Overview

- Sequences
- Amped up RNNs (LSTMs + GRUs)
- Encoder - Decoder (Seq2Seq)

Deep Learning and Sequences

- Sequences
  - Variable length
  - Relationships between elements of sequence
- Examples
  - Text
  - Time Series
- Models
  - Continuous Bag of Words (CBOW)
  - 1D CNN
  - Recurrent Neural Network (RNN)

Deep Learning and Sequences

- Sequences
  - Variable length
  - Relationships between elements of sequence
- Examples
  - Text
  - Time Series
- Models
  - Continuous Bag of Words (CBOW)
  - 1D CNN
  - Recurrent Neural Network (RNN)
  - Attention (Next!)

Deep Learning and Sequences

- Models
    - Continuous Bag of Words (CBOW)
    - 1D CNN
    - Recurrent Neural Network (RNN)
- Average feature vectors together to get fixed length input
- Loose a lot information about the sequence

Deep Learning and Sequences

- Models
    - Continuous Bag of Words (CBOW)
    - 1D CNN
    - Recurrent Neural Network (RNN)
- Doesn’t care about sequence length
- Uses filters to construct features from local interactions
- Difficult to capture long range dependencies

Deep Learning and Sequences

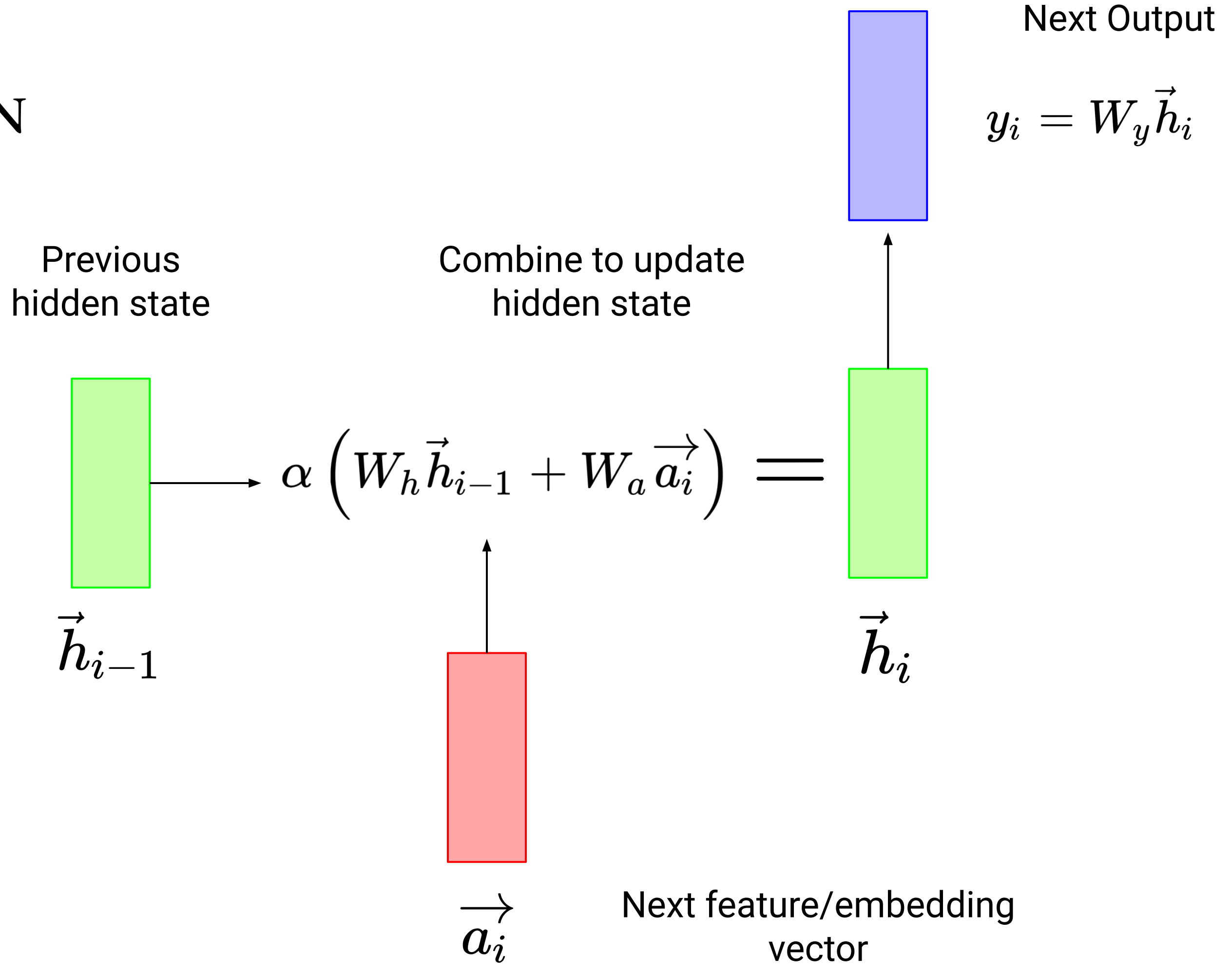
- Models
  - Continuous Bag of Words (CBOW)
  - 1D CNN
  - Recurrent Neural Network (RNN)
- Updates a hidden state as the sequence is fed into the RNN
- Vanishing/Exploding gradient problem
- Doesn’t have great long-term memory
- Slow (can’t parallelize updates to a hidden state)

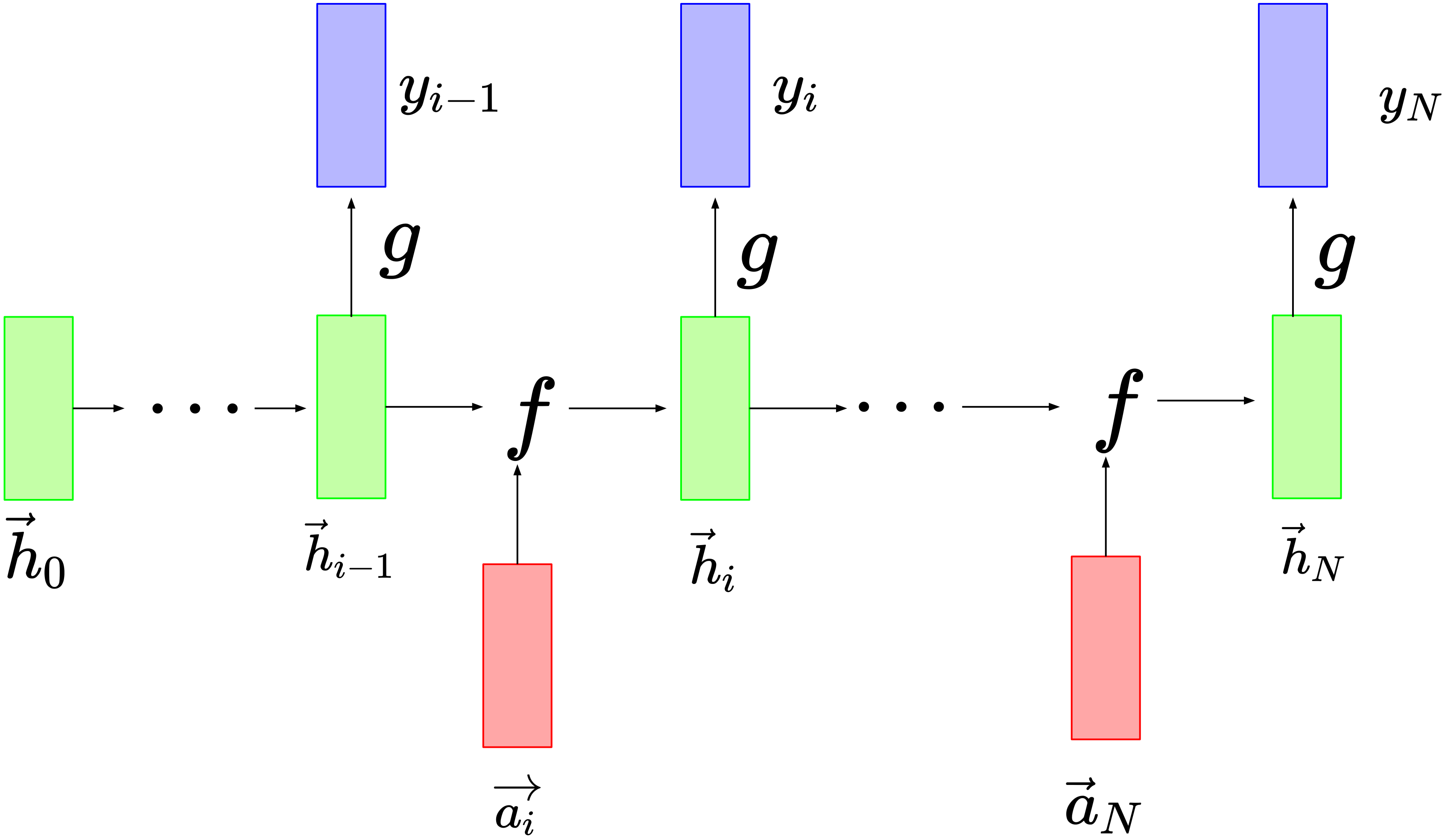
Deep Learning and Sequences

- Models
  - Continuous Bag of Words (CBOW)
  - 1D CNN
  - Recurrent Neural Network (RNN)
  - LSTMs, GRUs, and more!
- Fancier updates to a hidden state as the sequence is fed into the NN
- Helps with Vanishing/Exploding gradient problem
- Helps with long-term memory
- Still Slow (can't parallelize updates to a hidden state)

RNNs

- Vanilla RNN

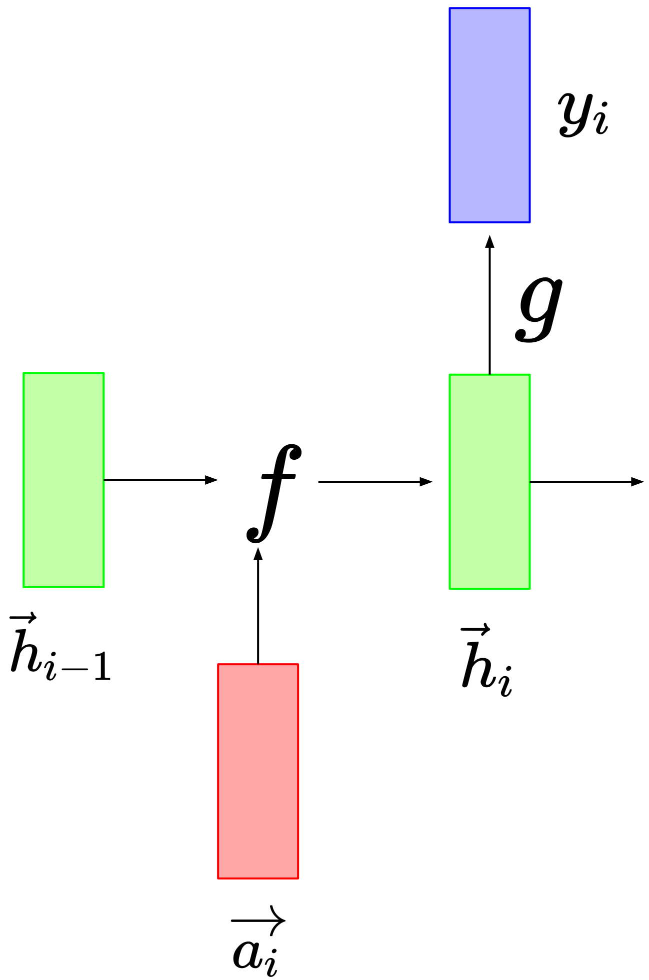




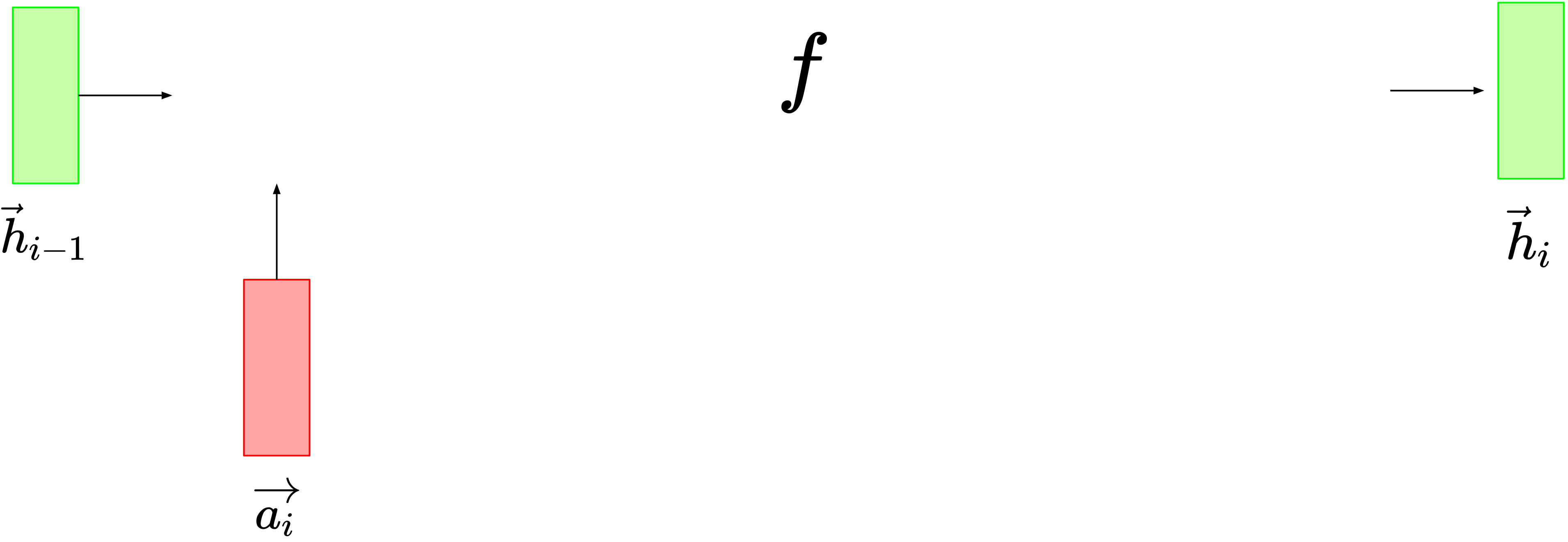
y_{i-1}
y_i
y_N
g
g
g
f
f
$\vec{h}_0$
$\vec{h}_{i-1}$
$\vec{h}_i$
$\vec{h}_N$
$\overrightarrow{a_i}$
$\vec{a}_N$

The GRU

- Gated Recurrent Unit
- Idea: Change the function *f* to address common RNN problems

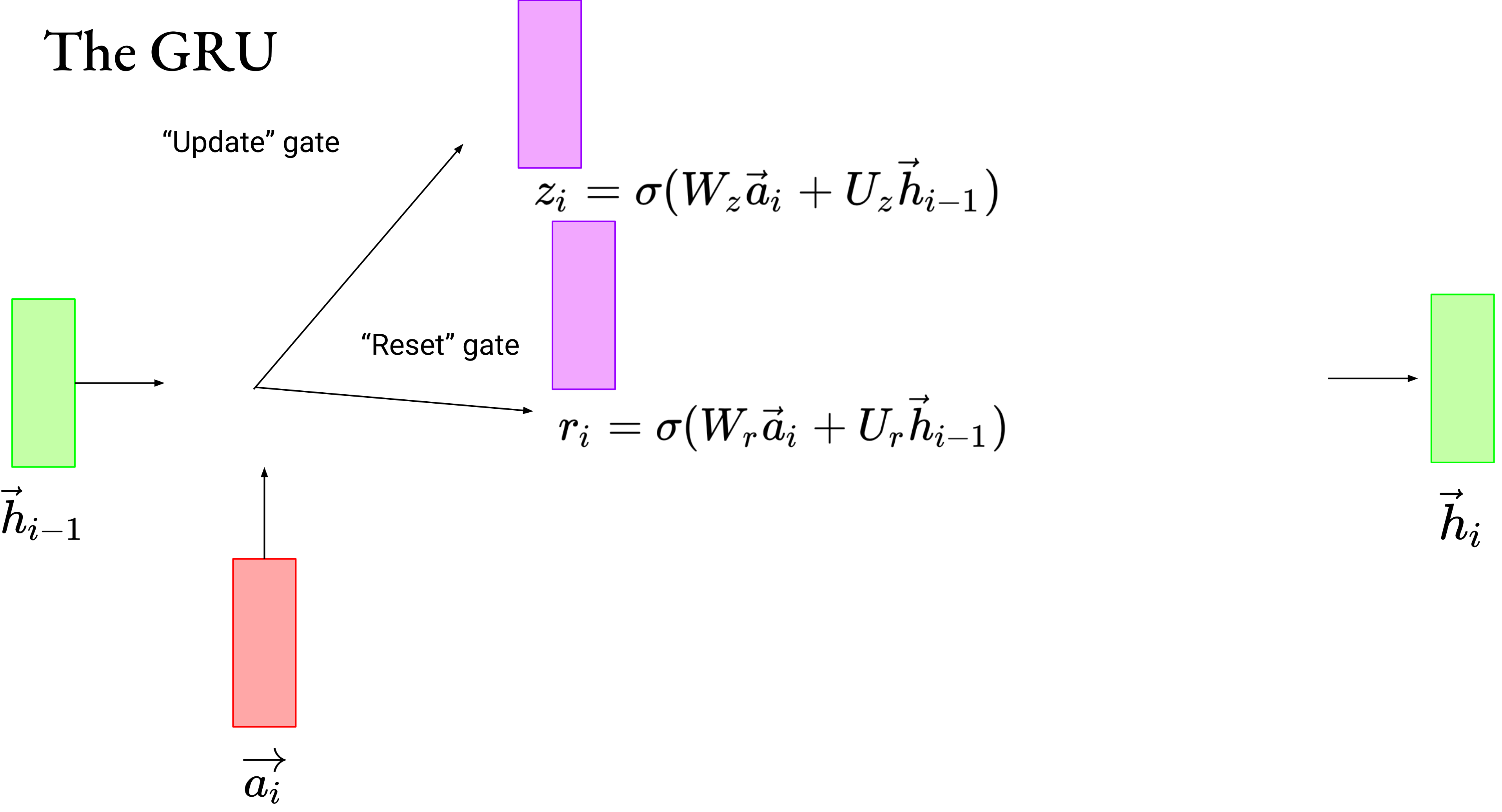


The GRU



The GRU

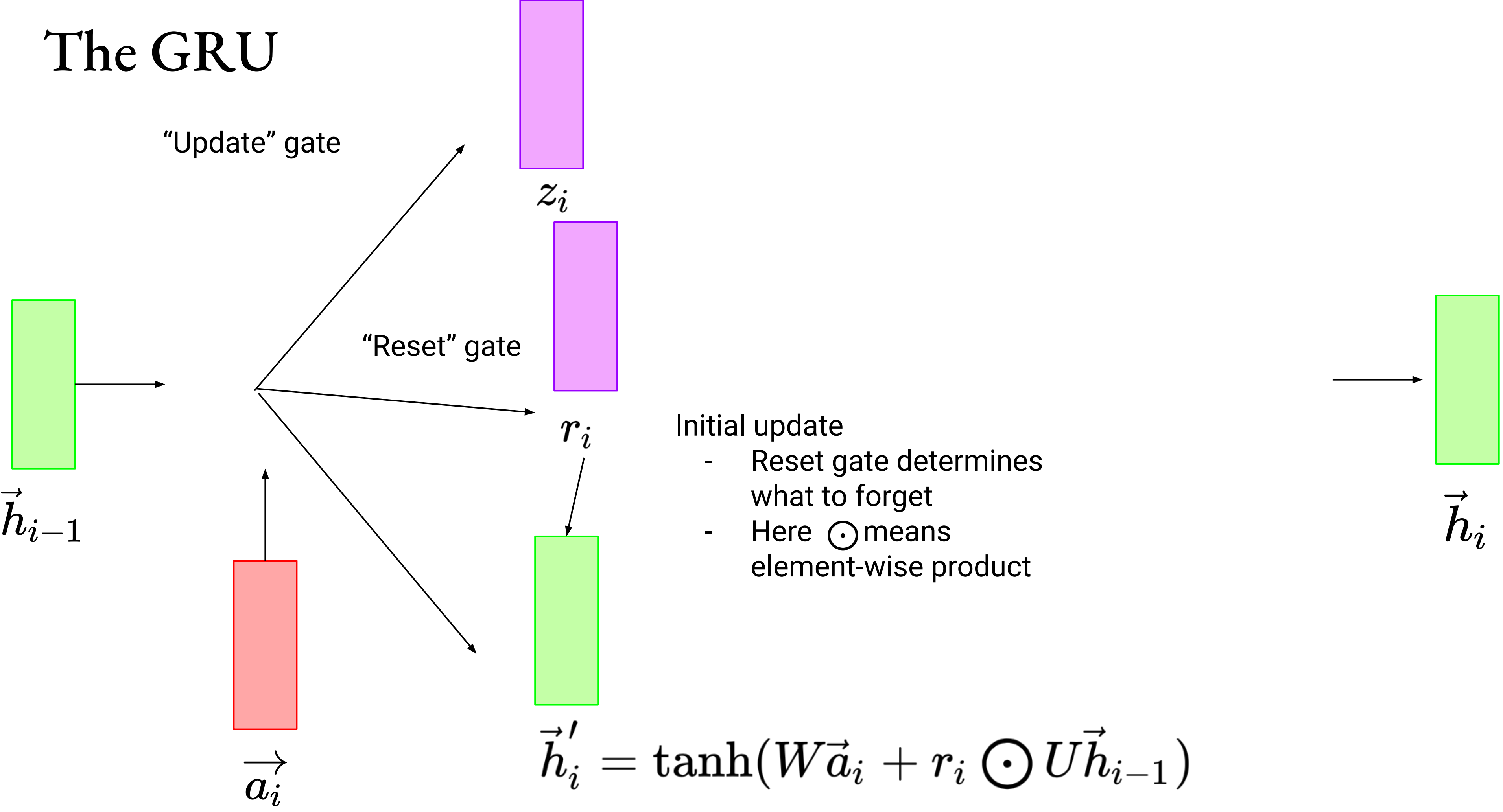
The GRU



$$z_i = \sigma(W_z \vec{a}_i + U_z \vec{h}_{i-1})$$

$$r_i = \sigma(W_r \vec{a}_i + U_r \vec{h}_{i-1})$$

The GRU



Initial update

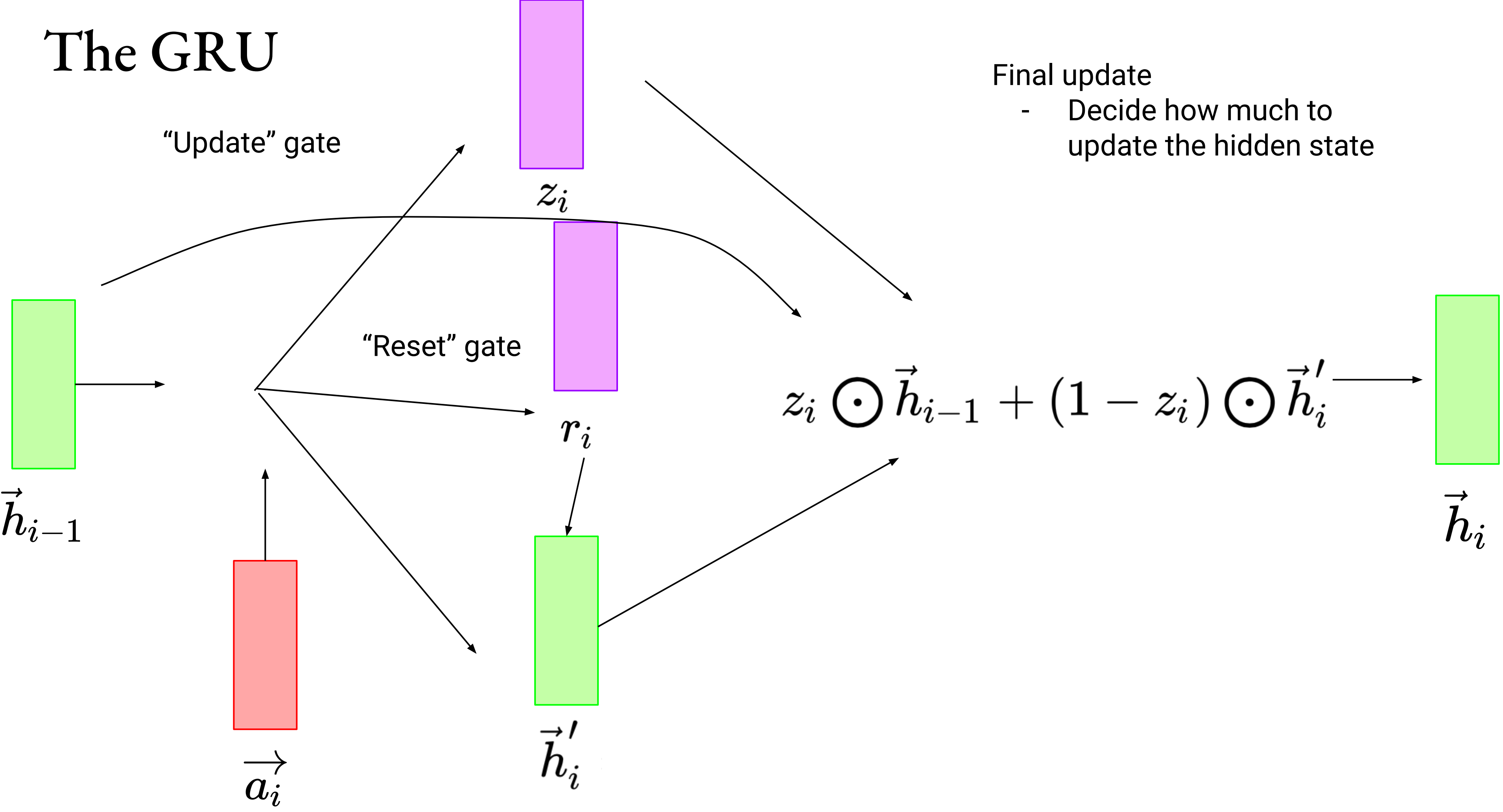
- Reset gate determines what to forget
- Here $\odot$ means element-wise product

$$\vec{h}'_i = \tanh(W\vec{a}_i + r_i \odot U\vec{h}_{i-1})$$

The GRU

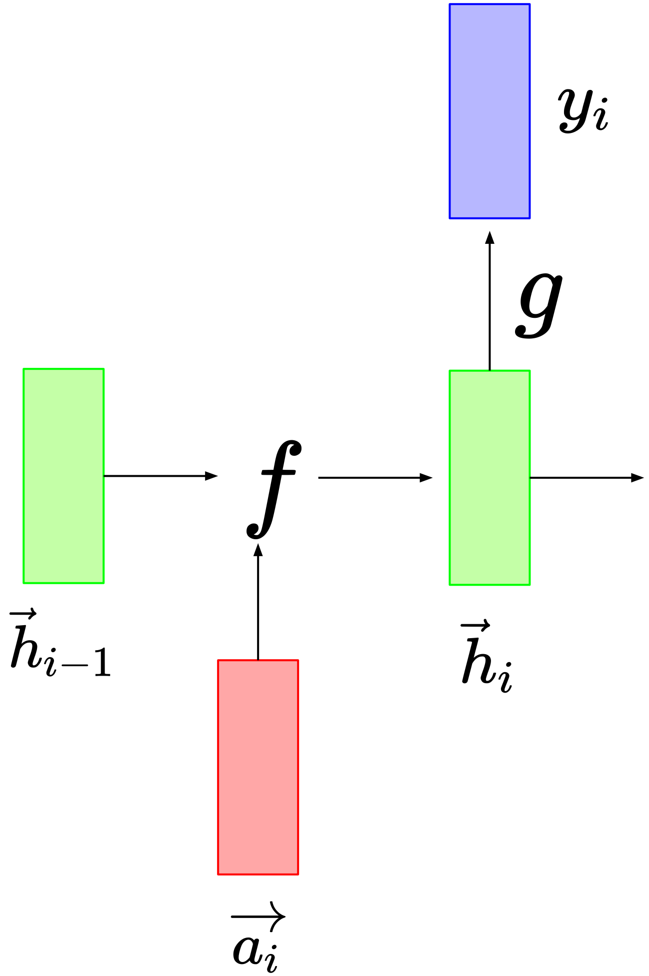
Final update

- Decide how much to update the hidden state



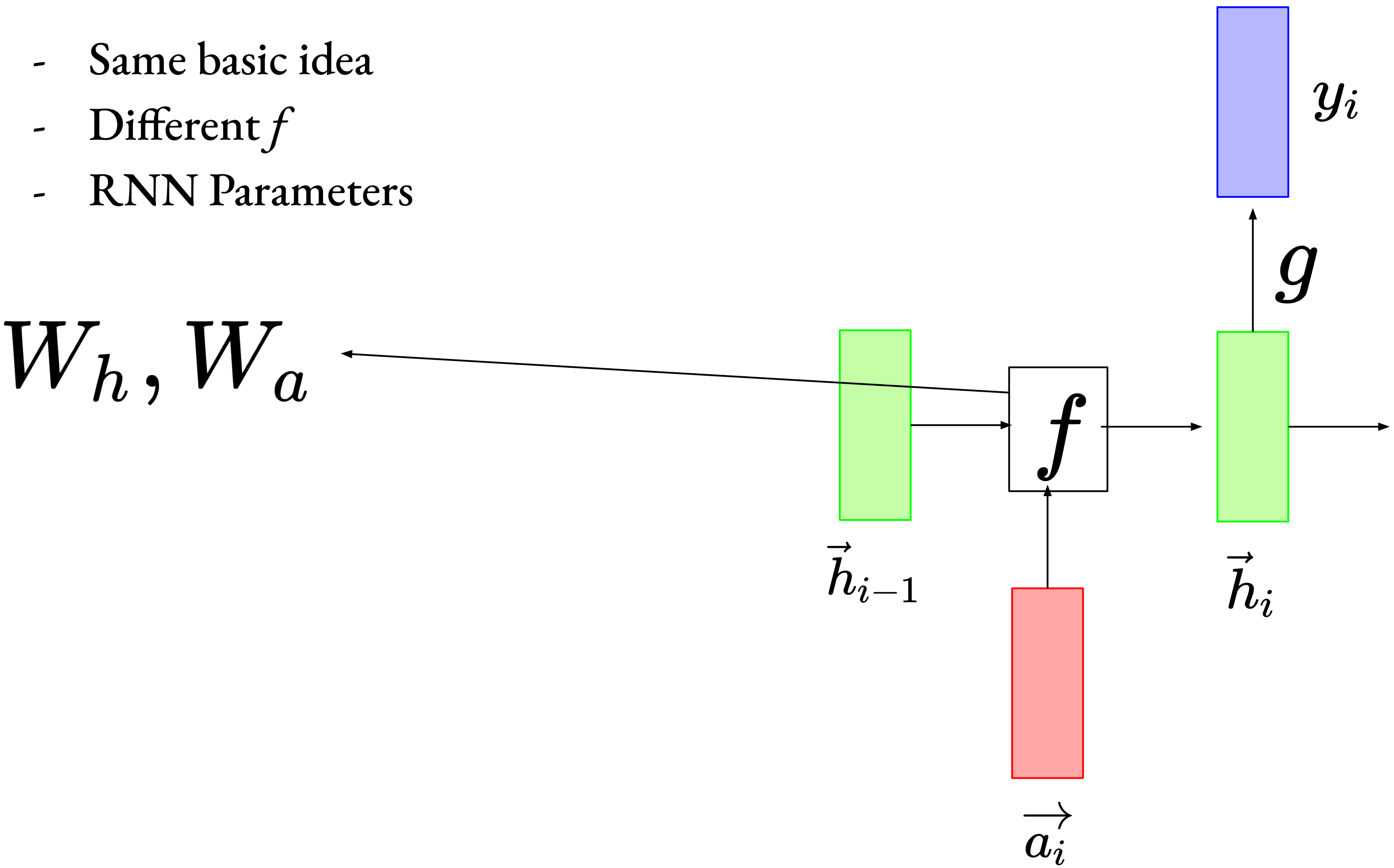
Recurrent Unit

- Same basic idea
- Different f



Recurrent Unit

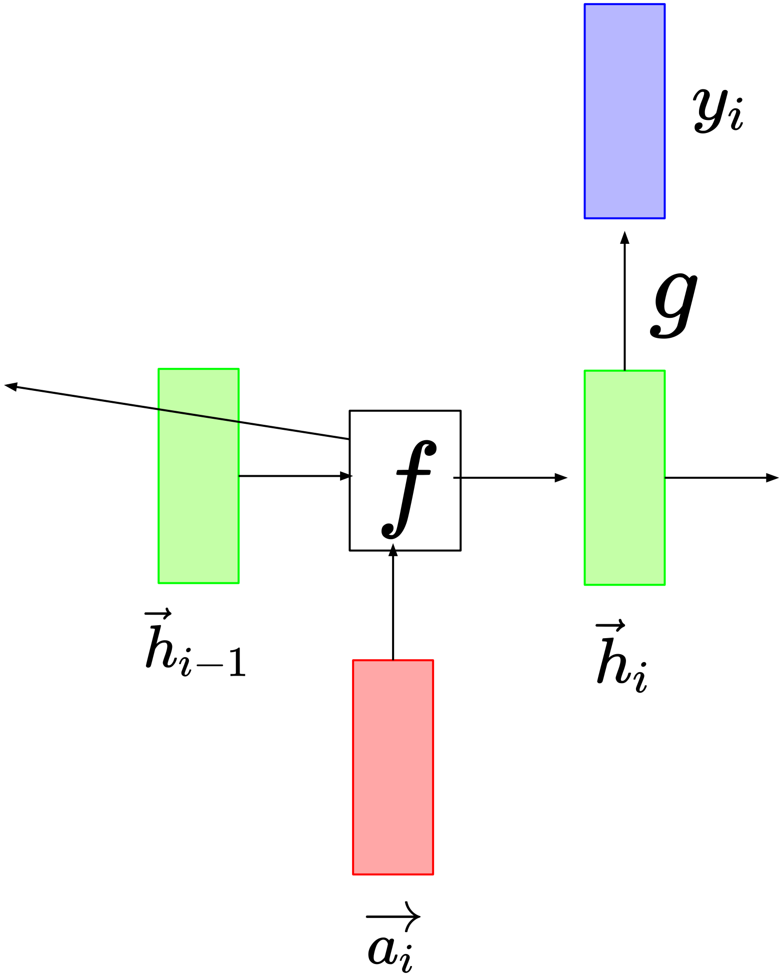
- Same basic idea
- Different f
- RNN Parameters



Recurrent Unit

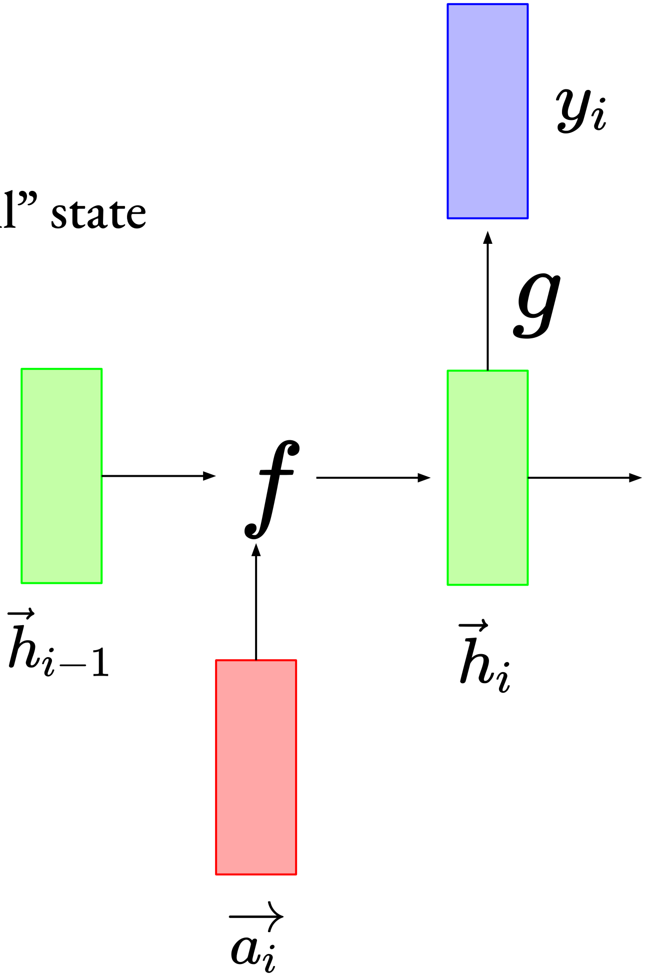
- Same basic idea
- Different *f*
- GRU Parameters

$$W, U, W_z, U_z W_a, U_a$$



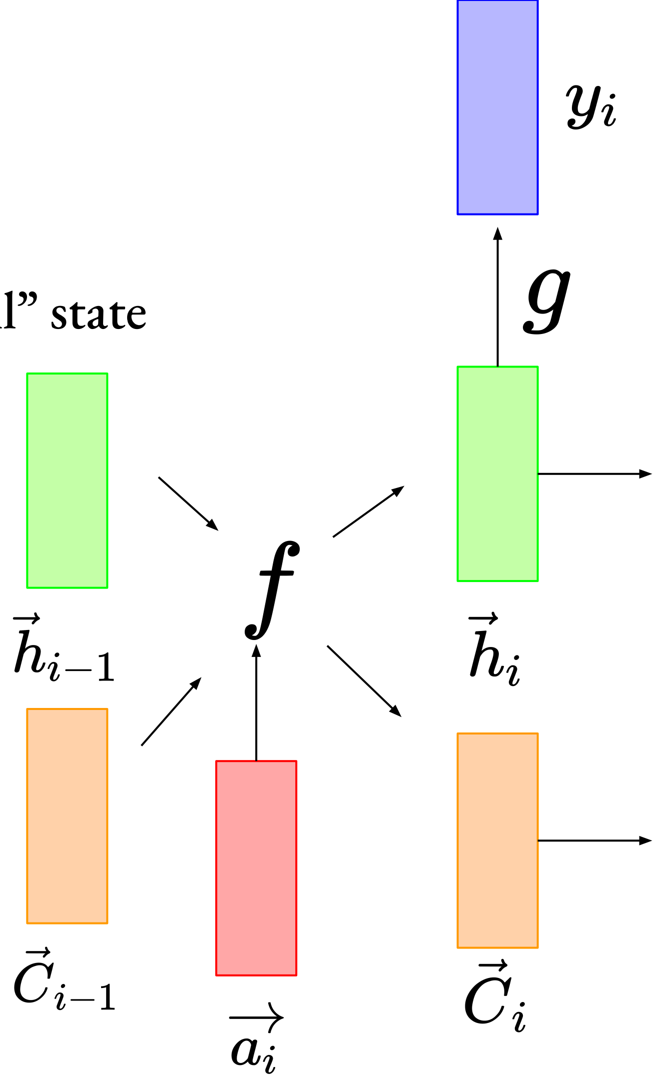
The LSTM

- Long Short-Term Memory
- Idea: Change the function *f* to address common RNN problems and add a "cell" state

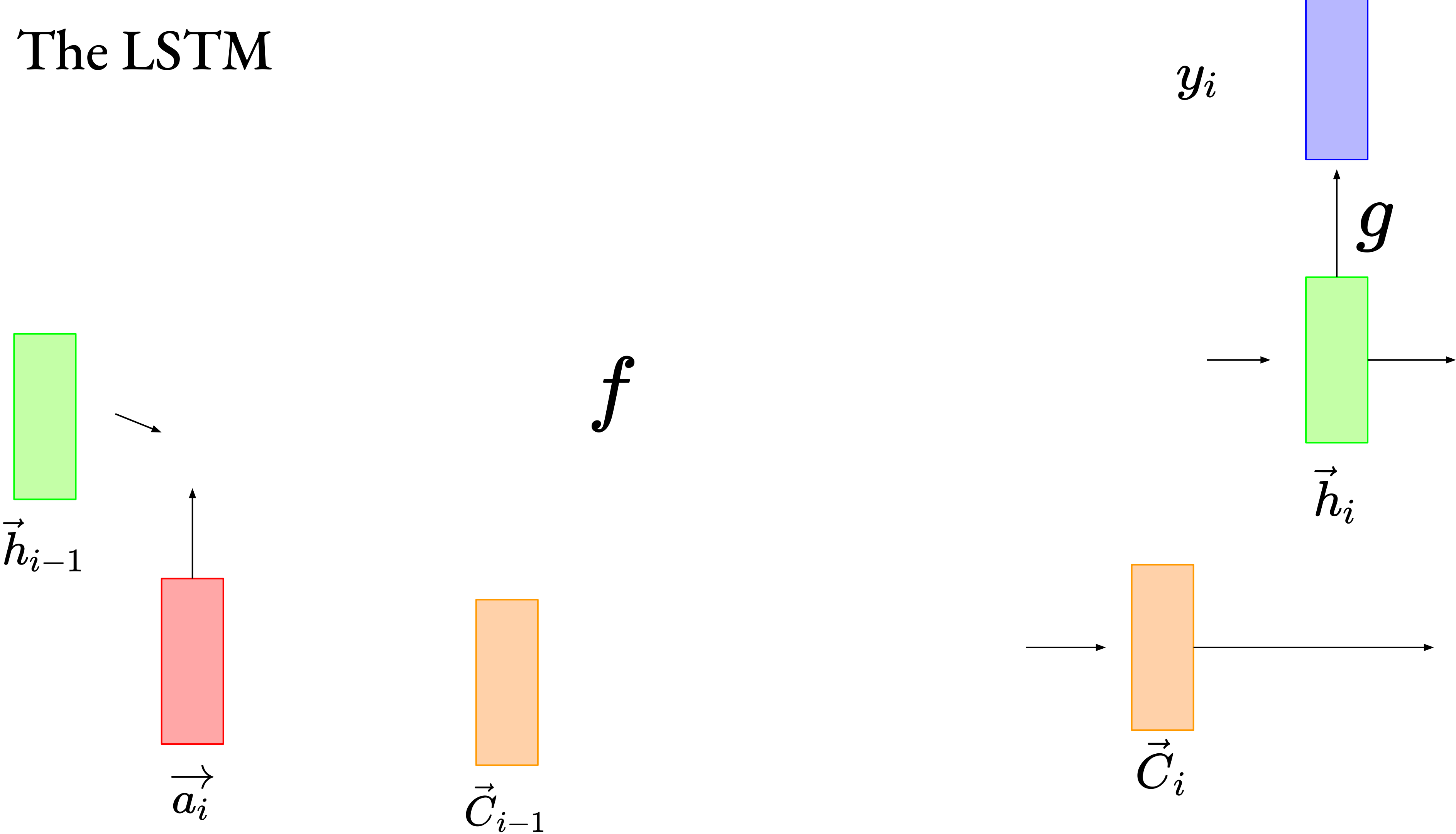


The LSTM

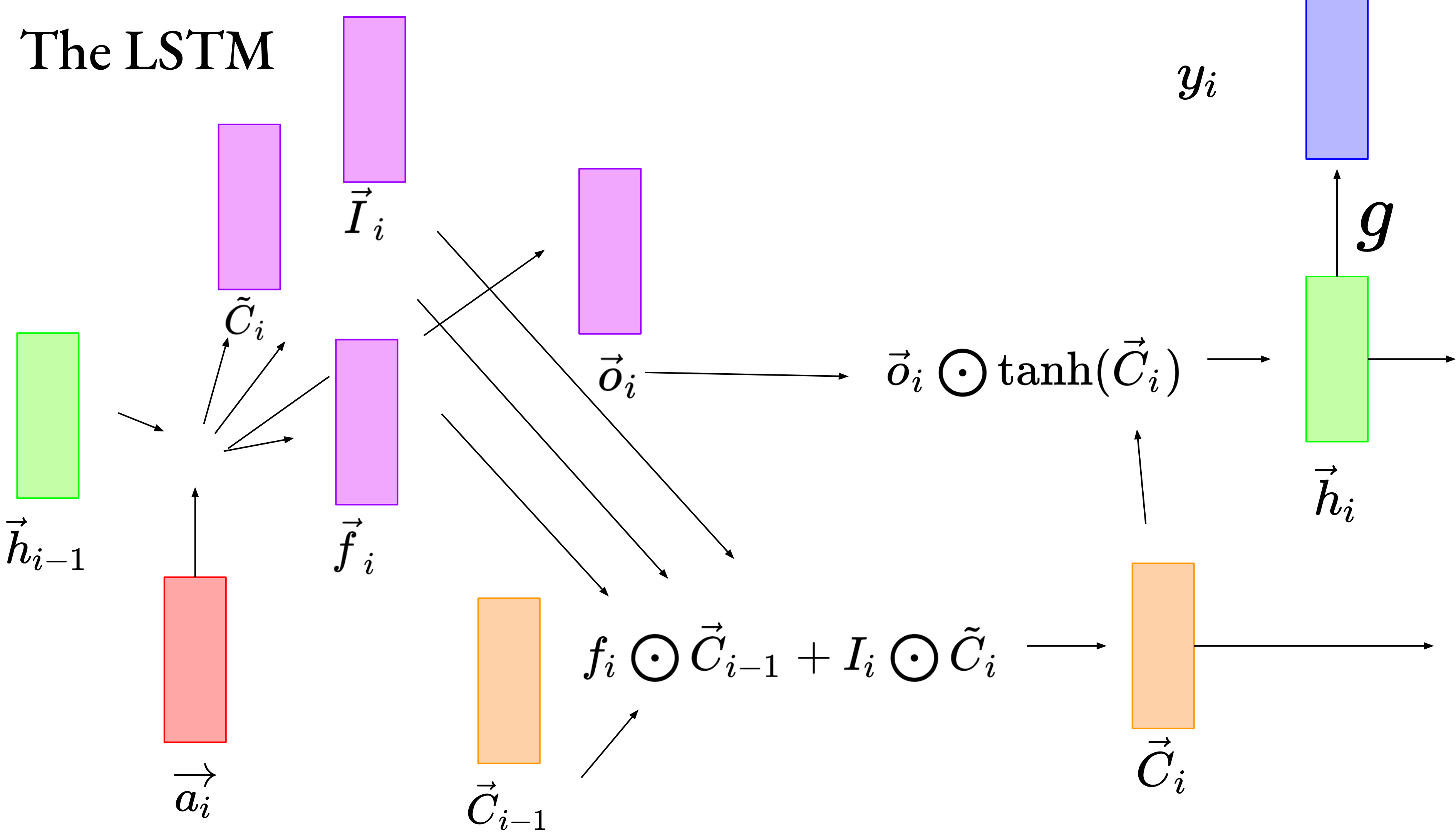
- Long Short-Term Memory
- Idea: Change the function *f* to address common RNN problems and add a “cell” state



The LSTM

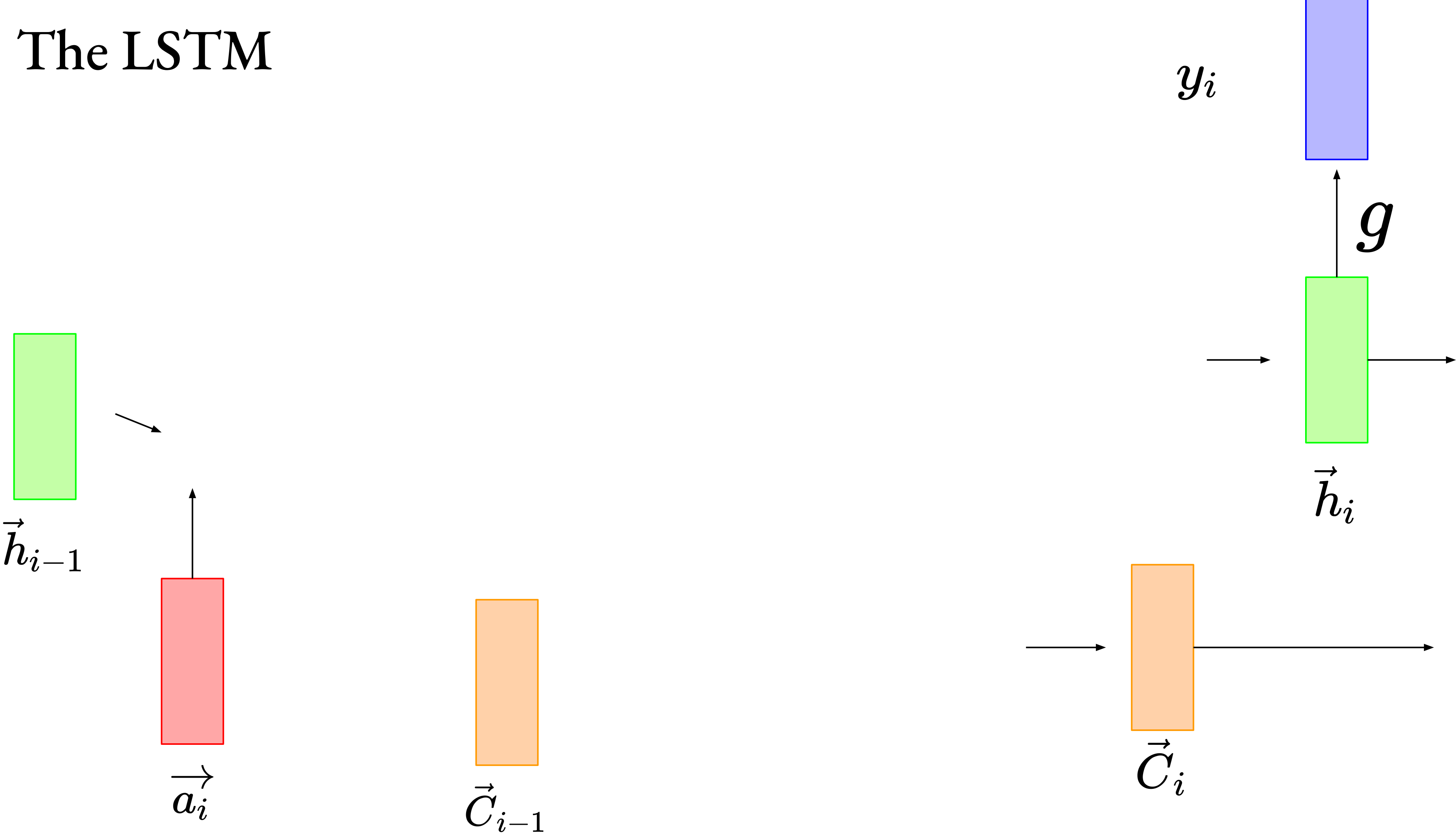


The LSTM

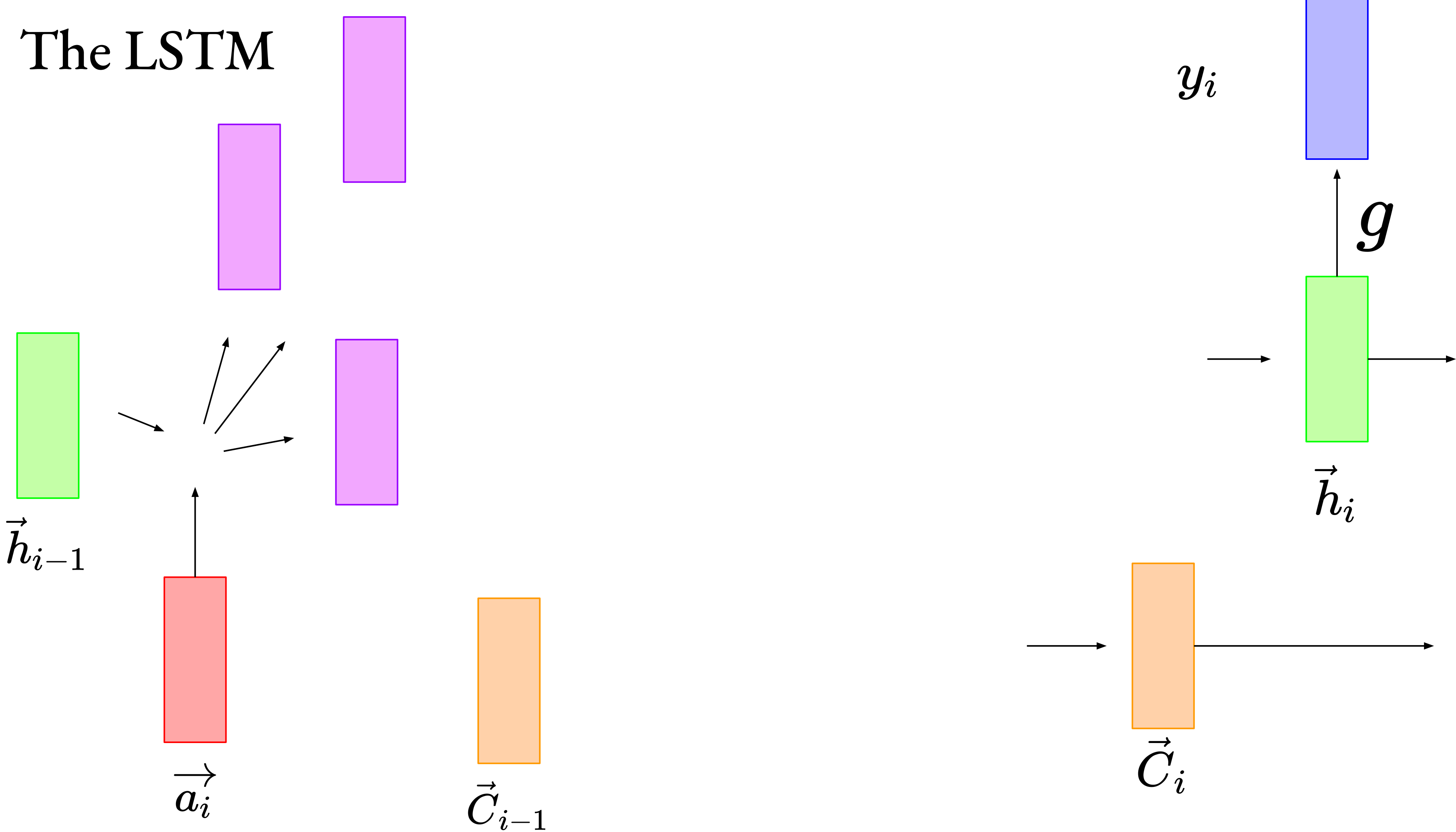


y_i
g
$\vec{I}_i$
$\tilde{C}_i$
$\vec{o}_i$
$\vec{o}_i \odot \tanh(\vec{C}_i)$
$\vec{h}_{i-1}$
$\vec{f}_i$
$\vec{h}_i$
$f_i \odot \vec{C}_{i-1} + I_i \odot \tilde{C}_i$
$\overrightarrow{a_i}$
$\vec{C}_{i-1}$
$\vec{C}_i$

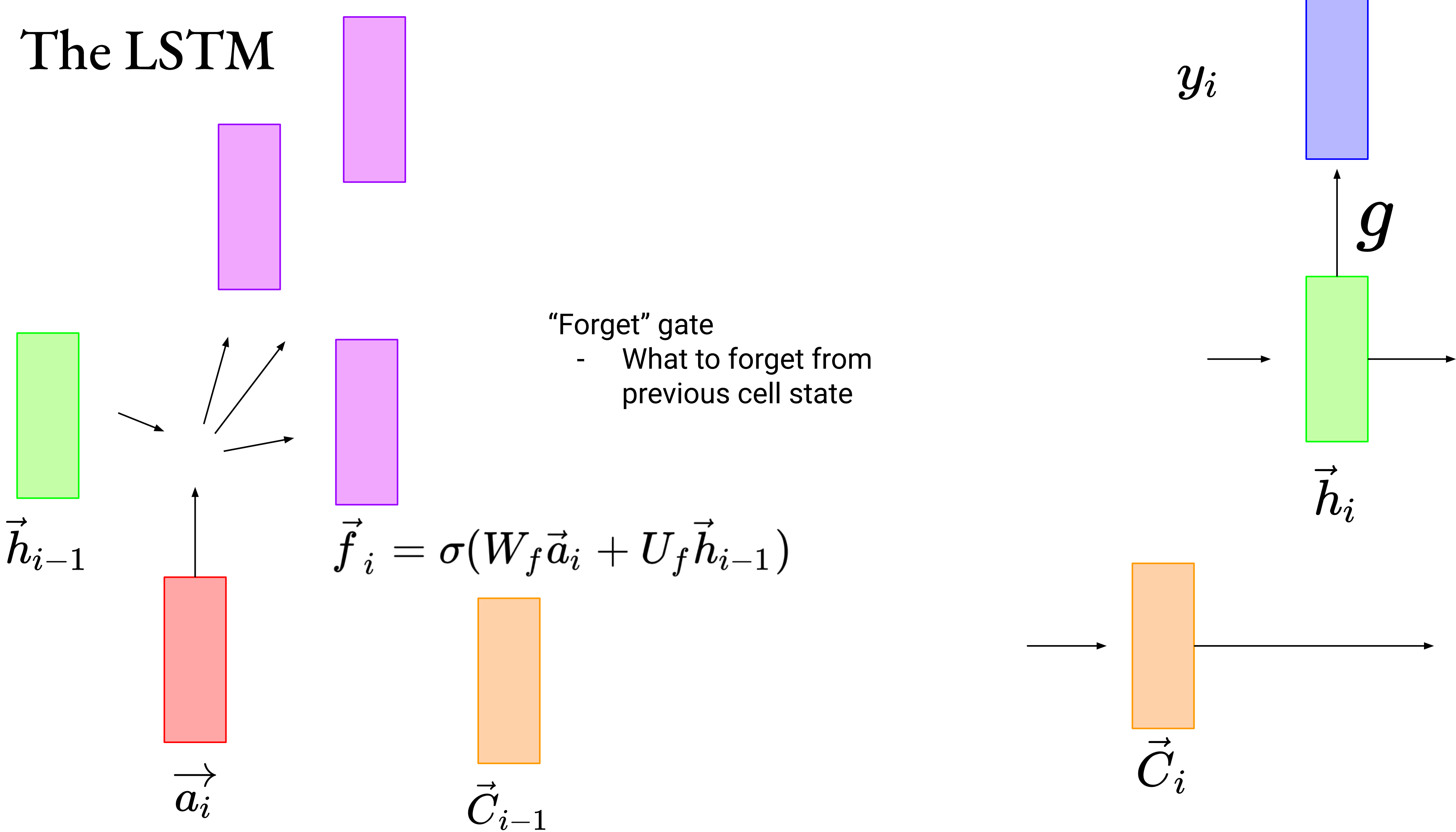
The LSTM



The LSTM



The LSTM

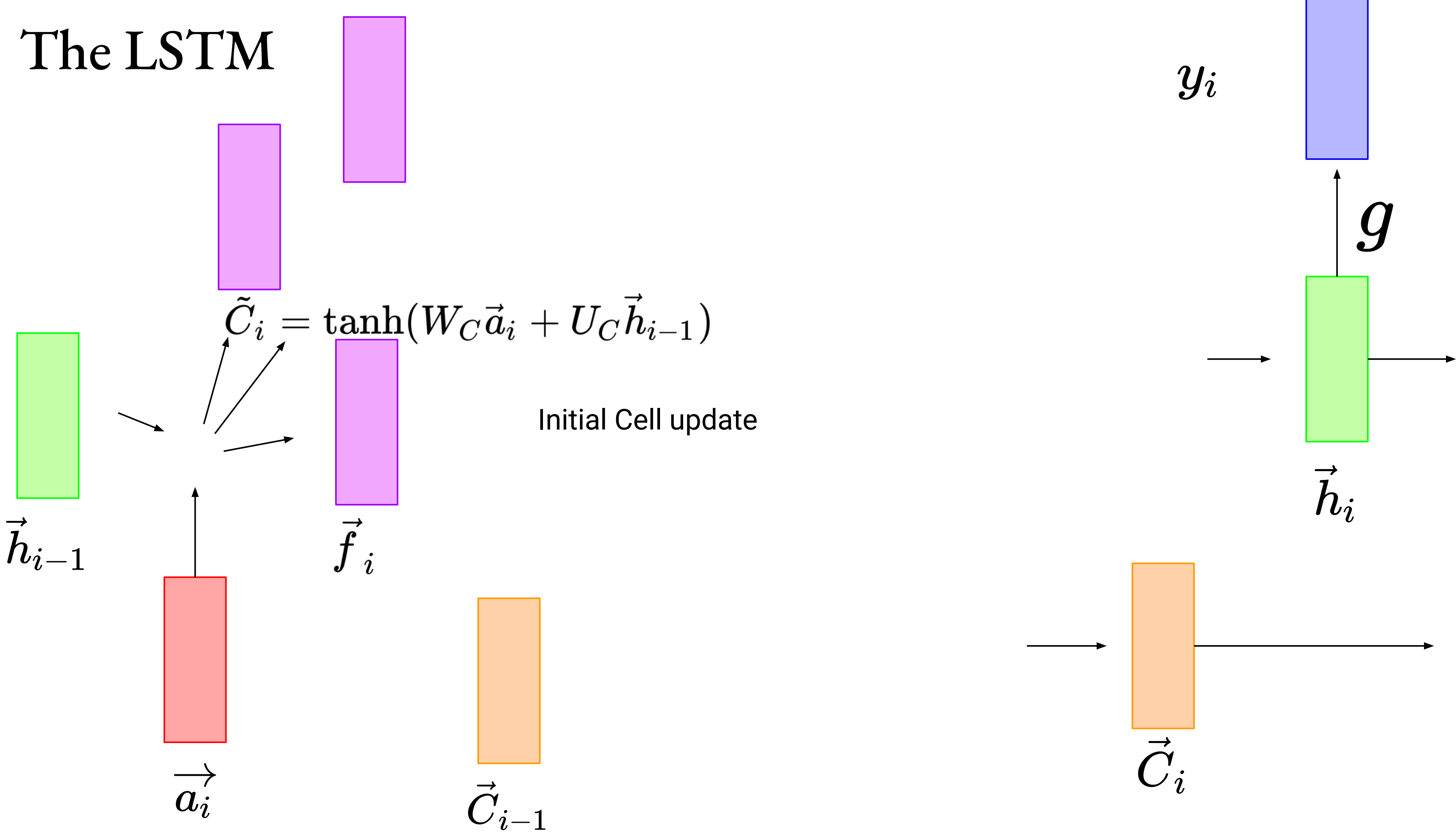


"Forget" gate

- What to forget from previous cell state

$$\vec{f}_i = \sigma(W_f \vec{a}_i + U_f \vec{h}_{i-1})$$

The LSTM

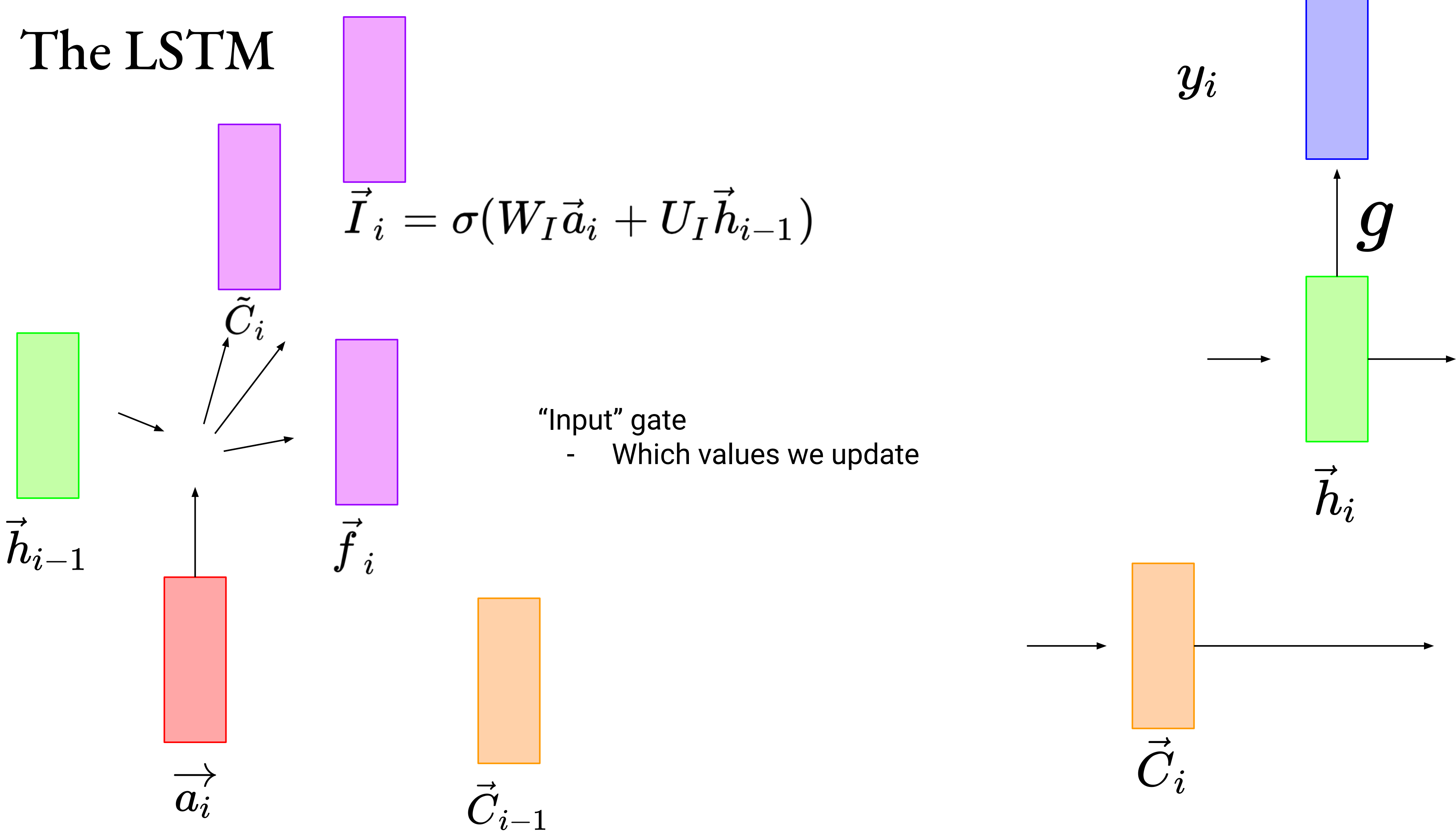


$$\tilde{C}_i = \tanh(W_C \vec{a}_i + U_C \vec{h}_{i-1})$$

Initial Cell update

The LSTM

$$\vec{I}_i = \sigma(W_I \vec{a}_i + U_I \vec{h}_{i-1})$$



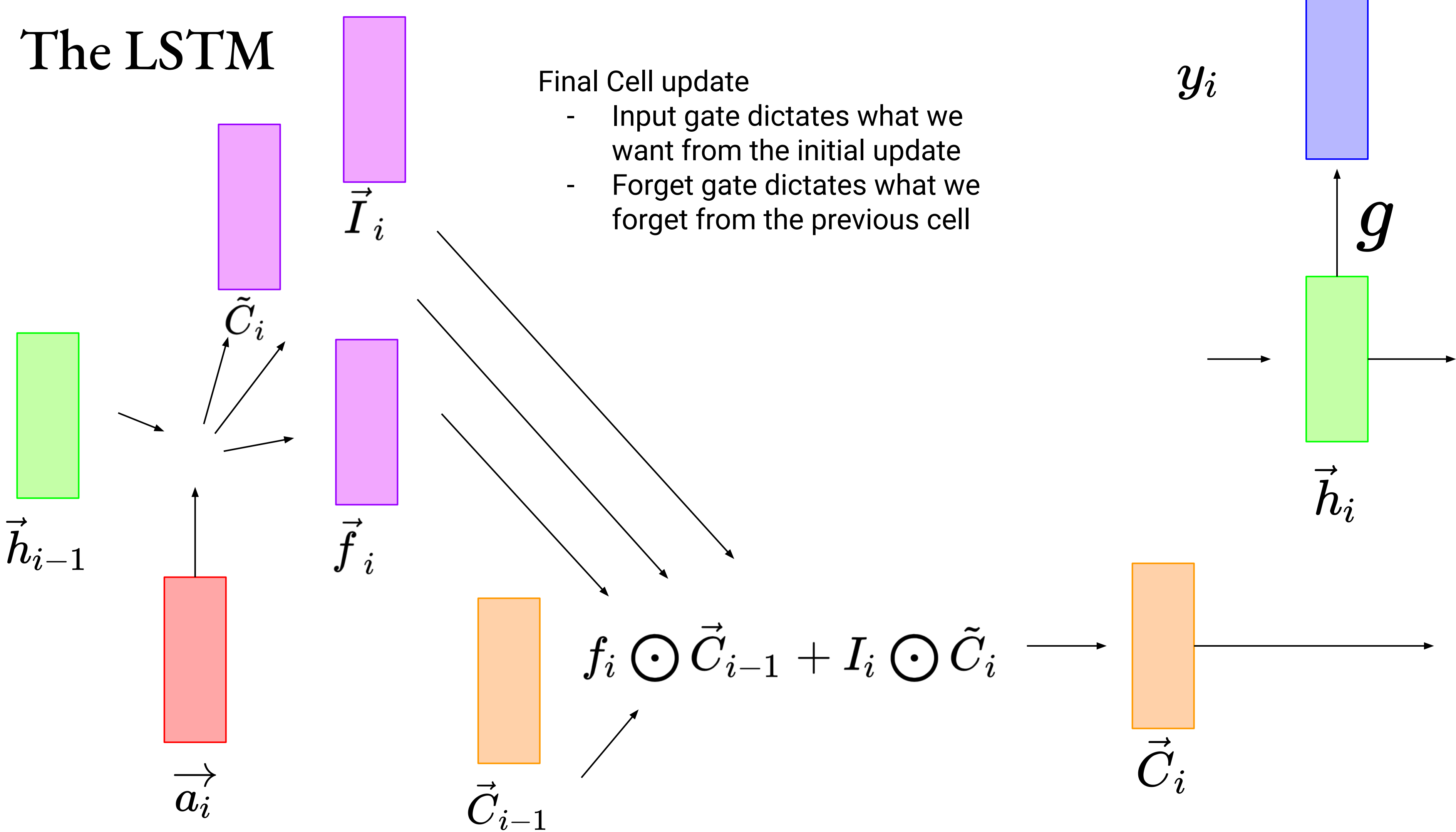
"Input" gate

- Which values we update

The LSTM

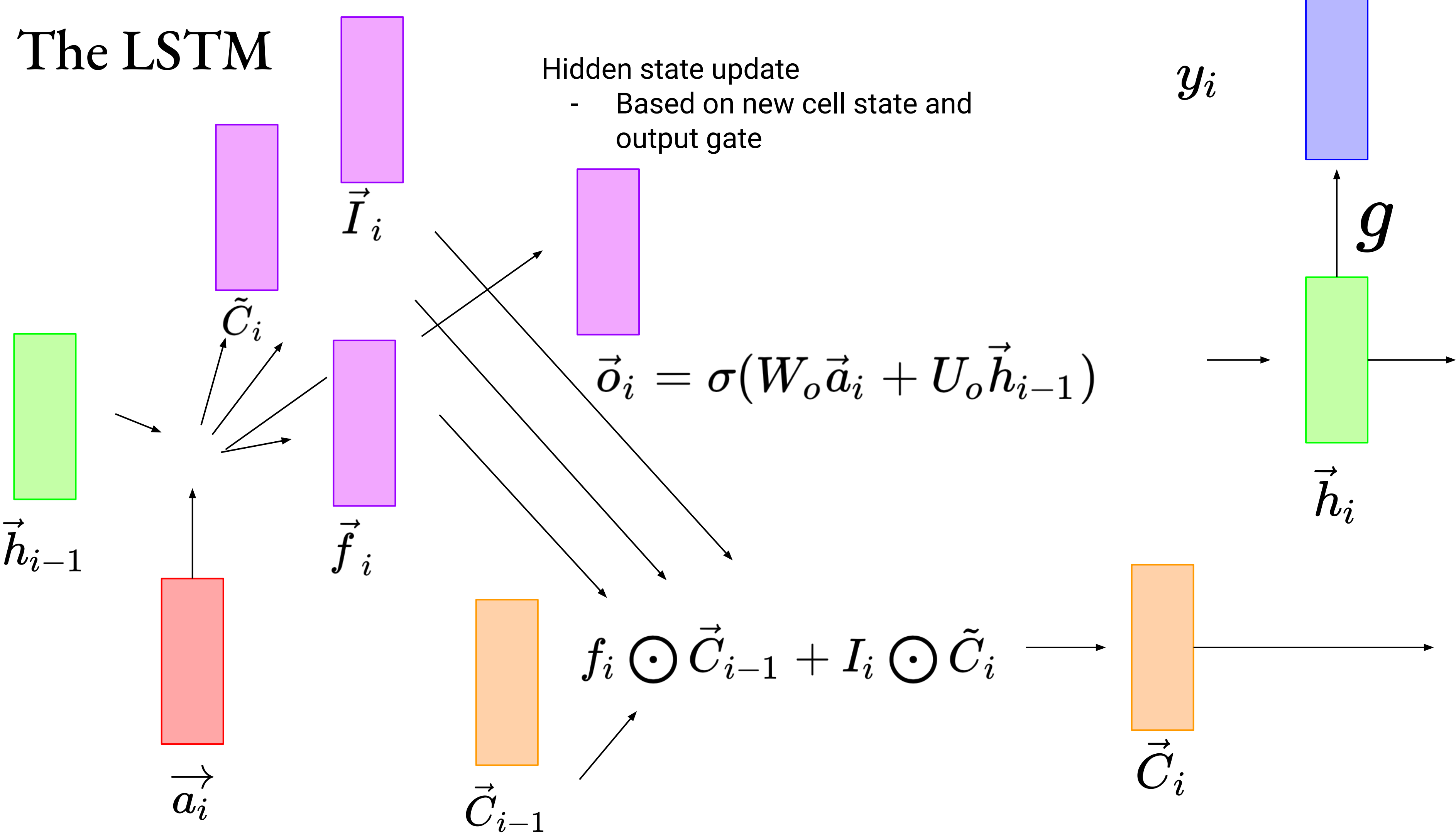
Final Cell update

- Input gate dictates what we want from the initial update
- Forget gate dictates what we forget from the previous cell



The LSTM

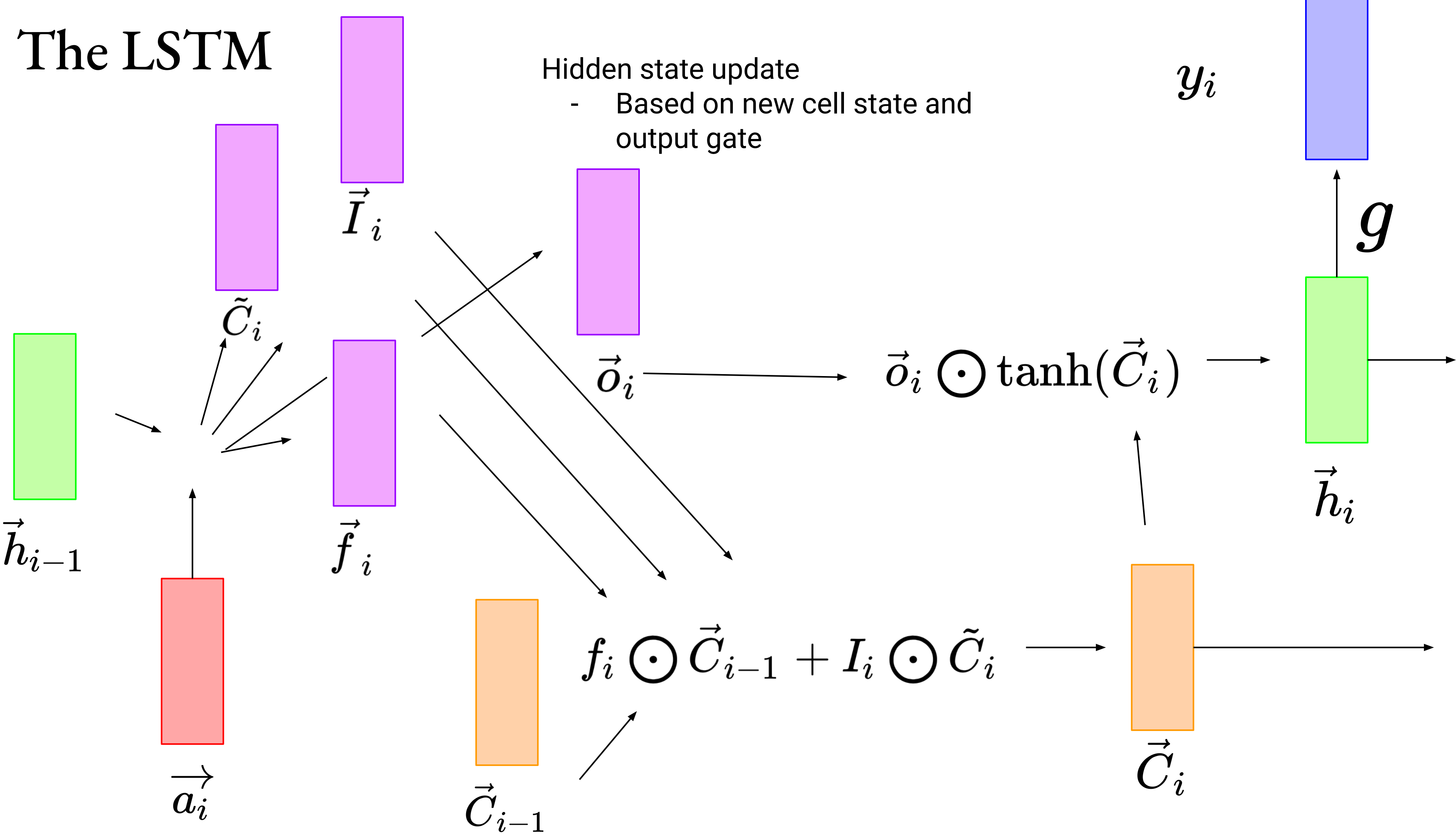
Hidden state update
- Based on new cell state and output gate

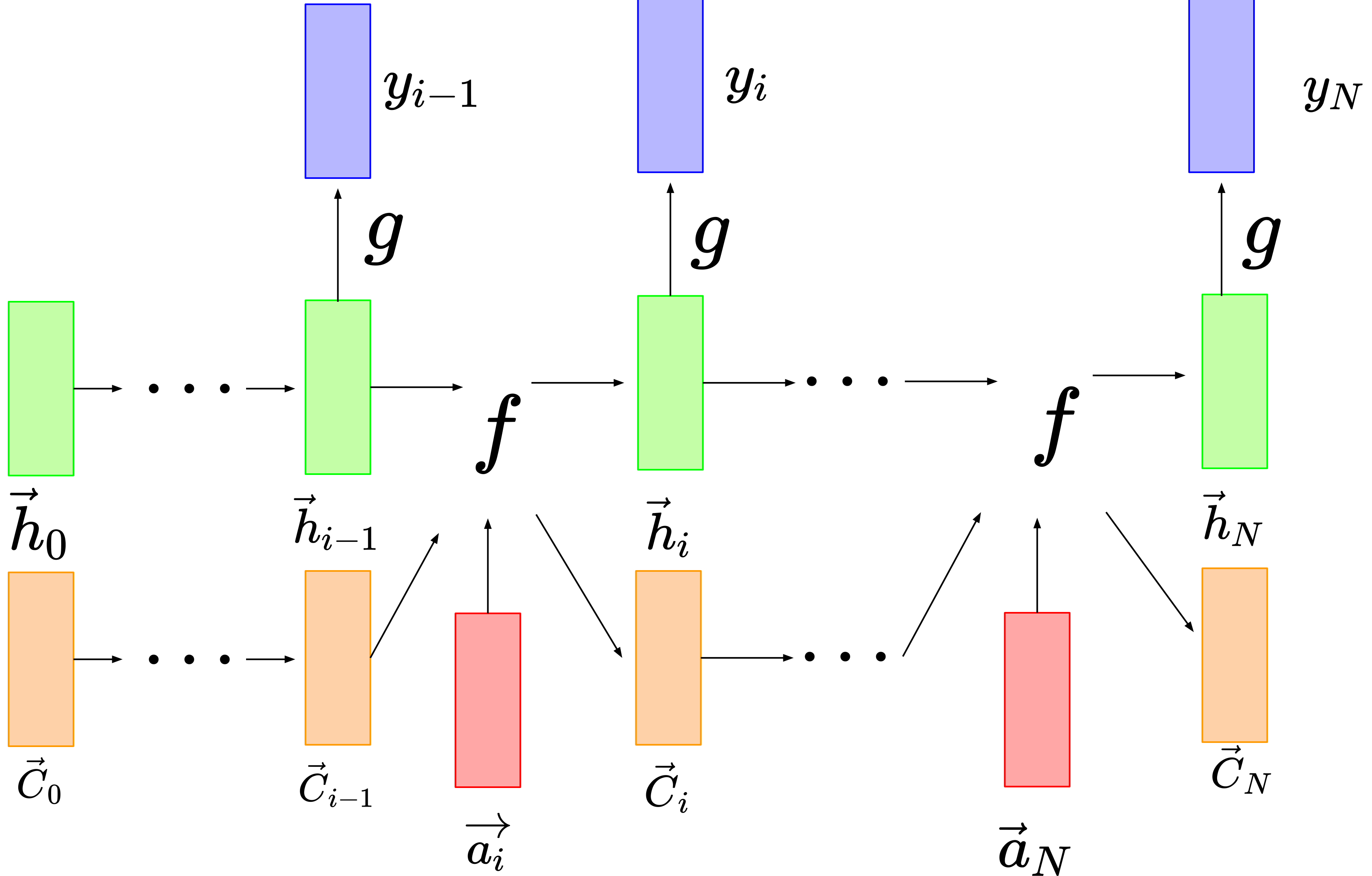


The LSTM

Hidden state update

- Based on new cell state and output gate




y_{i-1}
y_i
y_N
g
g
g
f
f
$\vec{h}_0$
$\vec{h}_{i-1}$
$\vec{h}_i$
$\vec{h}_N$
$\vec{C}_0$
$\vec{C}_{i-1}$
$\vec{C}_i$
$\vec{C}_N$
$\overrightarrow{a_i}$
$\vec{a}_N$

RNNs

- Different styles based on desired inputs/outputs

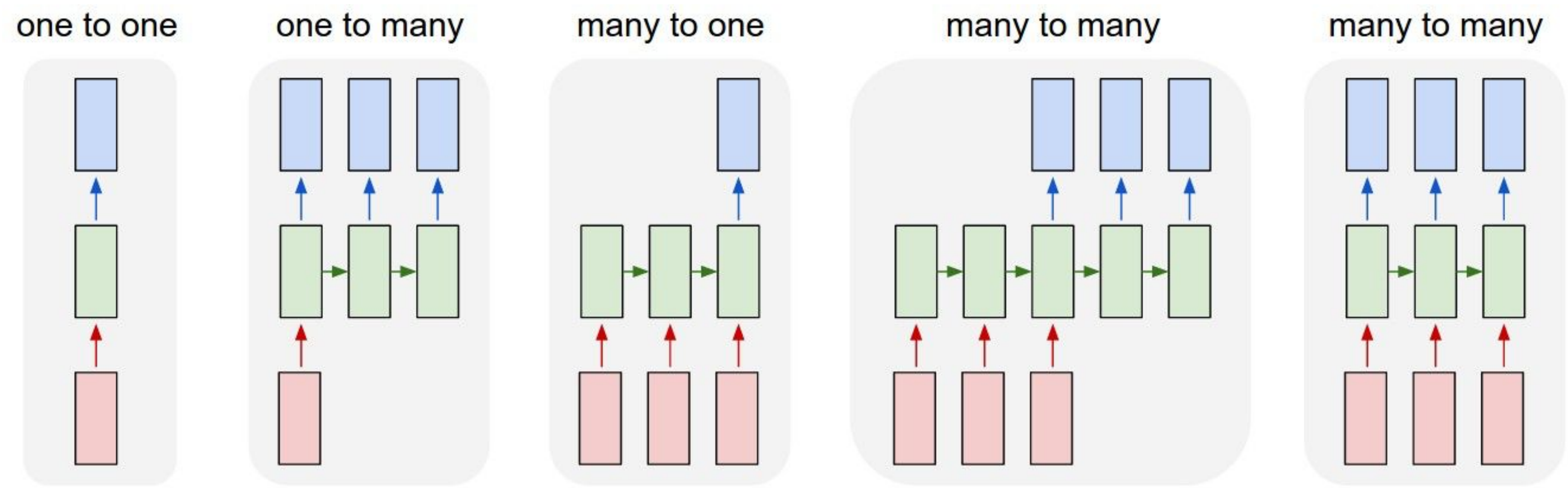


Image: Andrej Karpathy

Questions?

- From last lecture?
- From the lab assignment?

RNNs

- Different styles based on desired inputs/outputs

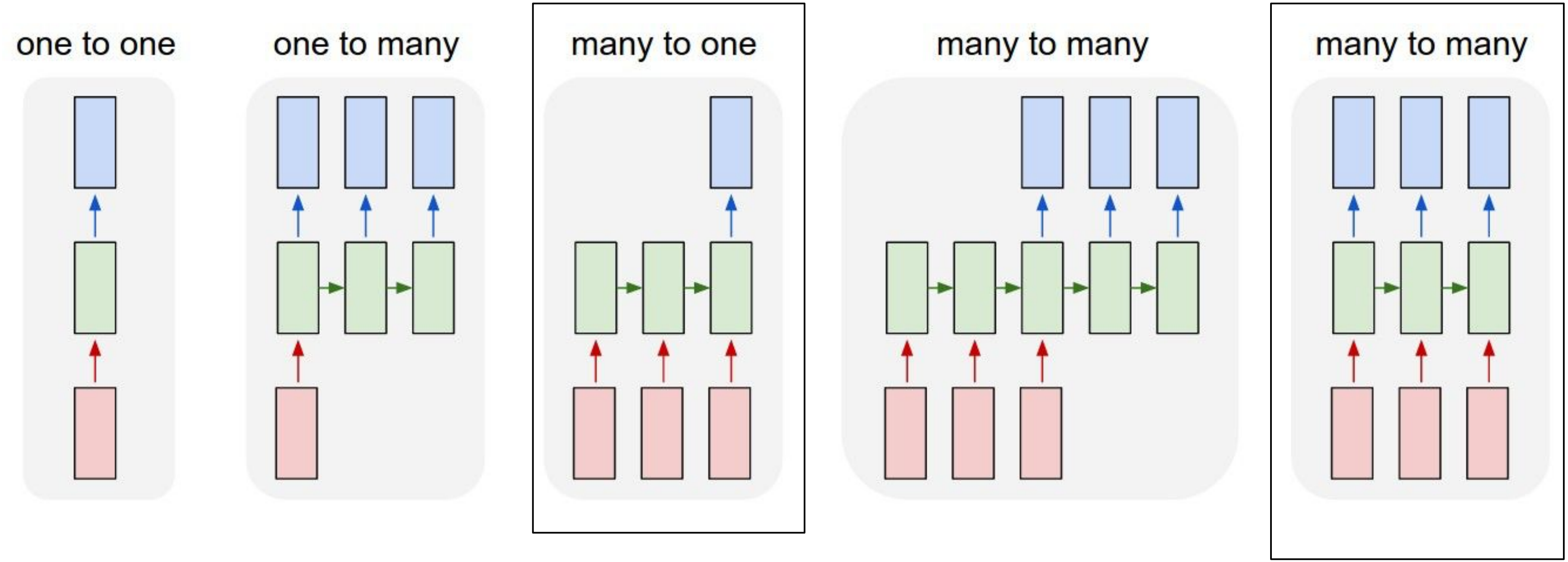
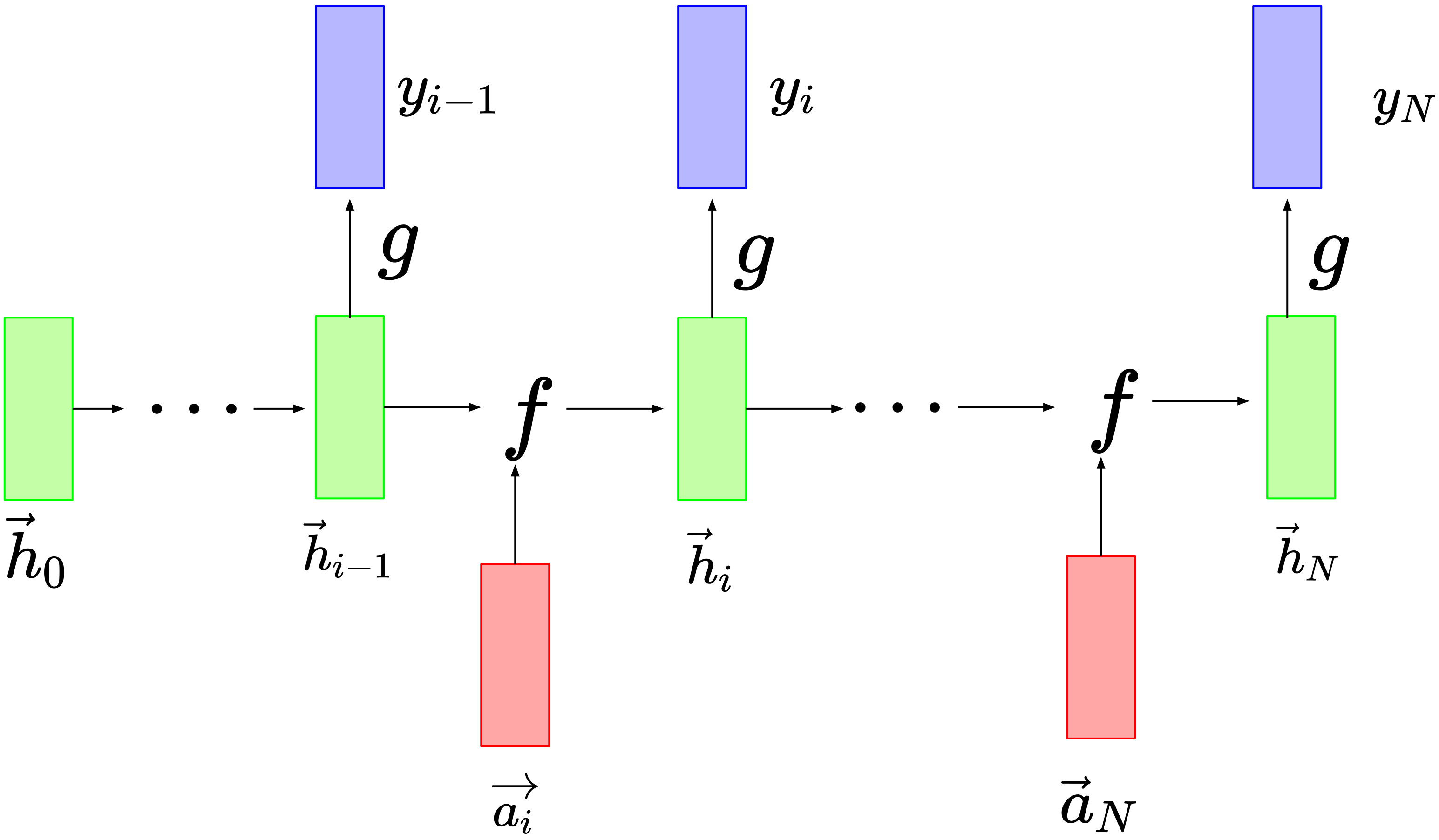
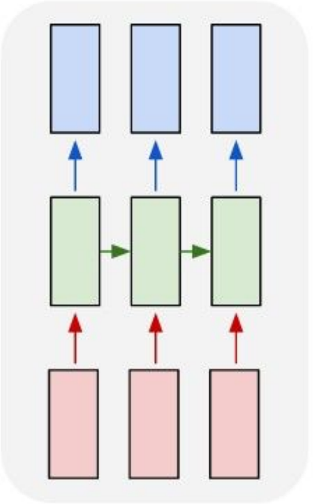


Image: [Andrej Karpathy]

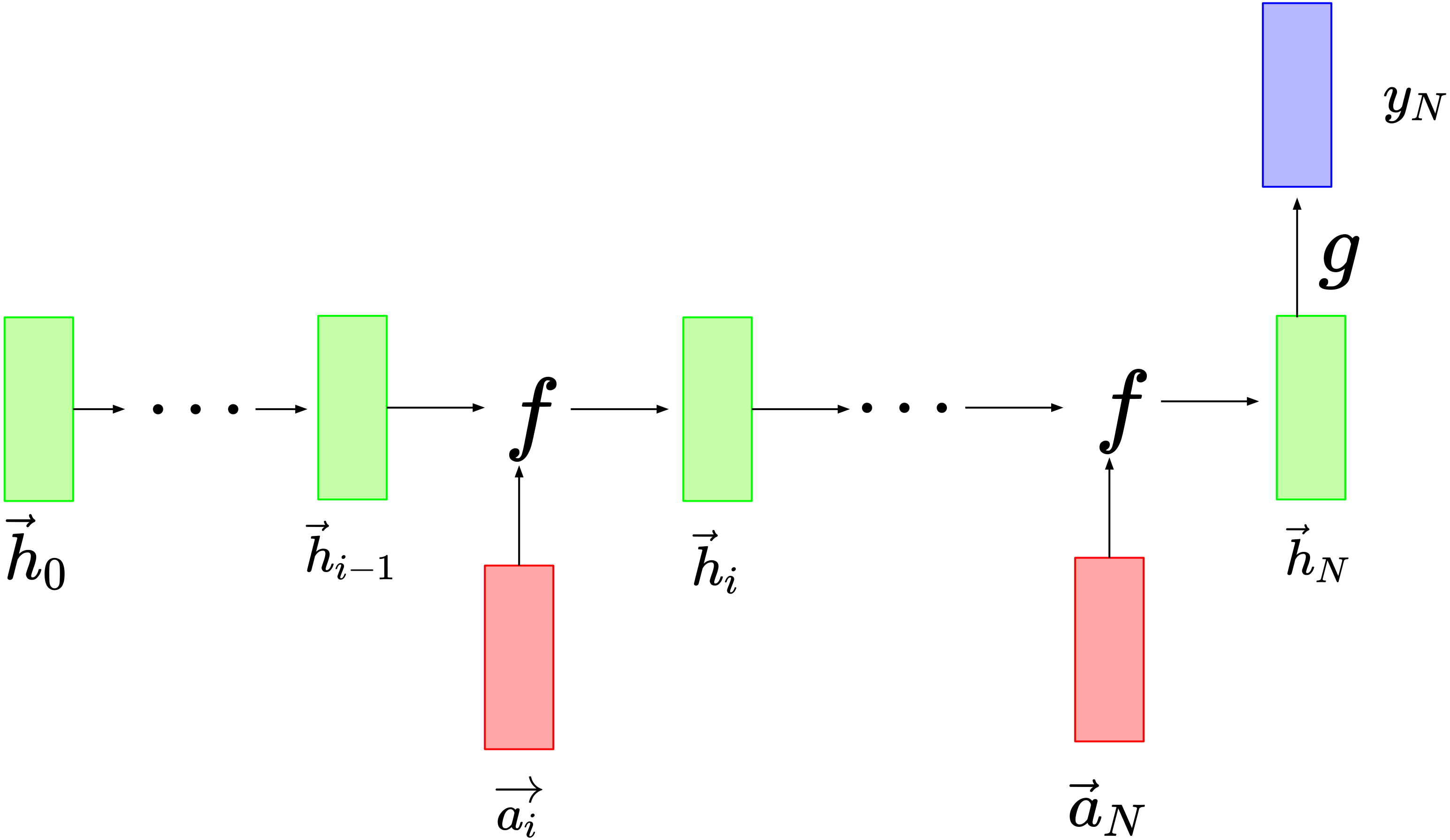
RNNs



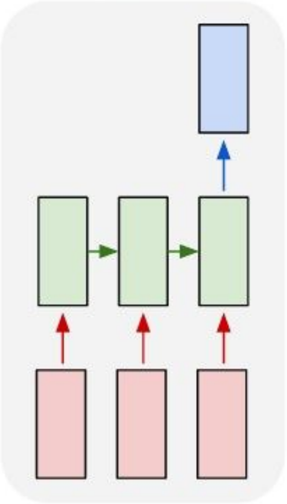
many to many



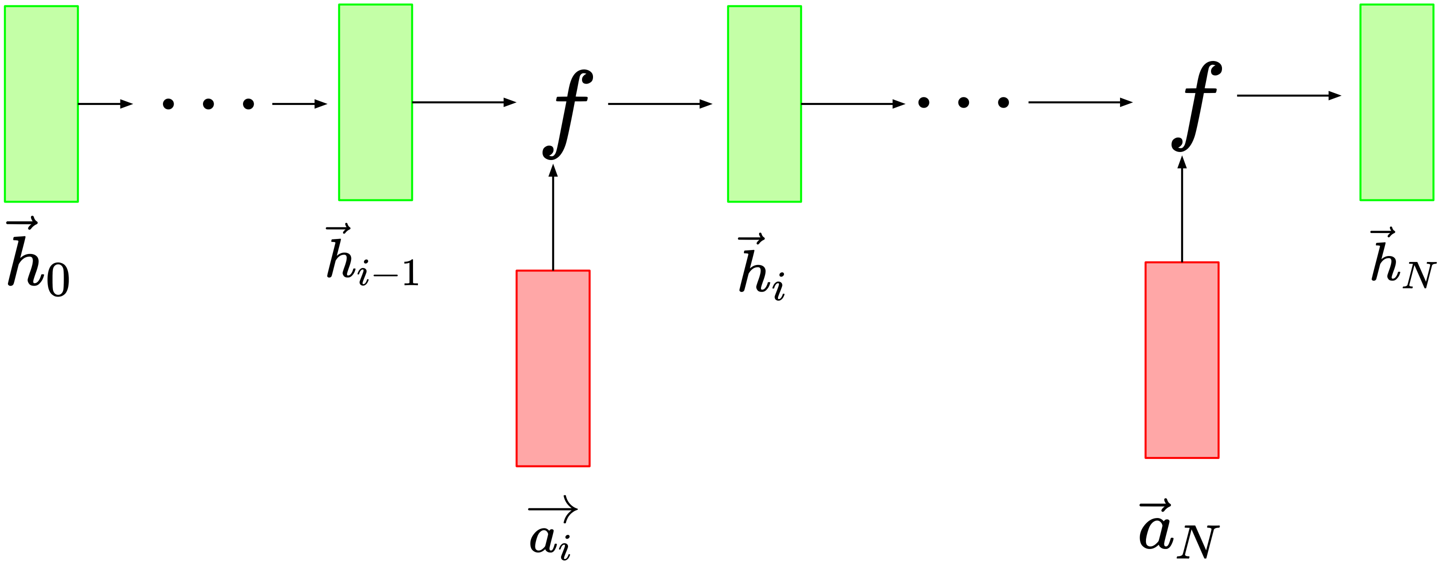
RNNs



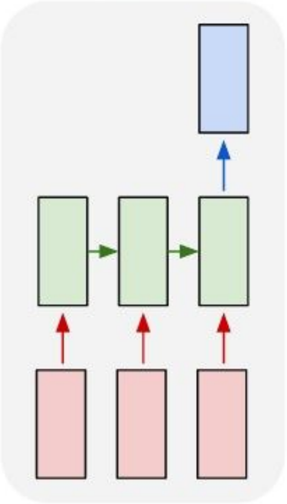
many to one



RNNs

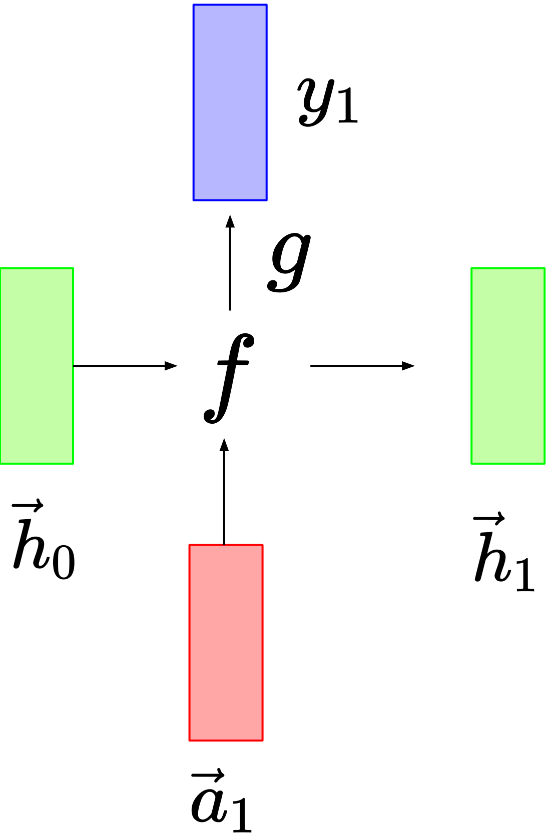


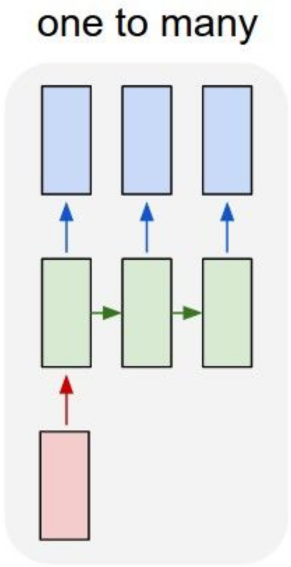
many to one



RNNs

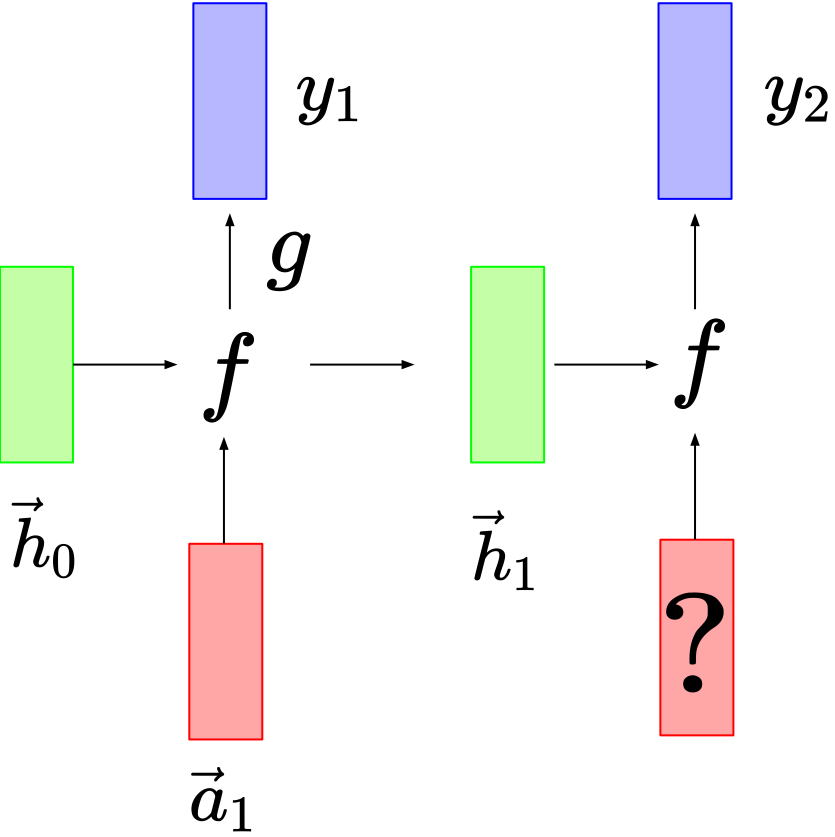
- Generate a sequence from a single input

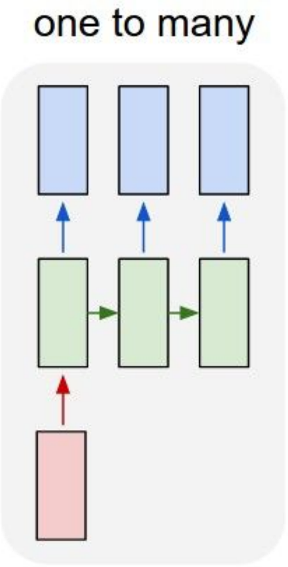




RNNs

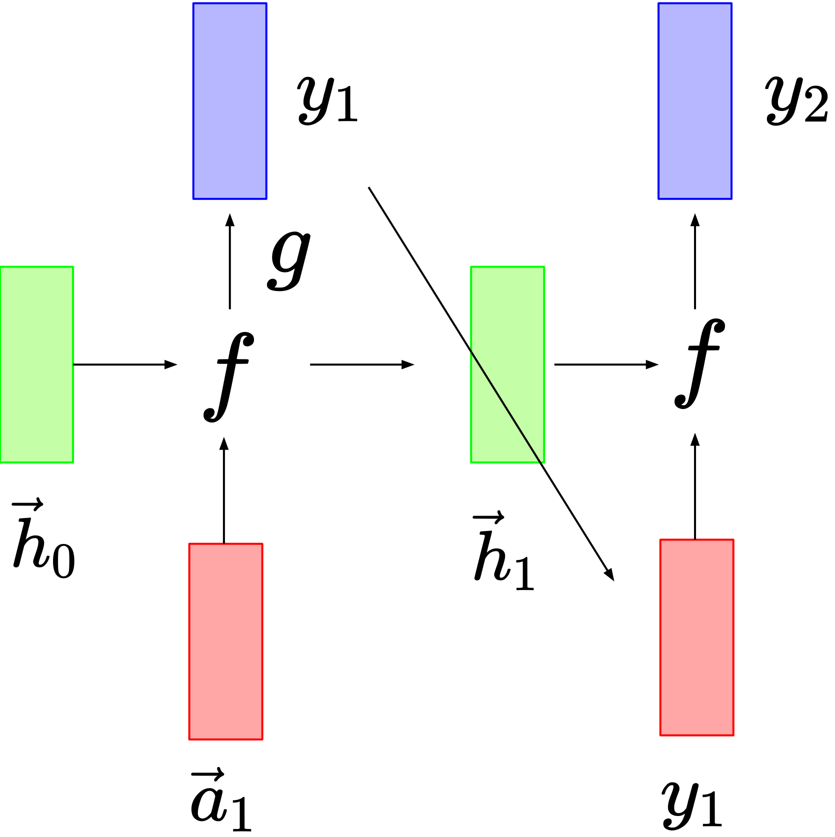
- Generate a sequence from a single input

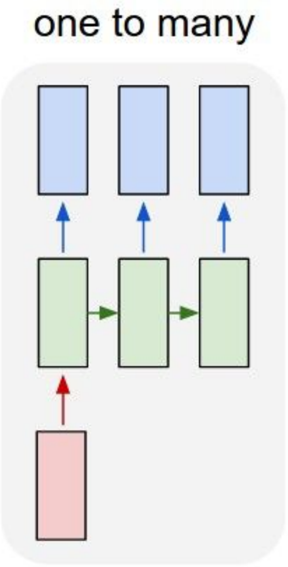




RNNs

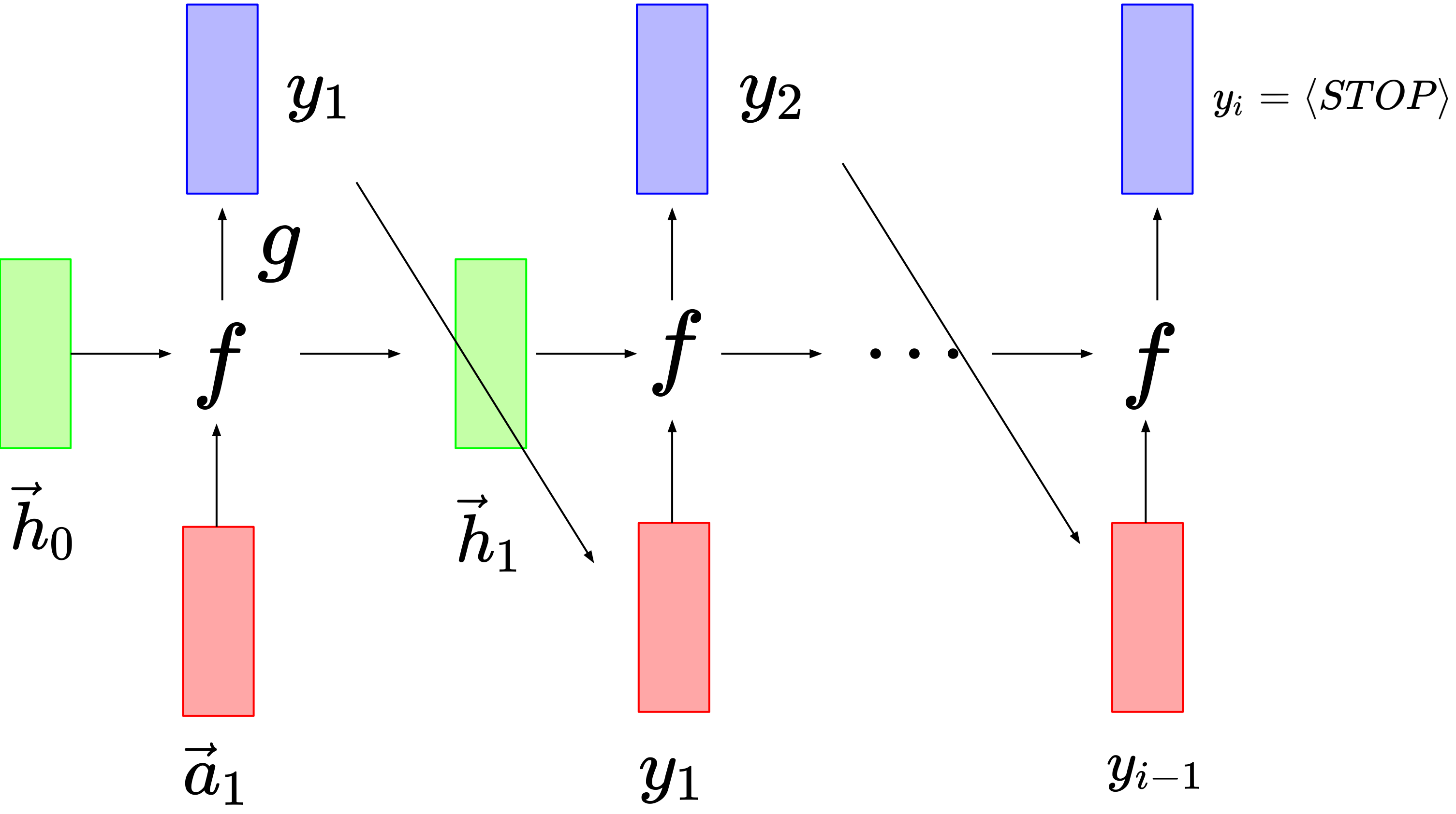
- Generate a sequence from a single input

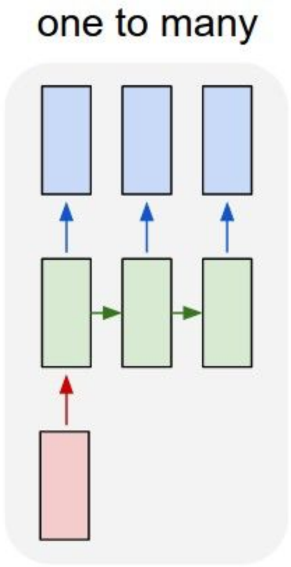




RNNs

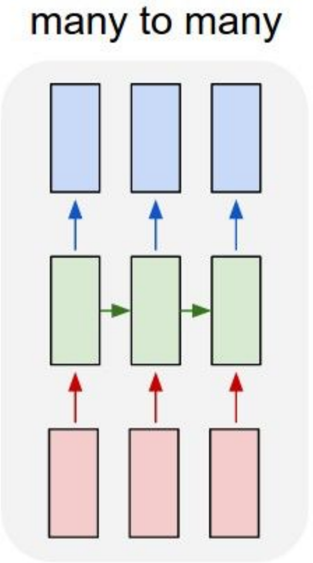
- Generate a sequence from a single input





Problems with this set-up

- Late parts of input sequence don’t inform early predictions

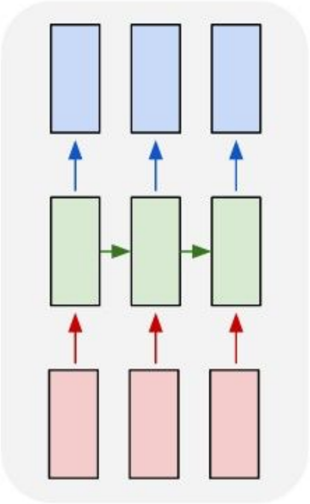


Problems with this set-up

- Late parts of input sequence don’t inform early predictions
- Problem in translation

Ich muss auf den Markt gehen. → I must go to the Market

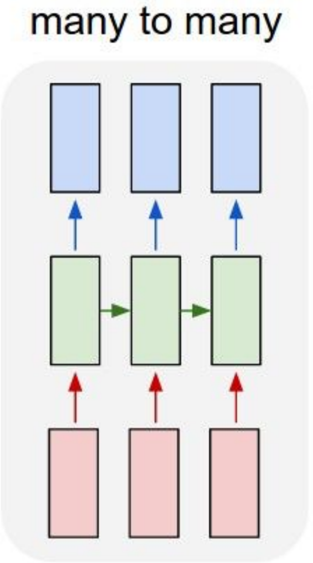
many to many



Problems with this set-up

- Late parts of input sequence don’t inform early predictions
- Problem in translation

Ich muss auf den Markt gehen. → I must go to the Market



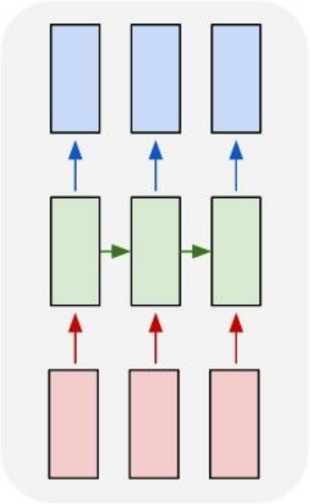
Problems with this set-up

- Late parts of input sequence don't inform early predictions
- Problem in translation

Ich muss auf den Markt gehen. ⟶ I must go to the Market

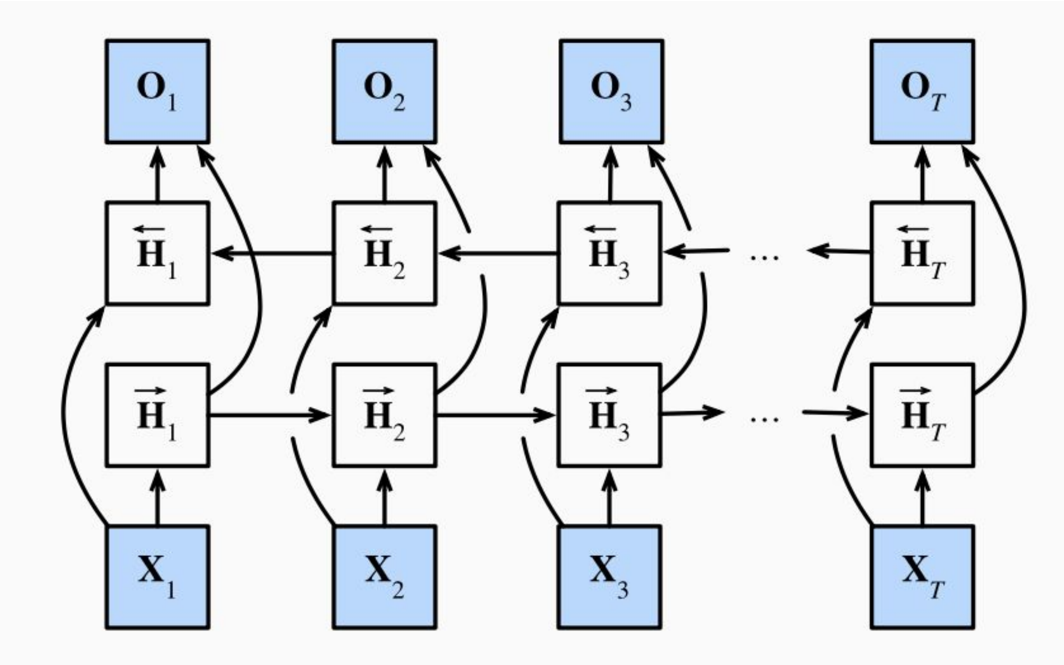
?

many to many



Problems with this set-up

- Late parts of input sequence don't inform early predictions
- Problem in translation
- Bidirectional RNN



many to many

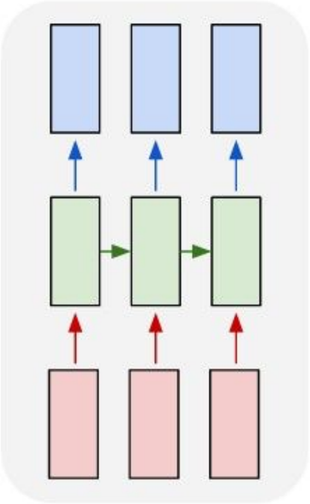
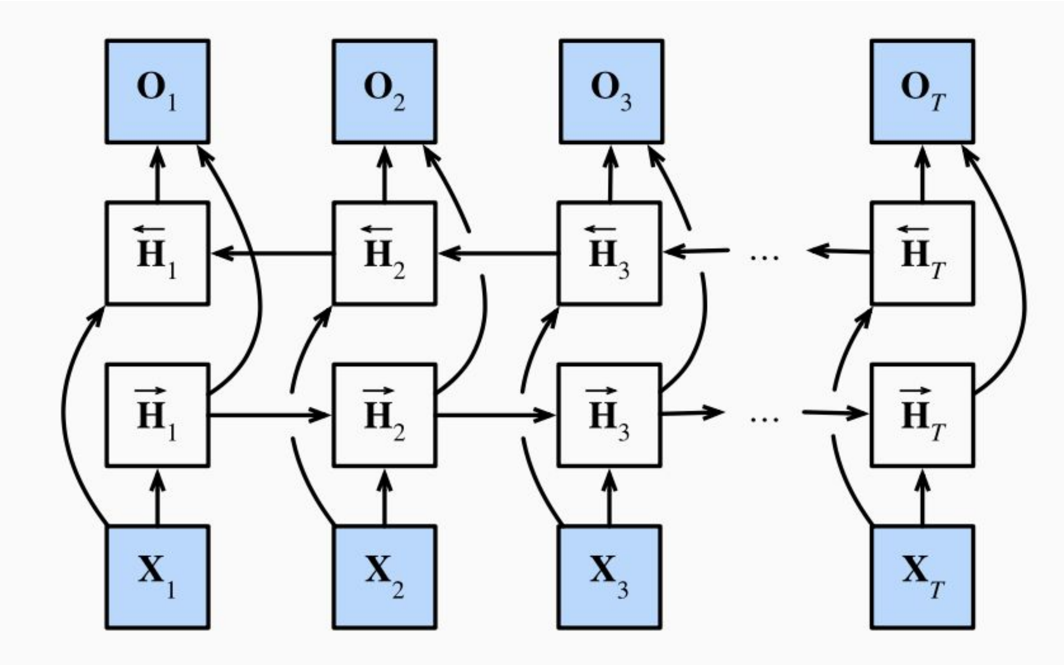


Image source

Problems with this set-up

- Late parts of input sequence don’t inform early predictions
- Problem in translation
- Bidirectional RNN
  - SLOW



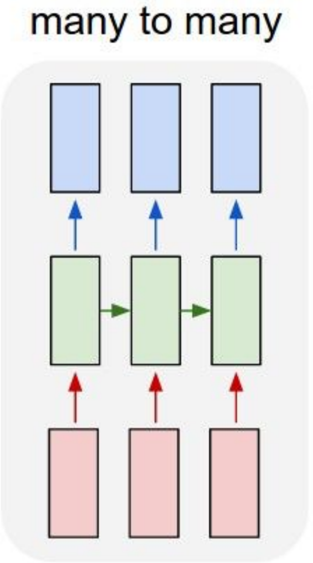
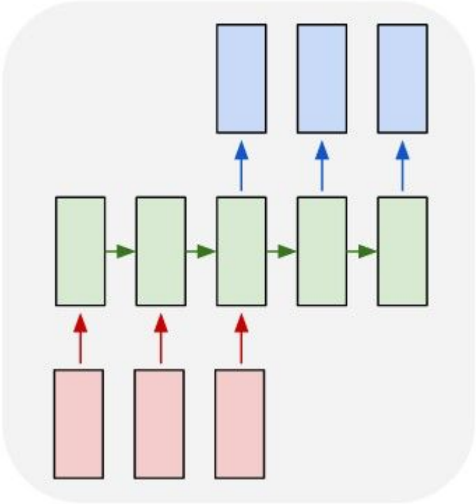


Image source

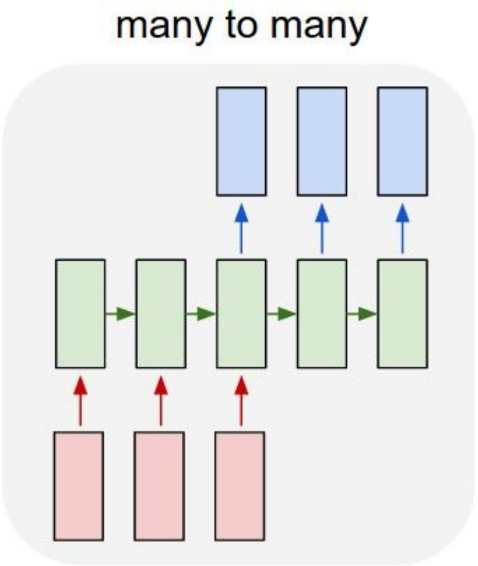
Seq2Seq Models

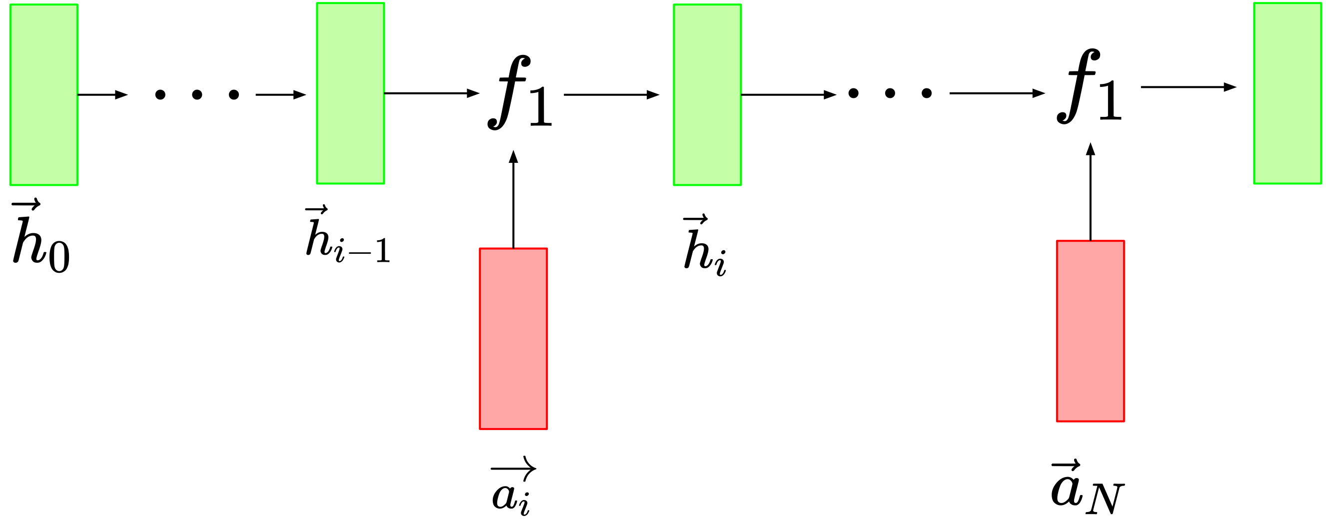
- Generate a sequence using encoder/decoder framework
- Idea: Different RNNs for encoding vs. decoding



Seq2Seq Models

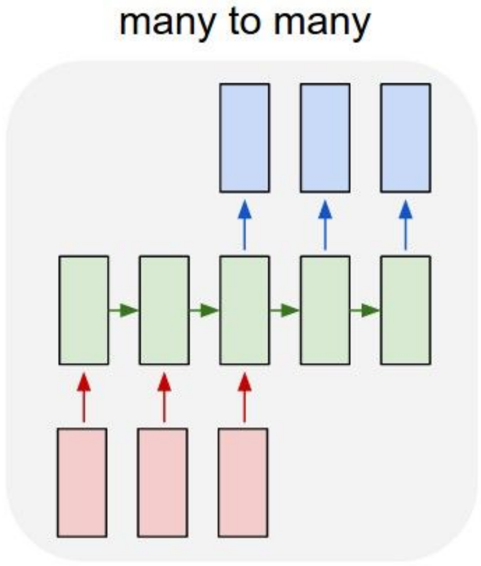
- Generate a sequence using encoder/decoder framework
- Idea: Different RNNs for encoding vs. decoding

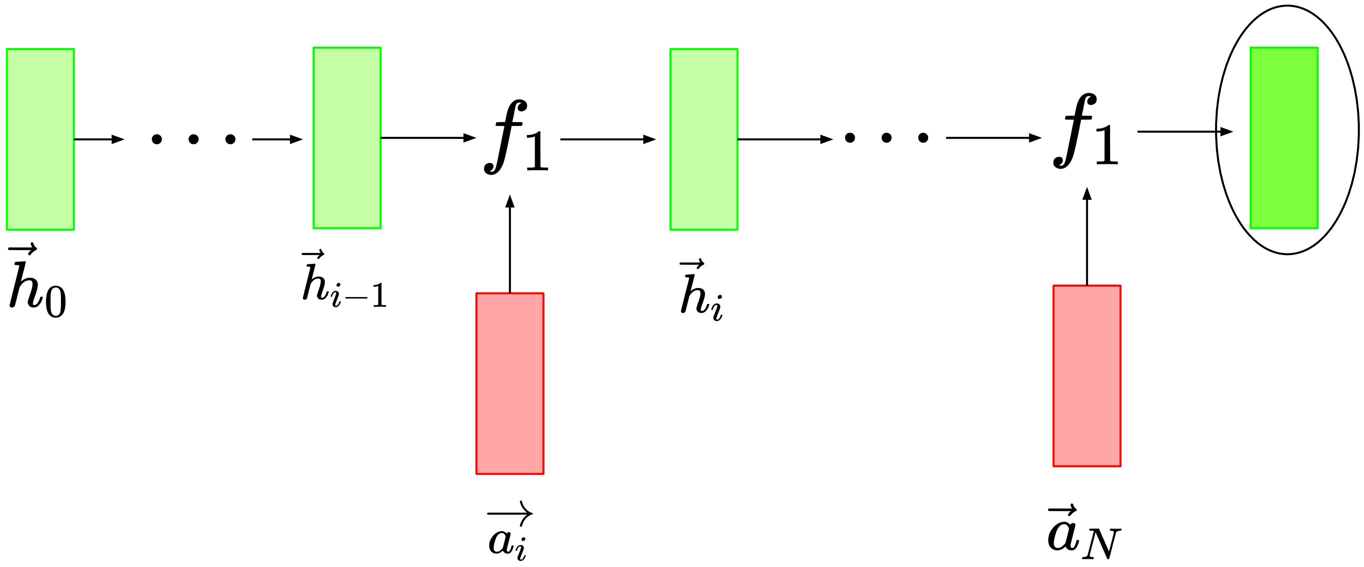




Seq2Seq Models

- Generate a sequence using encoder/decoder framework
- Idea: Different RNNs for encoding vs. decoding

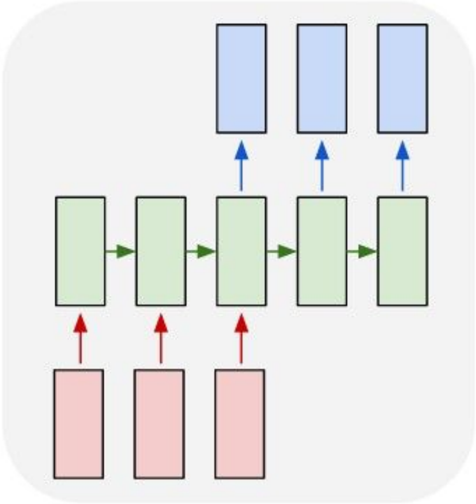


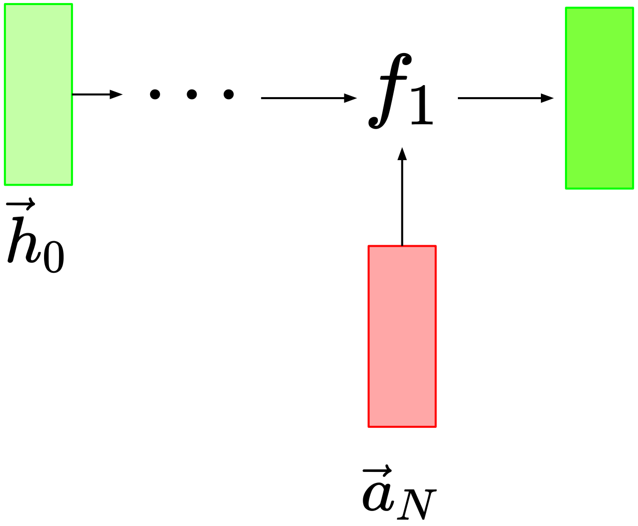


Seq2Seq Models

- Generate a sequence using encoder/decoder framework
- Idea: Different RNNs for encoding vs. decoding

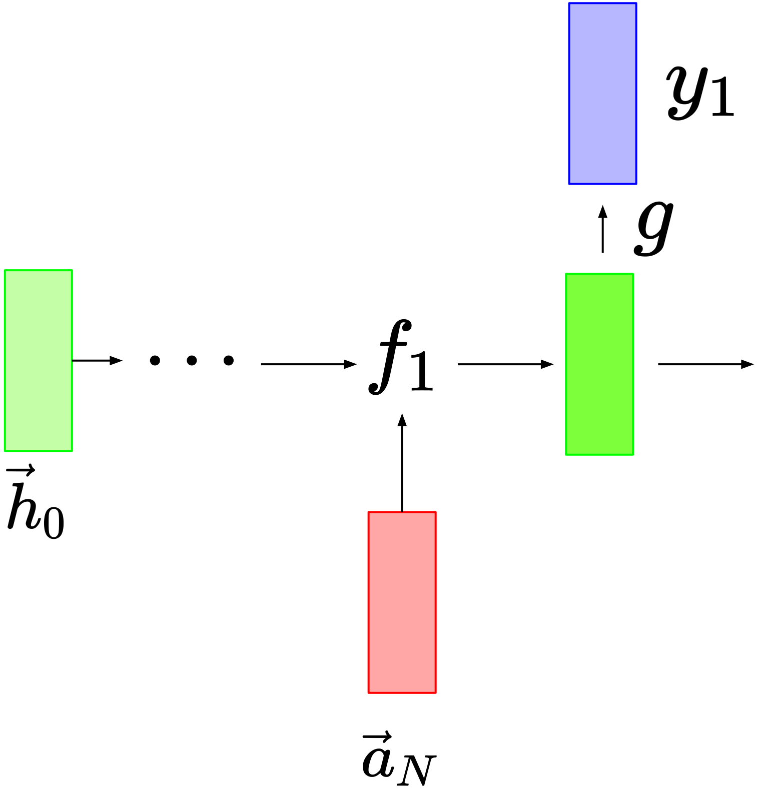
many to many

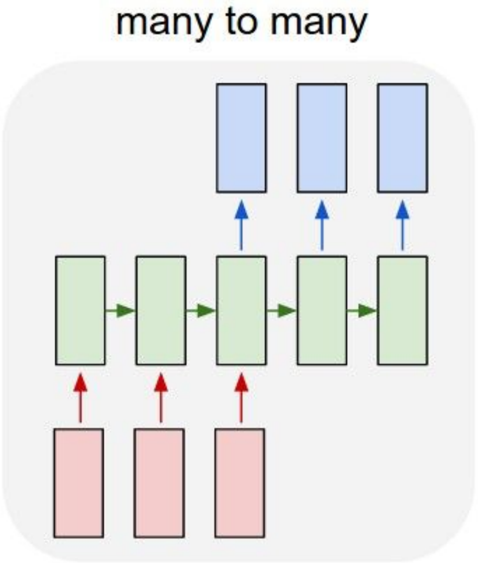




Seq2Seq Models

- Generate a sequence using encoder/decoder framework
- Idea: Different RNNs for encoding vs. decoding





Seq2Seq Models

- Generate a sequence using encoder/decoder framework
- Idea: Different RNNs for encoding vs. decoding