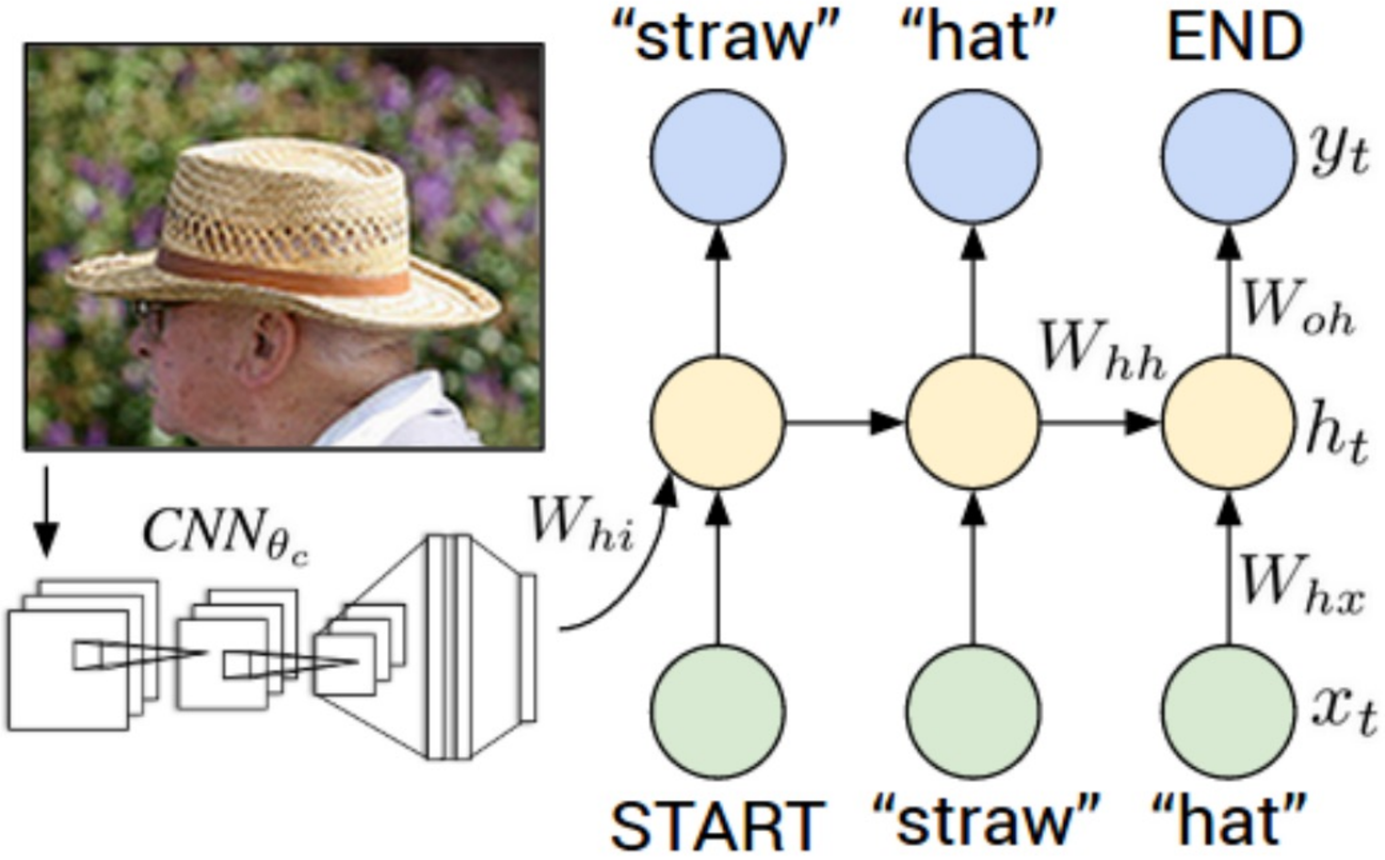


Figure from Karpathy, Andrej, and Li Fei-Fei. "Deep visual-semantic alignments for generating image descriptions." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3128-3137. 2015.

Image captioning

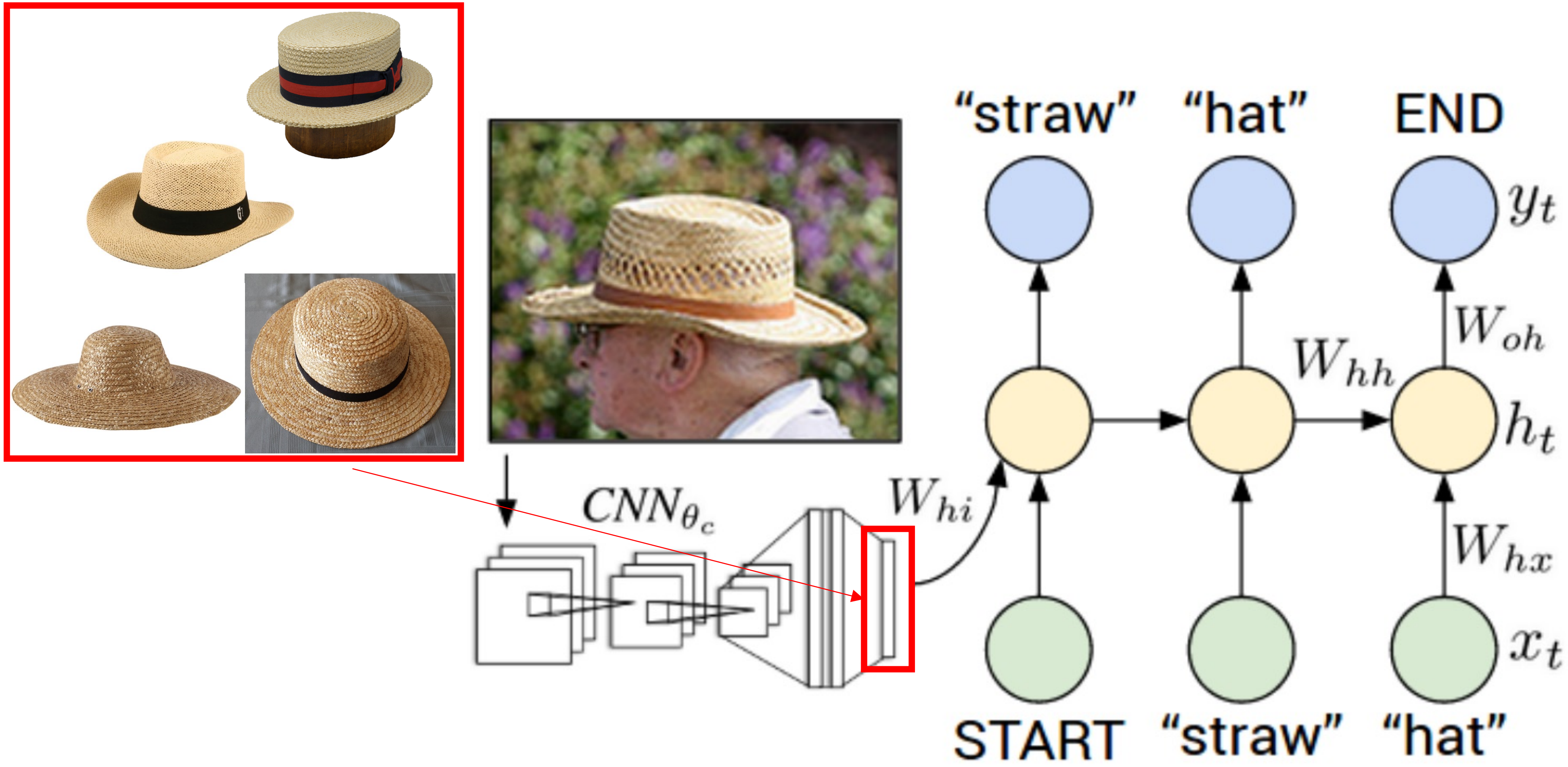


Figure from Karpathy, Andrej, and Li Fei-Fei. "Deep visual-semantic alignments for generating image descriptions." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3128-3137. 2015.

Image captioning

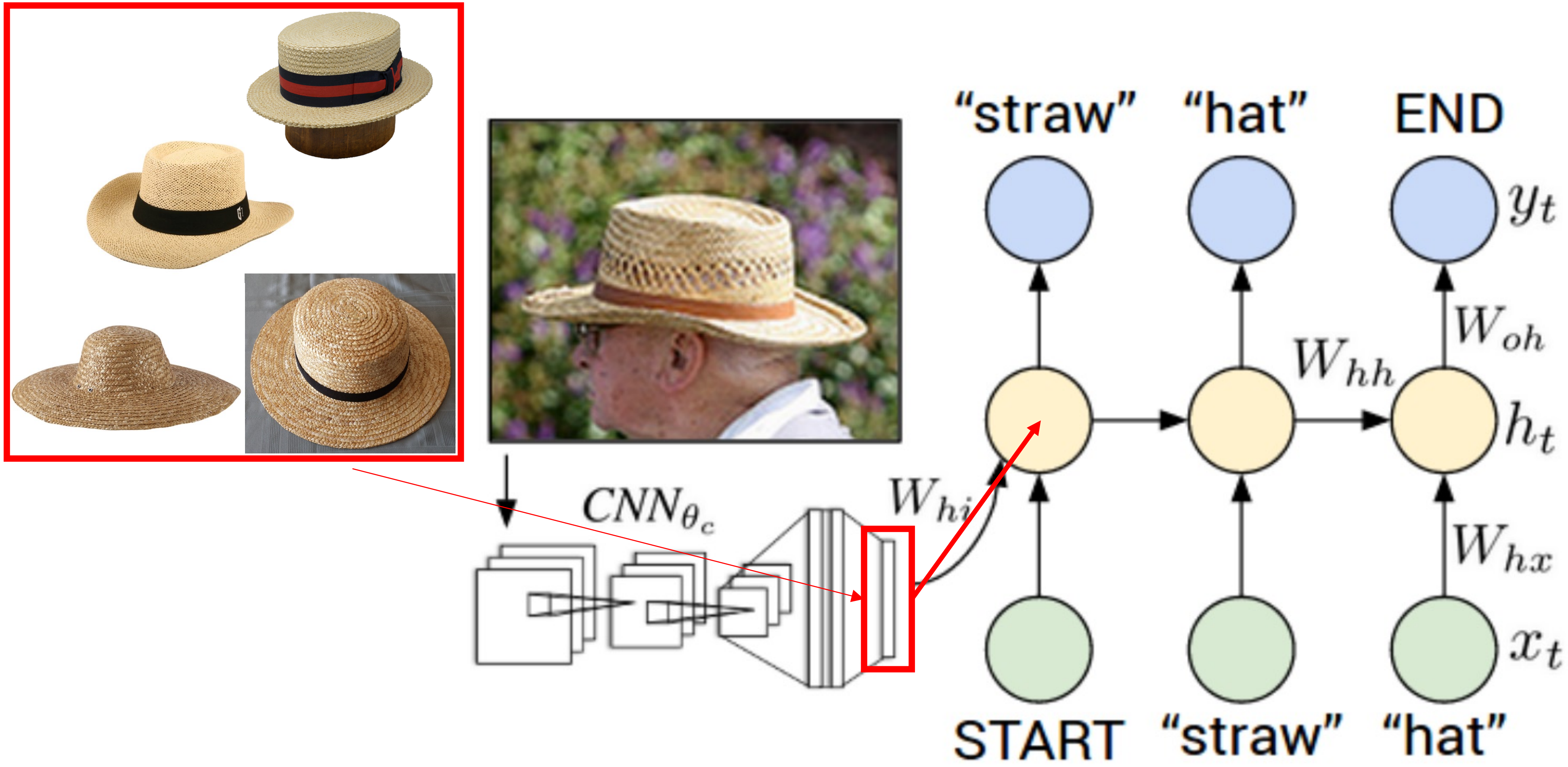


Figure from Karpathy, Andrej, and Li Fei-Fei. "Deep visual-semantic alignments for generating image descriptions." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3128-3137. 2015.

Image captioning

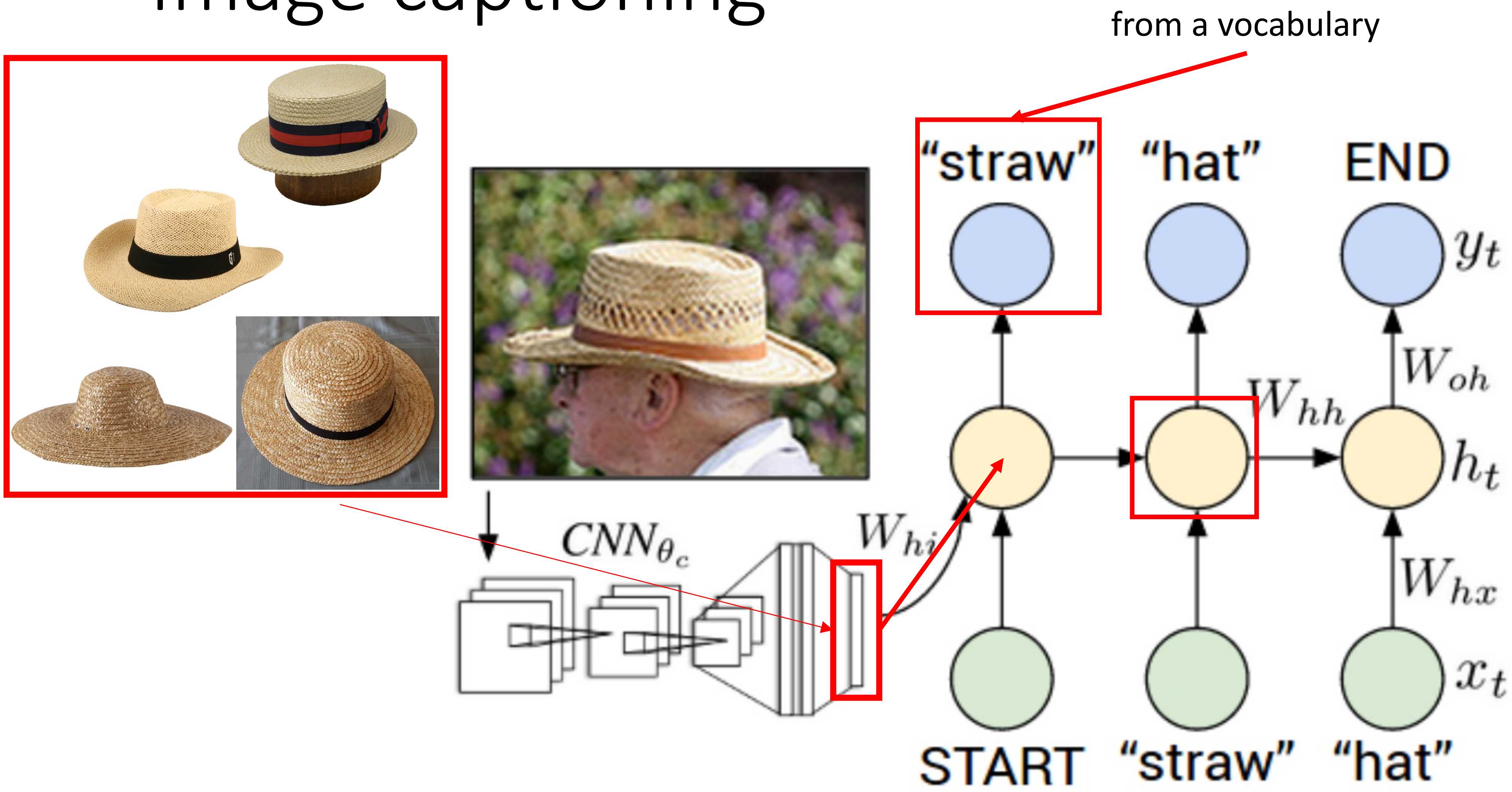


Figure from Karpathy, Andrej, and Li Fei-Fei. "Deep visual-semantic alignments for generating image descriptions." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3128-3137. 2015.

Image captioning

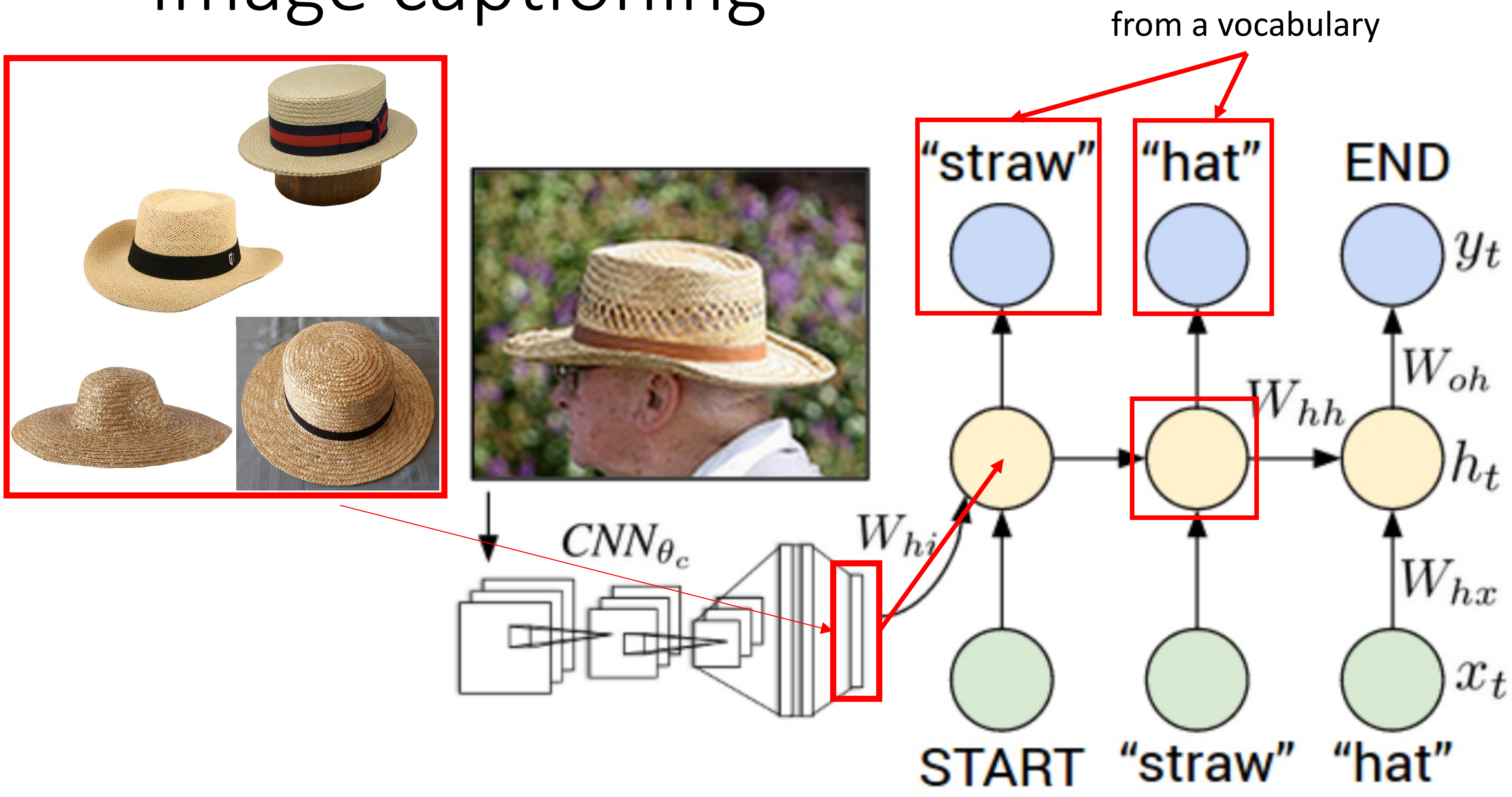
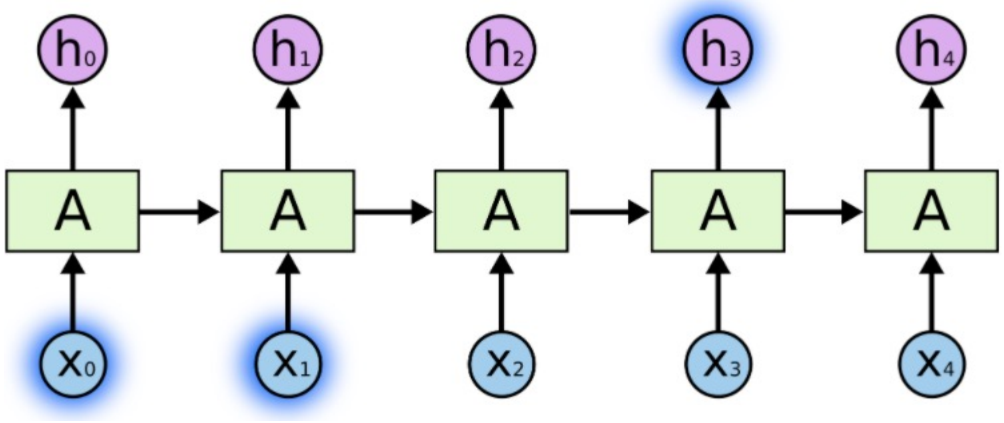


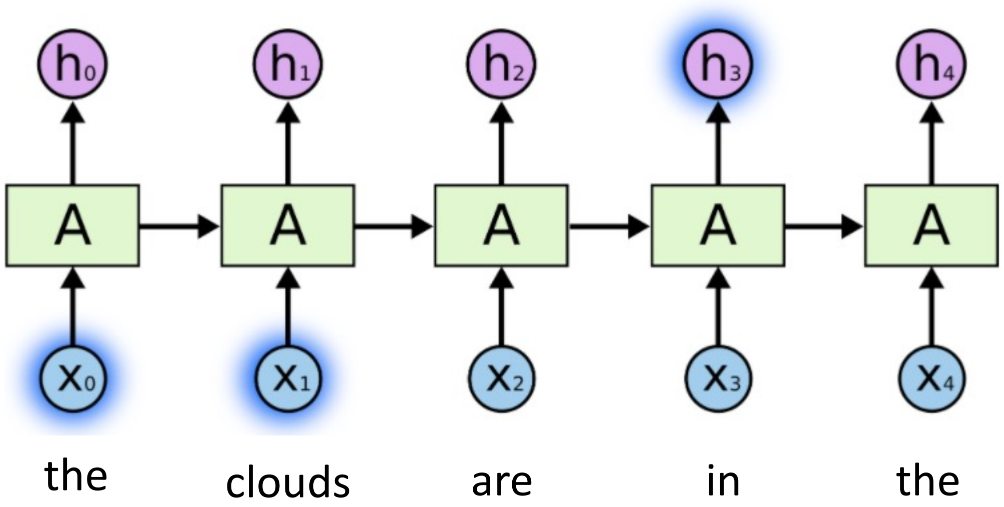
Figure from Karpathy, Andrej, and Li Fei-Fei. "Deep visual-semantic alignments for generating image descriptions." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3128-3137. 2015.

Short-term dependence



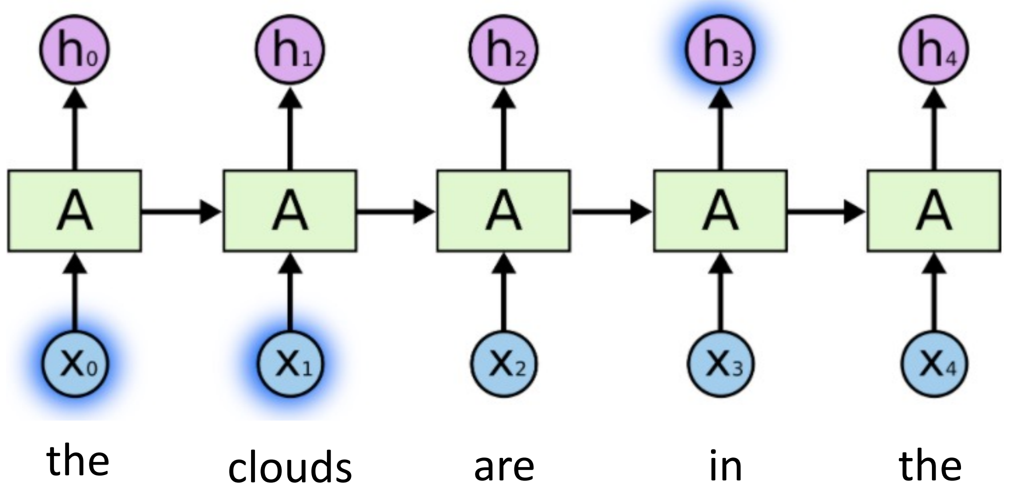
the clouds are in the ???

Short-term dependence



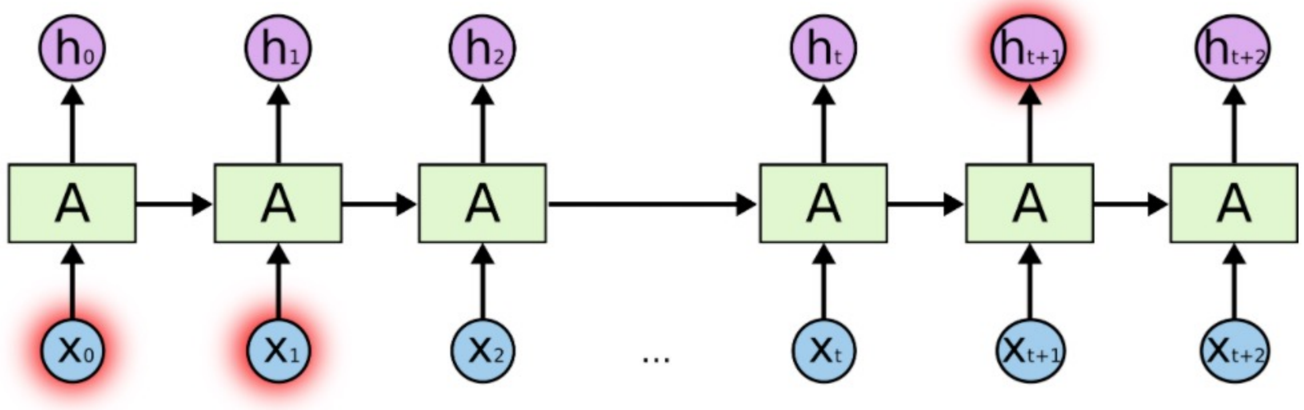
the clouds are in the ???

Short-term dependence



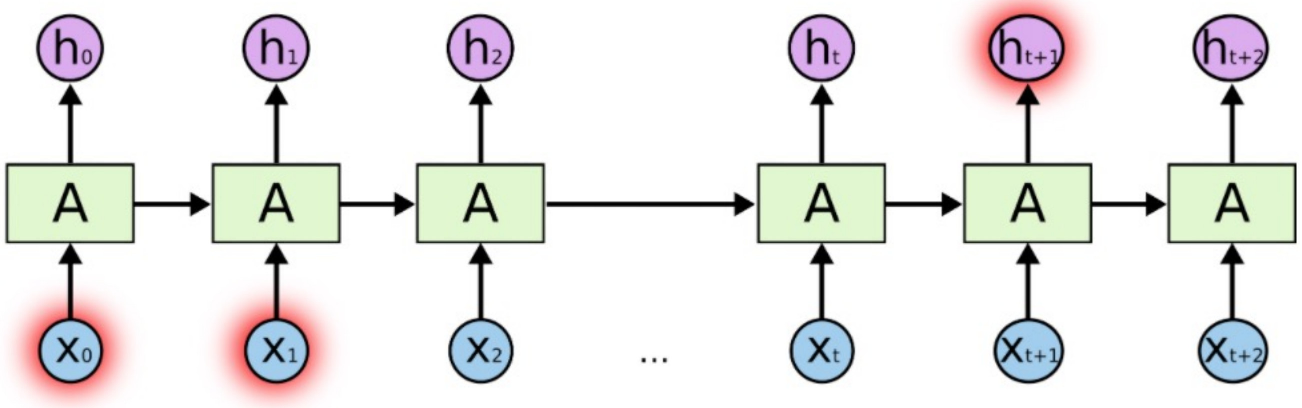
the clouds are in the sky

Long-term dependence



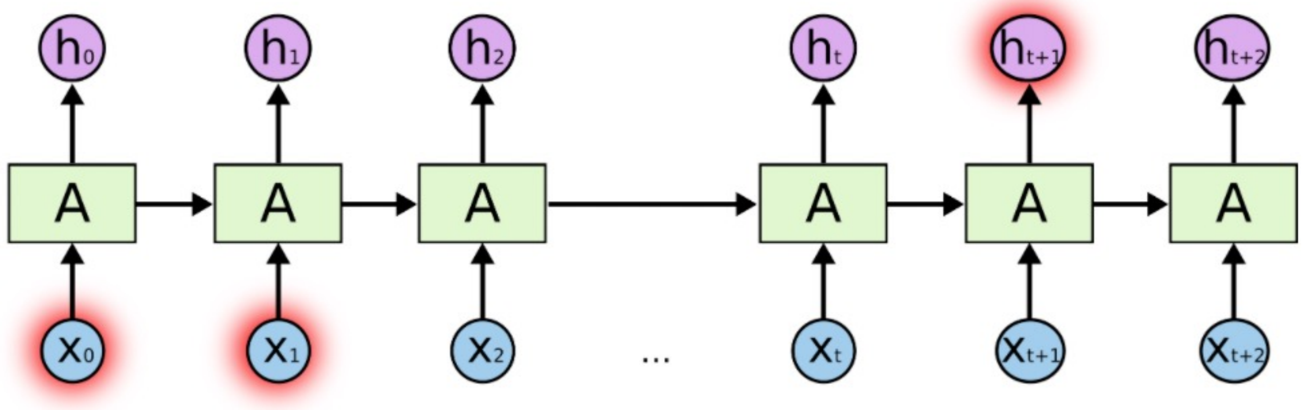
I like this town very much. I started my undergraduate study in 2020 and my major is computer science. I like programming and reading. I usually get up at 7AM and do some exercise. I also go fishing at weekend. I grew up in France.

Long-term dependence



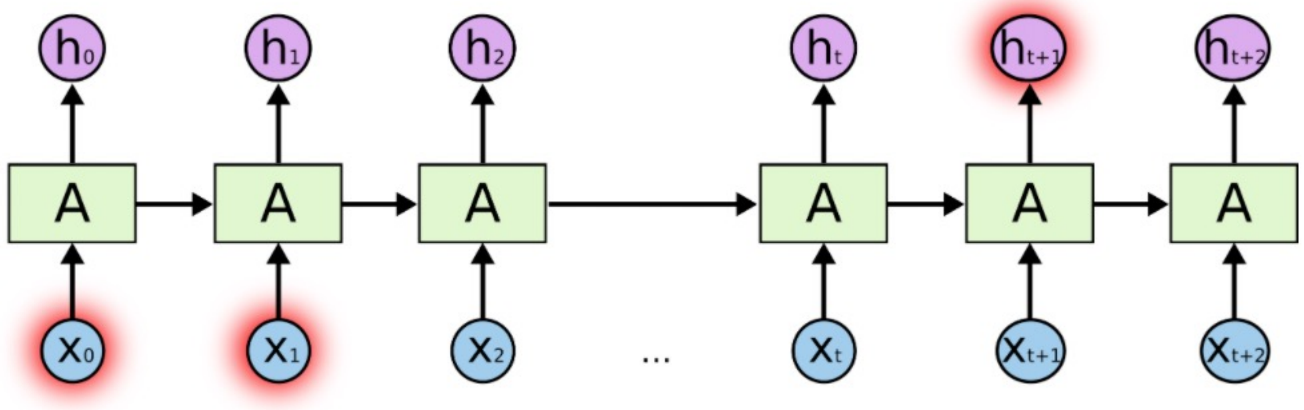
I spent my childhood outdoors.

Long-term dependence



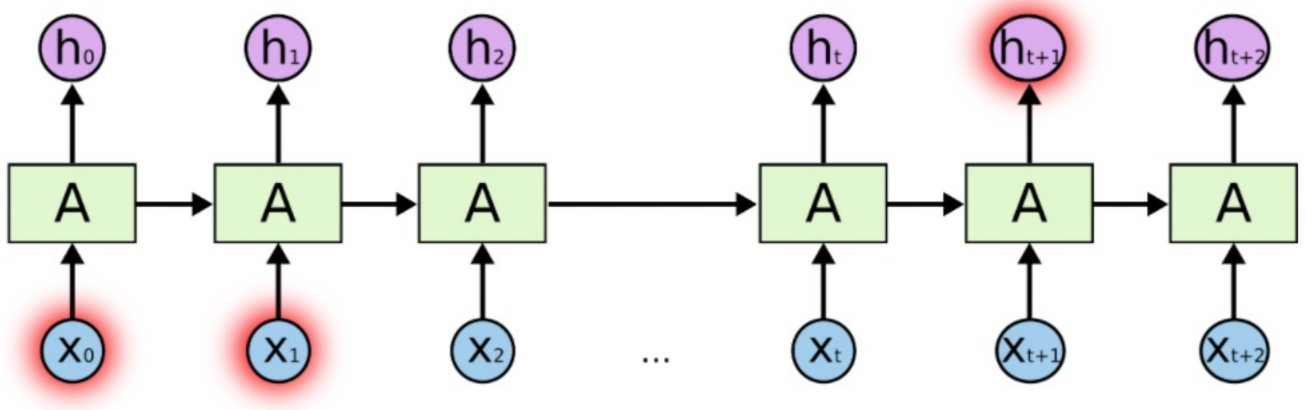
Whether it was riding my bicycle around my neighborhood pretending it was a motorcycle,
making mud cakes, going on treasure hunts, making and selling perfume out of strong smelling flowers,

Long-term dependence



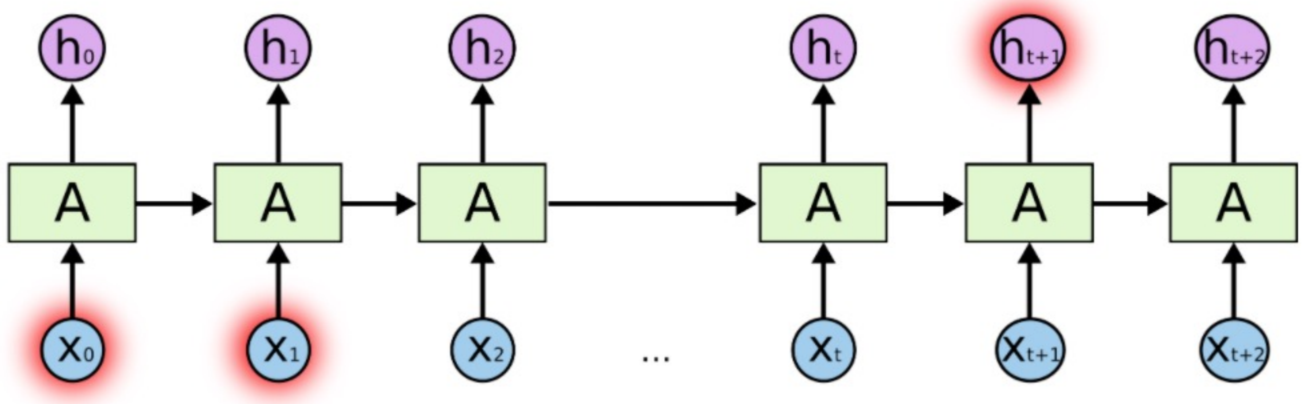
or simply laying on the grass underneath the sun with a soccer ball waiting for someone to come out and play with me, the outdoors was where I spent my childhood and I cannot be more appreciative of it.

Long-term dependence



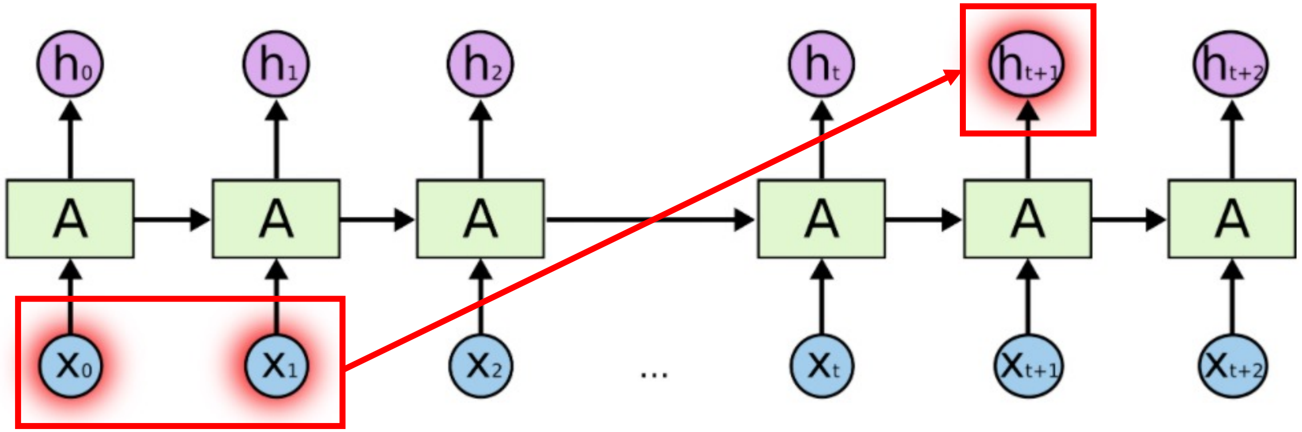
I speak fluent *???*.

Long-term dependence



I like this town very much. I started my undergraduate study in 2020 and my major is computer science. I like programming and reading. I usually get up at 7AM and do some exercise. I also go fishing at weekend. I grew up in France.
I spent my childhood outdoors. Whether it was riding my bicycle around my neighborhood pretending it was a motorcycle, making mud cakes, going on treasure hunts, making and selling perfume out of strong smelling flowers, or simply laying on the grass underneath the sun with a soccer ball waiting for someone to come out and play with me, the outdoors was where I spent my childhood and I cannot be more appreciative of it.
I speak fluent *French*.

Long-term dependence

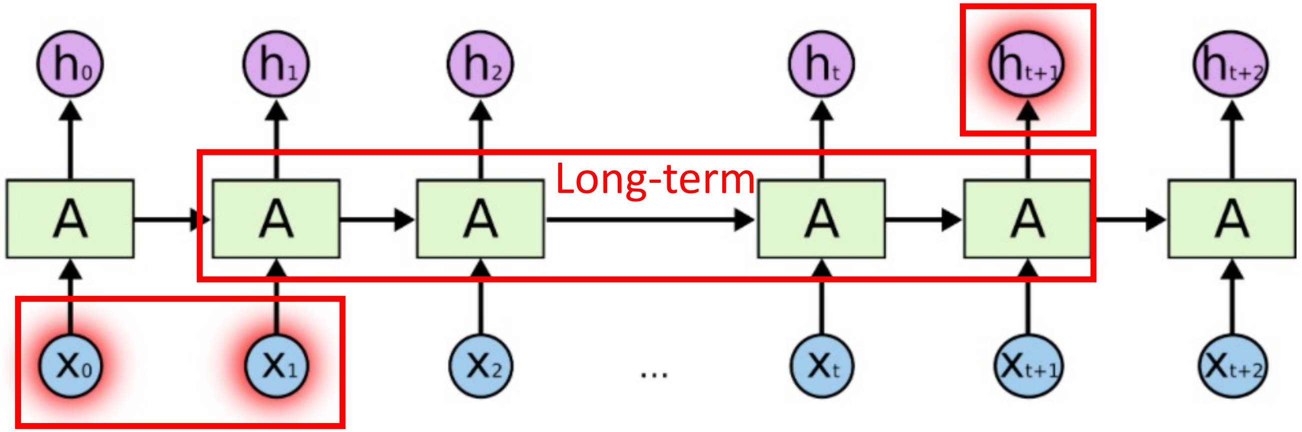


I like this town very much. I started my undergraduate study in 2020 and my major is computer science. I like programming and reading. I usually get up at 7AM and do some exercise. I also go fishing at weekend. I grew up in France.

I spent my childhood outdoors. Whether it was riding my bicycle around my neighborhood pretending it was a motorcycle, making mud cakes, going on treasure hunts, making and selling perfume out of strong smelling flowers, or simply laying on the grass underneath the sun with a soccer ball waiting for someone to come out and play with me, the outdoors was where I spent my childhood and I cannot be more appreciative of it.

I speak fluent *French*.

Long-term dependence



I like this town very much. I started my undergraduate study in 2020 and my major is computer science. I like programming and reading. I usually get up at 7AM and do some exercise. I also go fishing at weekend. I grew up in France.

I spent my childhood outdoors. Whether it was riding my bicycle around my neighborhood pretending it was a motorcycle, making mud cakes, going on treasure hunts, making and selling perfume out of strong smelling flowers, or simply laying on the grass underneath the sun with a soccer ball waiting for someone to come out and play with me, the outdoors was where I spent my childhood and I cannot be more appreciative of it.

I speak fluent *French*.

Reading

- Reference slides at http://cs231n.stanford.edu/slides/2020/lecture_10.pdf