Neural Language Generation

Yoav Goldberg

INLG 2018

B I U
N L P

Bar-Ilan University

ALLEN INSTITUTE for ARTIFICIAL INTELLIGENCE

NLG and me

My PhD supervisor

Michael Elhadad

NLG and me

His PhD supervisor

Kathleen McKeown

My PhD supervisor

Michael Elhadad

NLG and me

His PhD supervisor

Kathleen McKeown

My PhD supervisor

Michael Elhadad

His other PhD students

Regina Barzilay

- Lexical chains
- Statistical NLG
- Coherence

....

NLG and me

His PhD supervisor

Kathleen McKeown

My PhD supervisor

Michael Elhadad

His other PhD students

Regina Barzilay

Yael Netzer

- Generation in Hebrew (determiners and quantifiers, noun-compounds)
- Generation for assistive tech
- **Organizer of INLG 2000**

....

NLG and me

His PhD supervisor

Kathleen McKeown

My PhD supervisor

Michael Elhadad

I'll do my PhD on syntactic parsing

His other PhD students

Regina Barzilay

Yael Netzer

Yoav Goldberg

Linguistic Creativity Workshop, 2009

Gaiku : Generating Haiku with Word Associations Norms

Yael Netzer* and **David Gabay** and **Yoav Goldberg**† and **Michael Elhadad**
Ben Gurion University of the Negev
Department of Computer Science
POB 653 Be'er Sheva, 84105, Israel
{yaeln,gabayd,yoavg,elhadad}@cs.bgu.ac.il

Michael Elhadad

David Gabay

Yael Netzer

Yoav Goldberg

NLG and me

- I am **NOT** an expert on NLG
- I **MAY** be considered an expert on "neural" NLP methods
- I **sometimes** say controversial things
- I **know enough** about NLG to identify when it is done wrong
- I think neural NLG methods are doing most things wrong

Who to invite to give a tutorial next year?

Mirella Lapata **Alexander Rush**

?

This tutorial

- How to use neural methods for generating text.
  - ...while somewhat controlling the resulting output.
- What are the common neural techniques in use today?
    - seq2seq+attention
- Some tools.
- May be trivial for many of you, drinking from the firehose to others.

This tutorial

- The components which I think are useful to know.
- Concepts, not details (sorry, no time).
- No "state of the art" (because there isn't any state-of-the-art).
- Some high-level observations.
- We may not get to the end, so feel free to interfere with questions.

This tutorial

- Part 1: The mechanics
- Part 2: Use cases / examples
- Part 3: Opinions

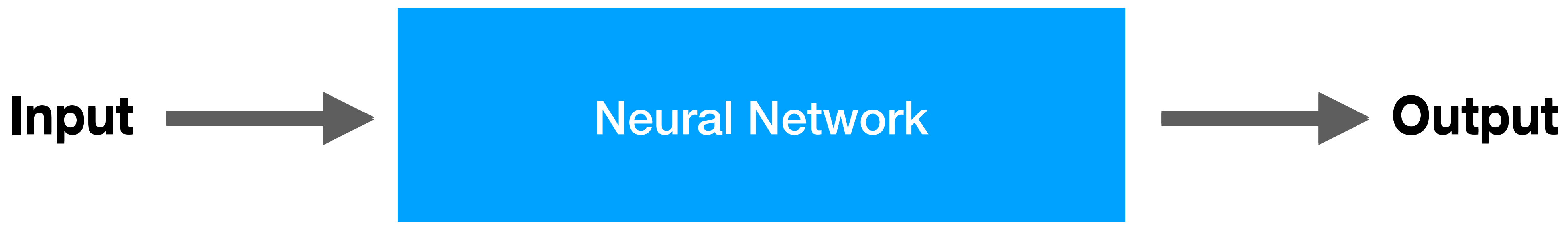
Classic NLG

Input → **Content selection**
Macro-planning
Packaging
Micro-planning
Referring Expression Generation
Linearization → **Output**

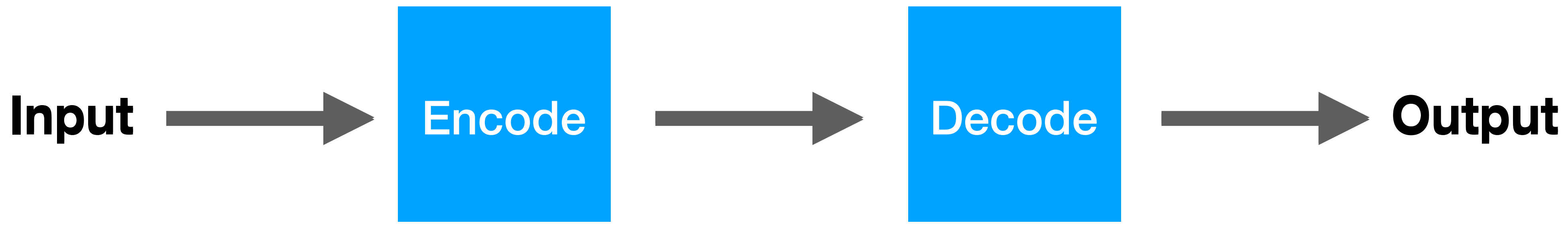
Neural NLG

Input → ~~Content selection~~ ~~Macro-planning~~ ~~Packaging~~ ~~Micro-planning~~ ~~Referring Expression Generation~~ ~~Linearization~~ → Output

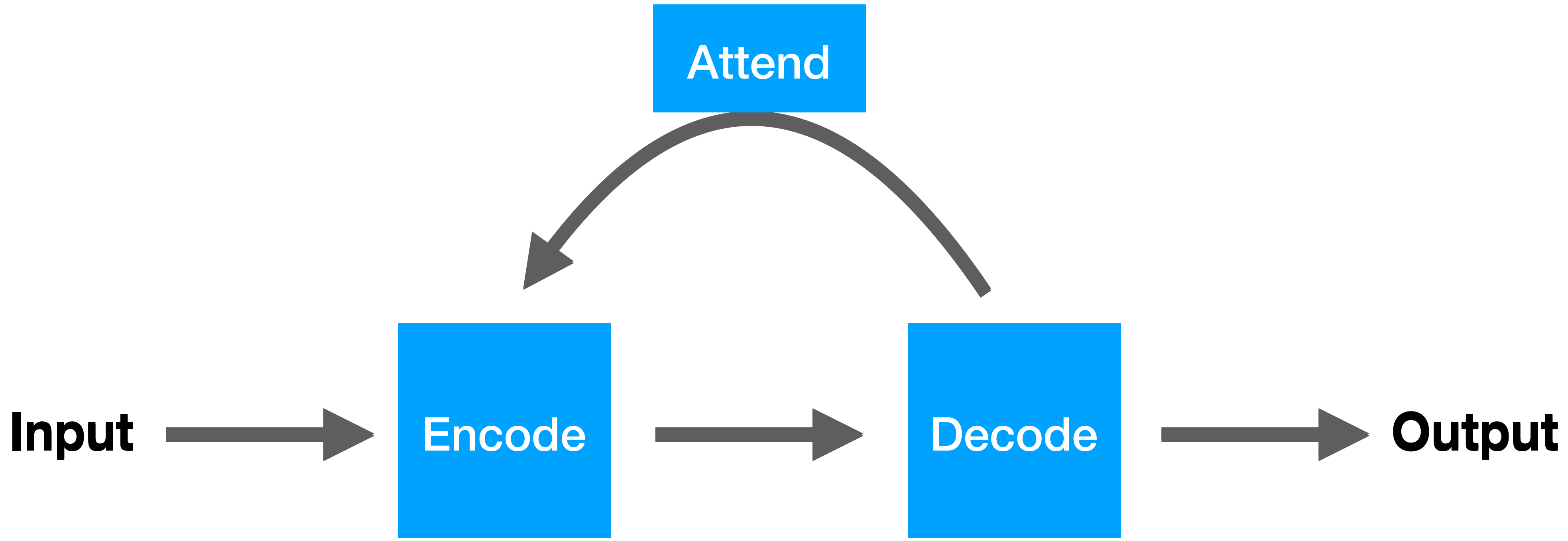
Neural NLG



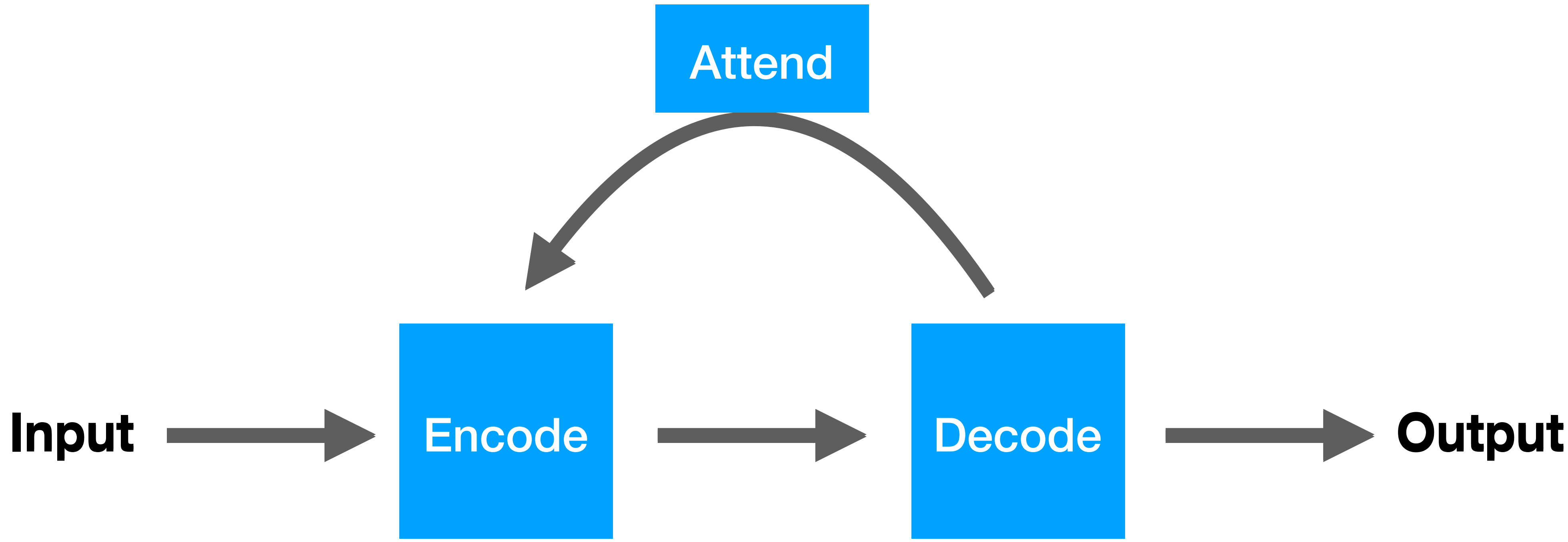
The basic abstraction



The basic abstraction



The basic abstraction

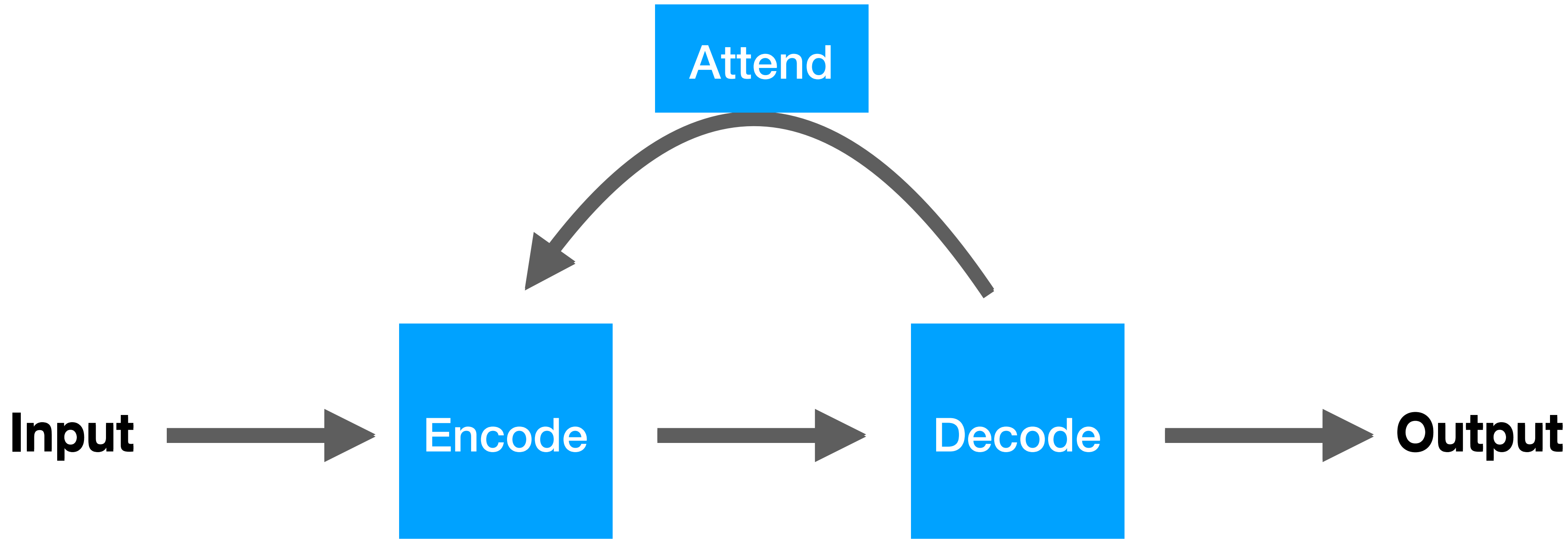


Neural Machine Translation

Neural NLG

- Neural networks are great at learning to map inputs to outputs based on examples.
- They are surprisingly effective at discovering regularities.
- They need many training examples.
- They are somewhat hard to control.

The basic abstraction



Neural Machine Translation

sequence to sequence text generation

(neural machine translation)

Language Model

- How to assign a probability to a sentence.
  - p(`I read a book about dogs`)
- another view: distribution over next word:
  - p(`dogs | I read a book about _____`)

Language Model

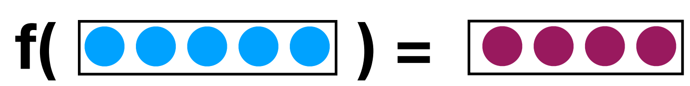
- **Can generate from a trained language model.**
- Probability of first word given empty sentence.
  - Sample the first word.
- Probability of second word given first word.
  - Sample the second word.
- Probability of third word given first two words.
  - Sample the third word.
- ....

Language Model

- `gen_so_far` = ["<s>"]
- while True:
  - `next_word_distribution` = p(`next` | `gen_so_far`)
  - sample `next_word` from `next_word_distribution`
  - if `next_word` == "</s>": break
  - `gen_so_far`.append(`next_word`)

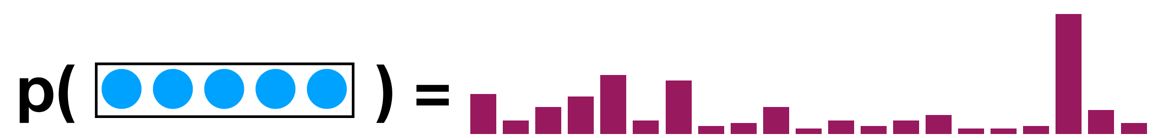
Let's build a neural LM

Neural Networks



functions from vectors to vectors

Neural Networks



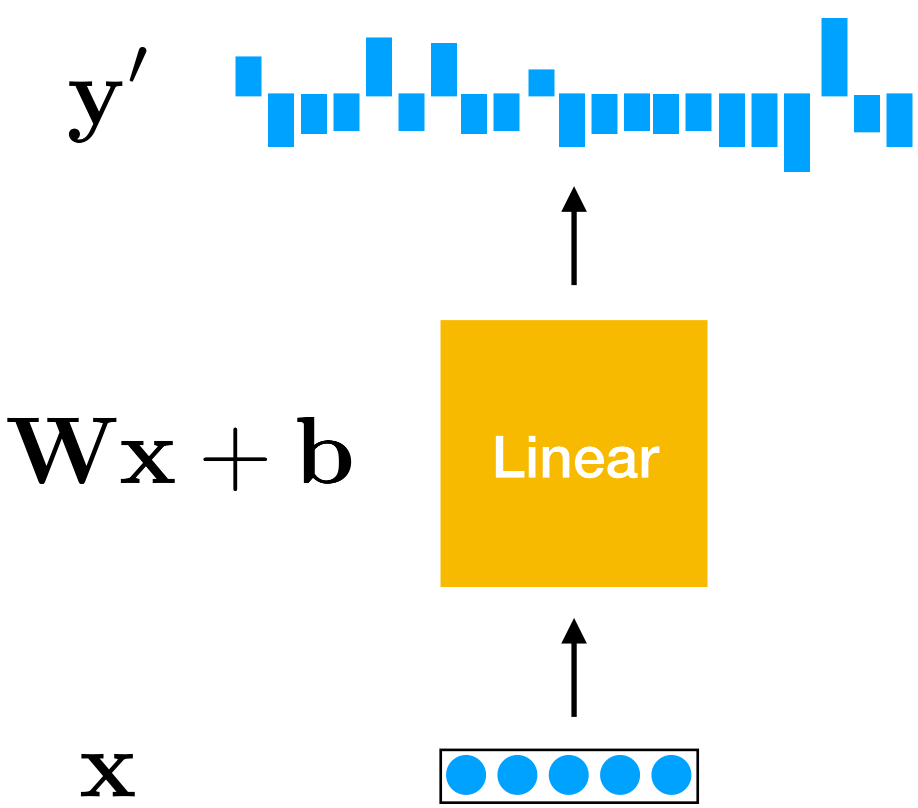
functions from vectors
to probabilities

(these are still functions from vectors to vectors)

Predicting from a vector

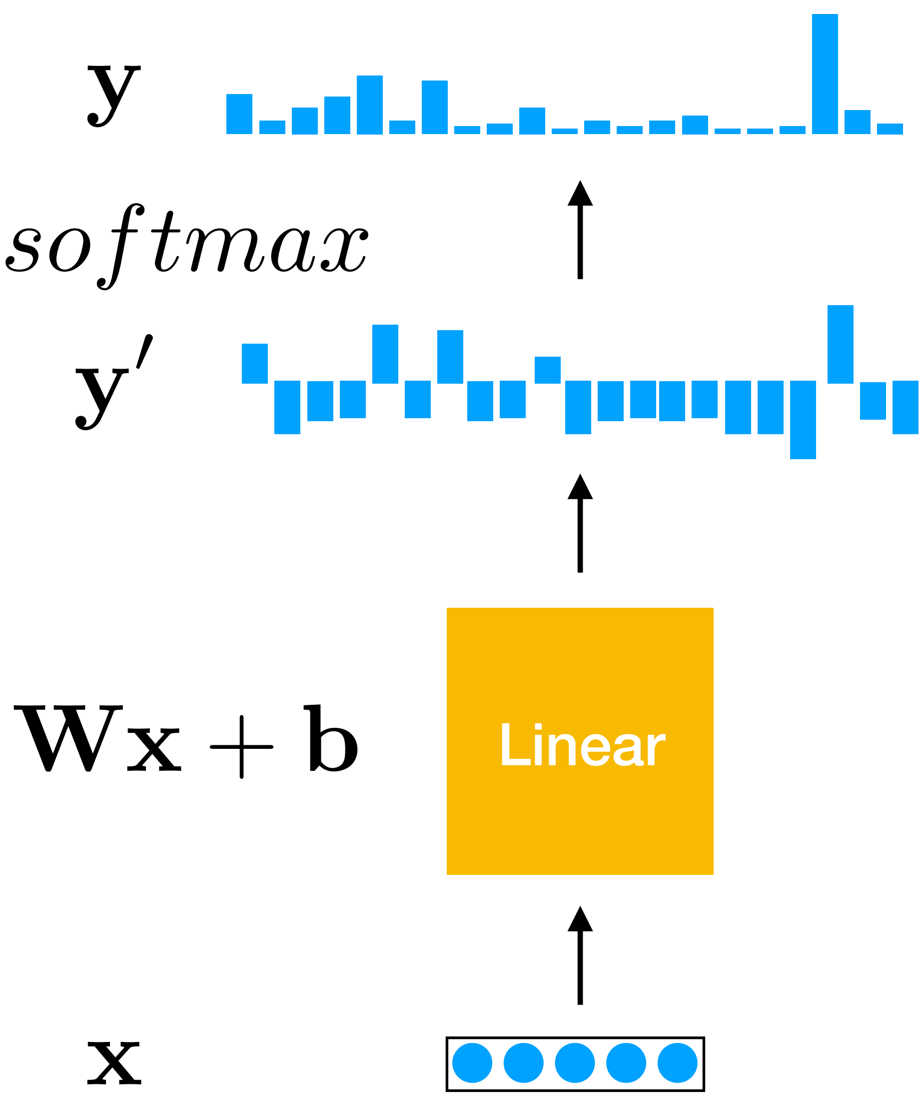
Predict from a vector (Linear Layer + softmax)

$$p(y = ?|\mathbf{x})$$



$$predict(\mathbf{x}) = \mathbf{W}\mathbf{x} + \mathbf{b}$$

Predict from a vector (Linear Layer + softmax)



$$p(y=?|\mathbf{x})$$

$$predict(\mathbf{x}) = softmax(\mathbf{W}\mathbf{x}+\mathbf{b})$$

$$softmax(\mathbf{x})_{[i]} = \frac{e^{\mathbf{x}_{[i]}}}{\sum_j e^{\mathbf{x}_{[j]}}}$$

Predict from a vector (Linear Layer + softmax)

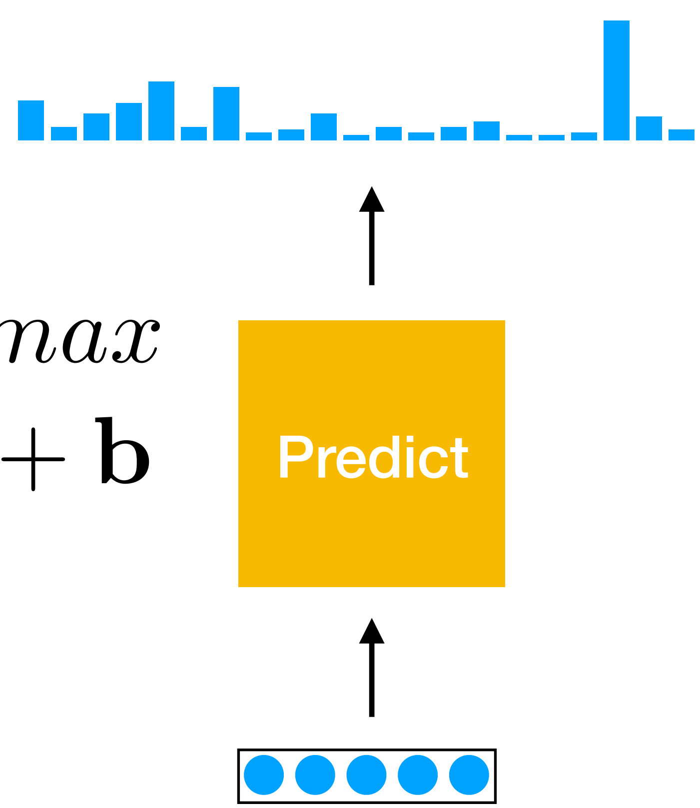
$$p(y = ? | \mathbf{x})$$

$\mathbf{y}$

$softmax$

$\mathbf{W}\mathbf{x} + \mathbf{b}$

$\mathbf{x}$



$$predict(\mathbf{x}) = softmax(\mathbf{W}\mathbf{x} + \mathbf{b})$$

$$softmax(\mathbf{x})_{[i]} = \frac{e^{\mathbf{x}_{[i]}}}{\sum_j e^{\mathbf{x}_{[j]}}}$$

Predict from a vector (Linear Layer + softmax)

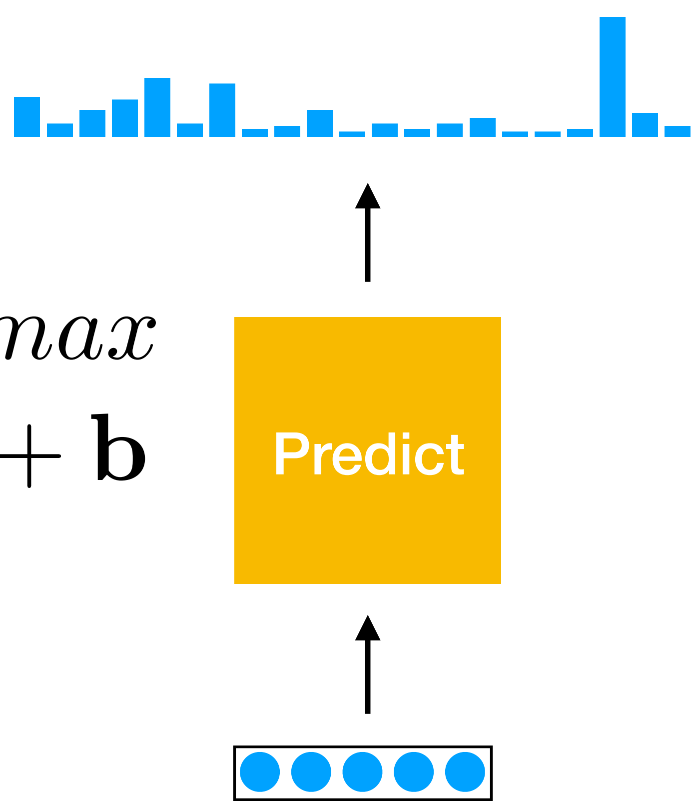
$$p(y=?|\mathbf{x})$$

$\mathbf{y}$

$softmax$

$\mathbf{W}\mathbf{x}+\mathbf{b}$

$\mathbf{x}$



$$predict(\mathbf{x}) = softmax(\mathbf{W}\mathbf{x}+\mathbf{b})$$

$$softmax(\mathbf{x})_{[i]} = \frac{e^{\mathbf{x}_{[i]}}}{\sum_j e^{\mathbf{x}_{[j]}}}$$

Discuss training

Predicting from words

Neural **NLP** Building Blocks

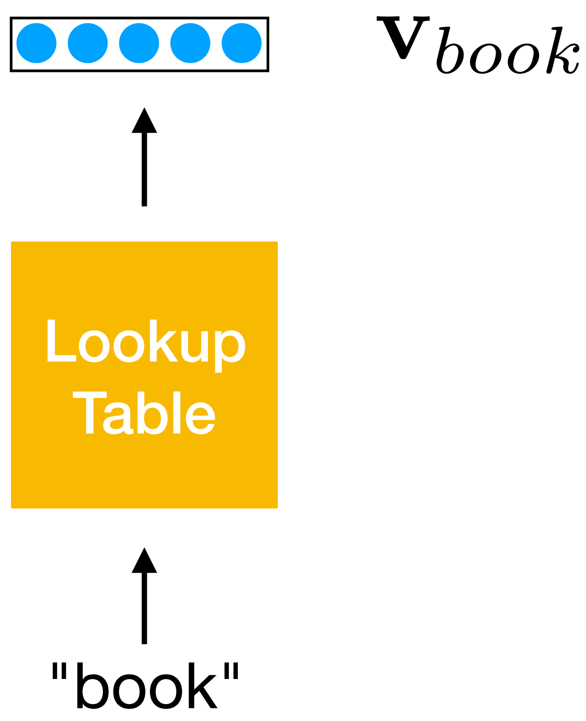
- Word Embeddings: translate a word to a vector.
- Ways of combining vectors.

Word Embeddings

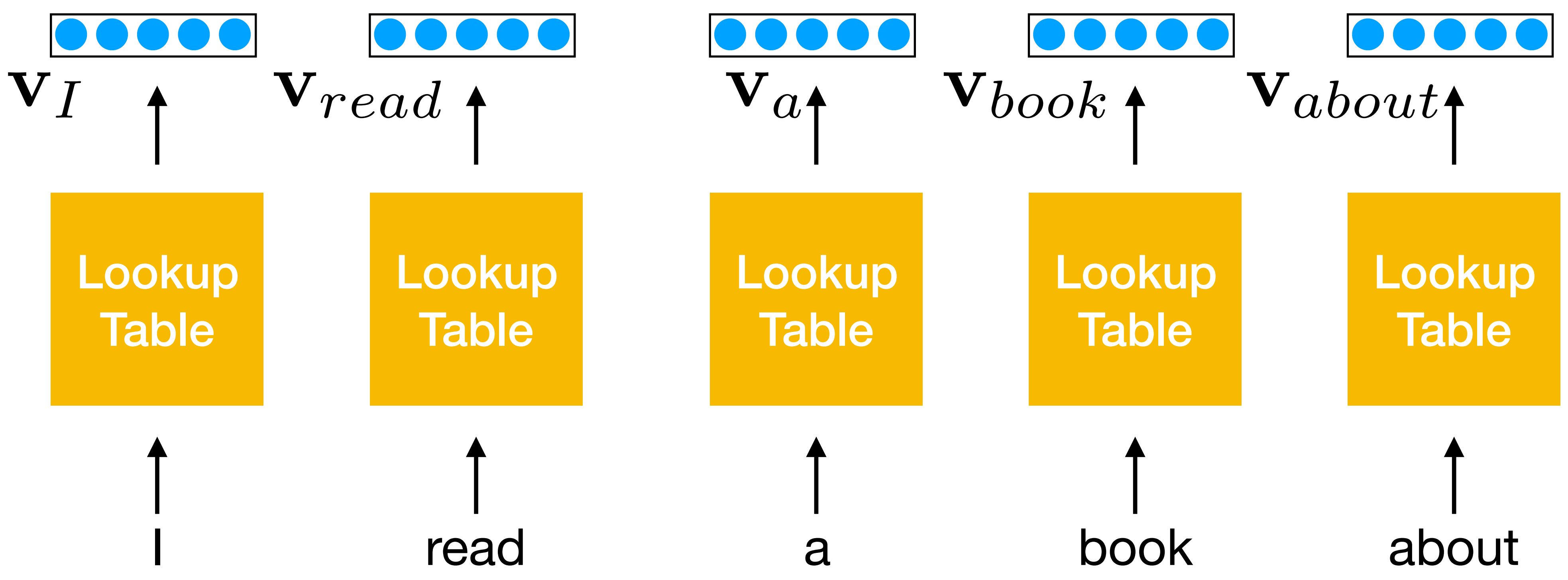
- Translate each word in the (fixed) vocabulary to a vector.
  - Typical dimensions: 100-300
  - Translation is done using a lookup table.
  - Can be "pre-trained" (word2vec, glove)
- Dealing with "infinite" vocabularies:
  - word pieces, bpe

Word Embeddings

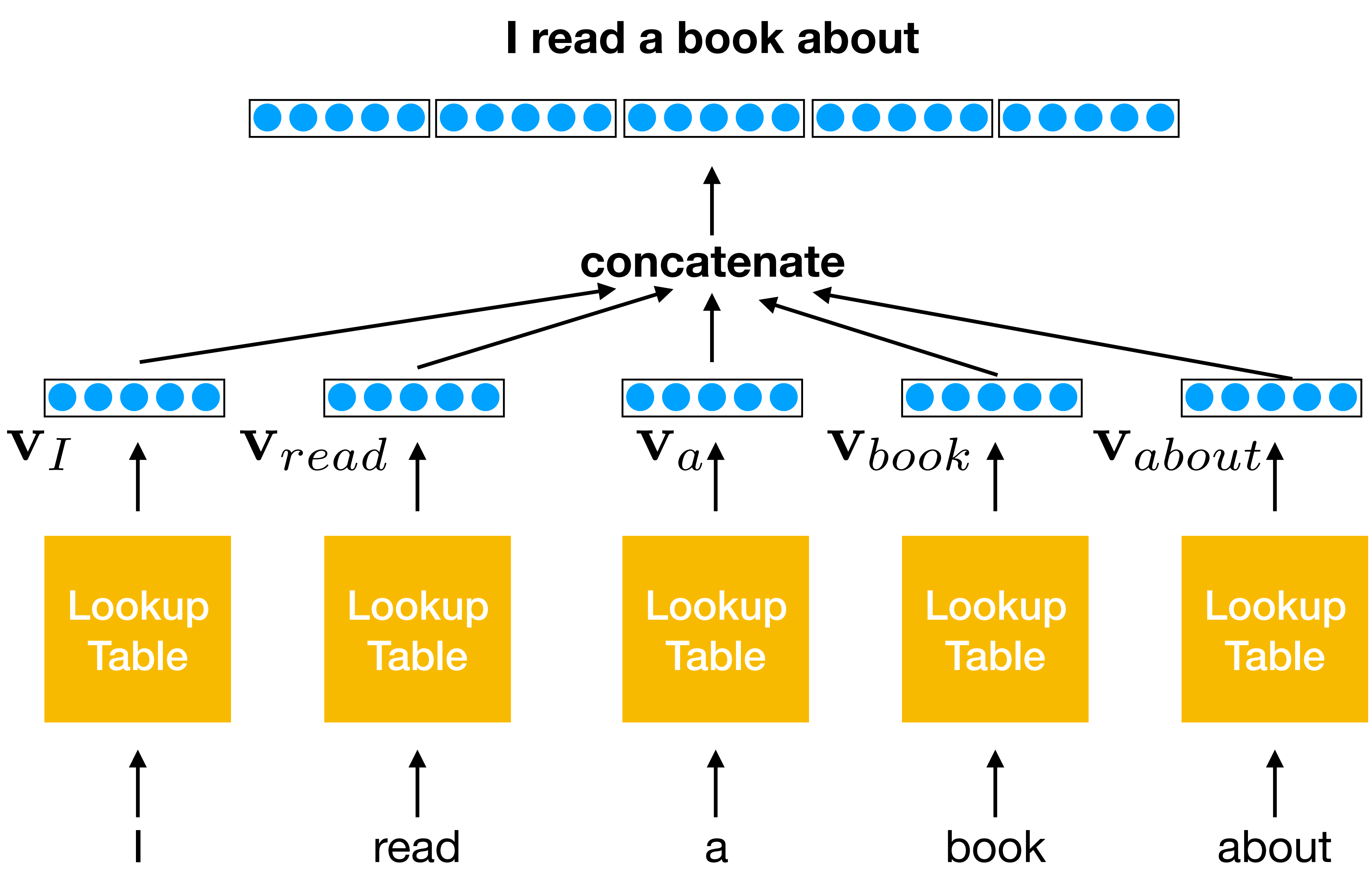
$$\mathbf{v}_{book} = \mathbf{E}[book]$$



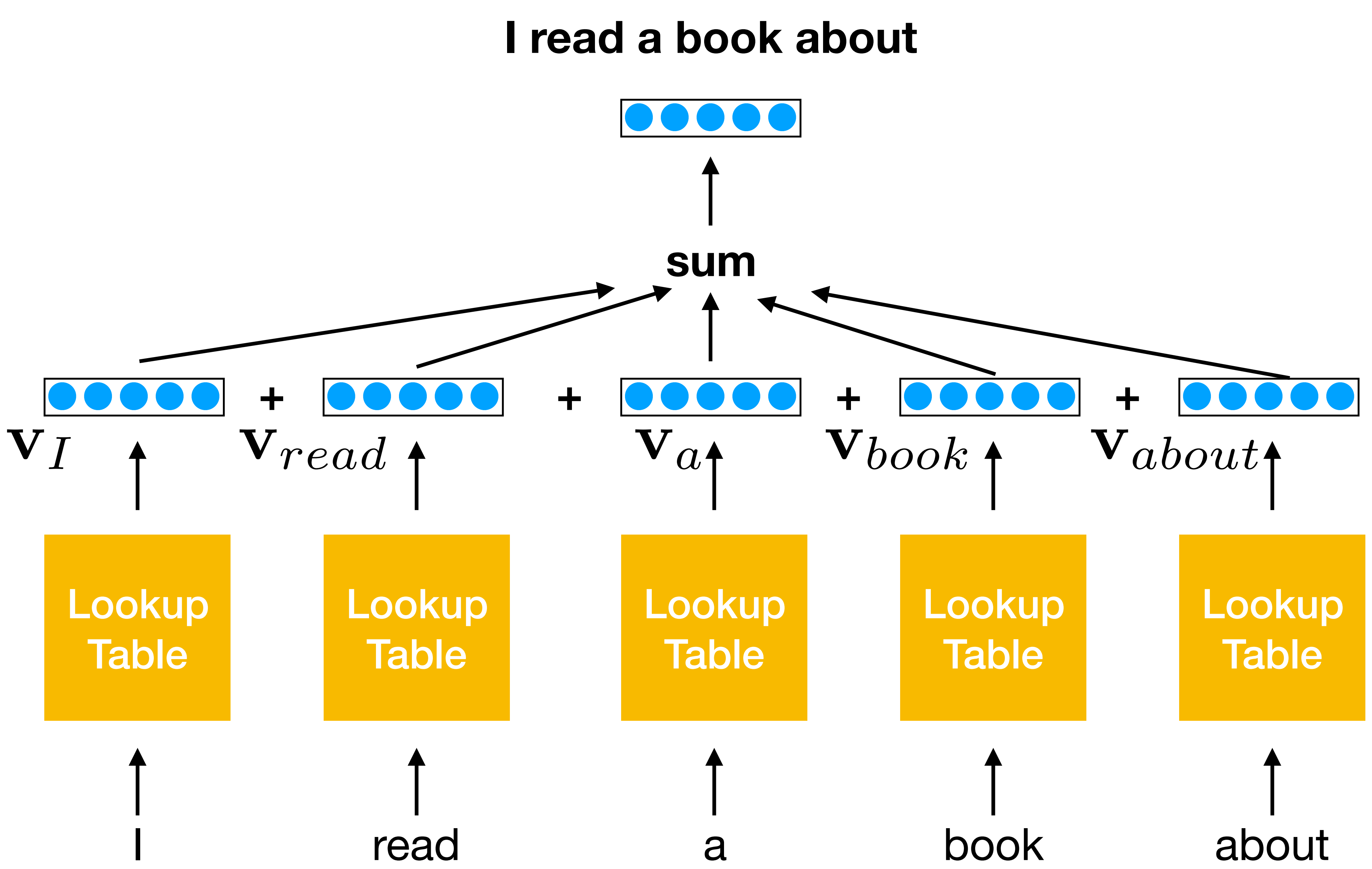
Combining Vectors



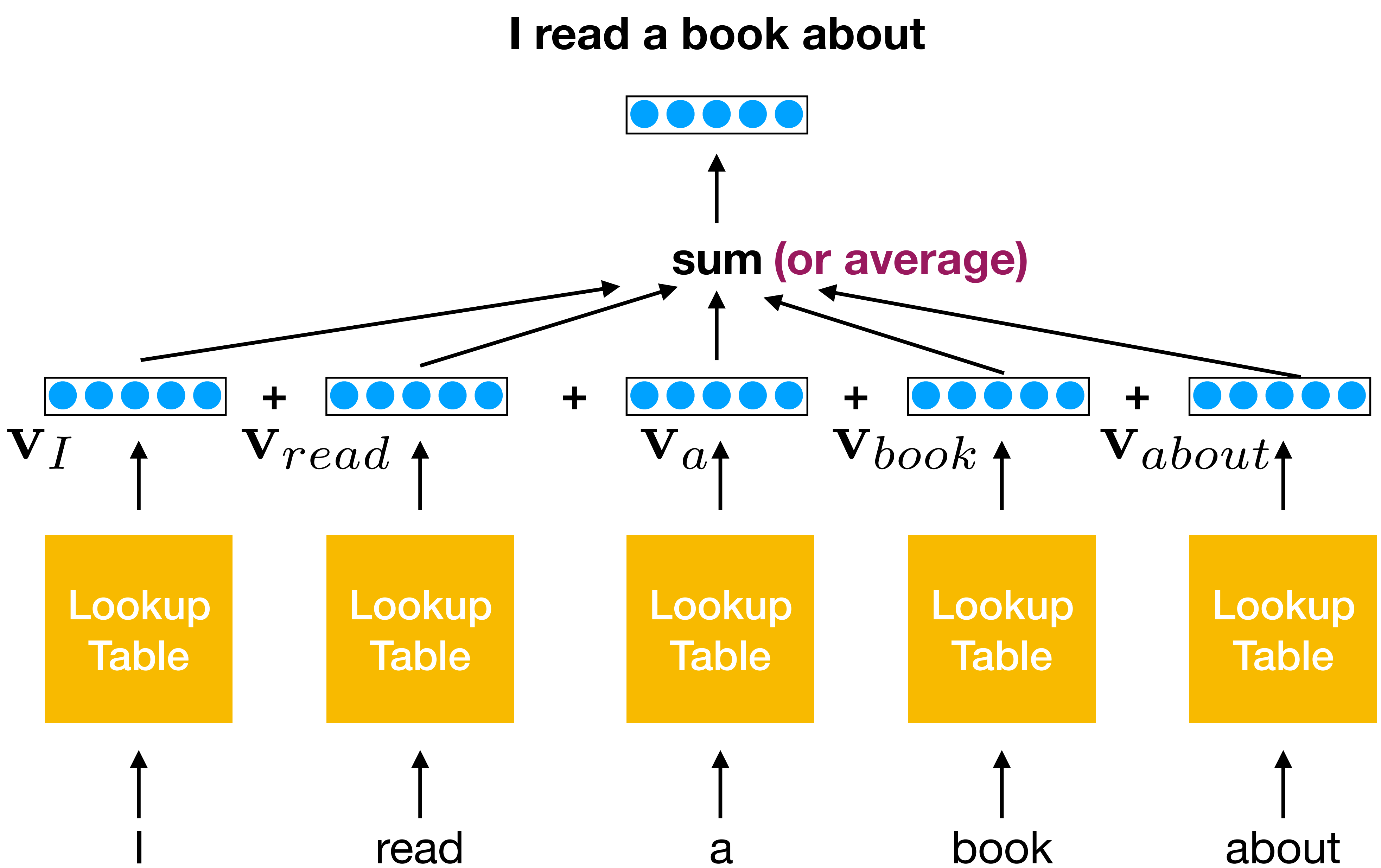
Combining Vectors



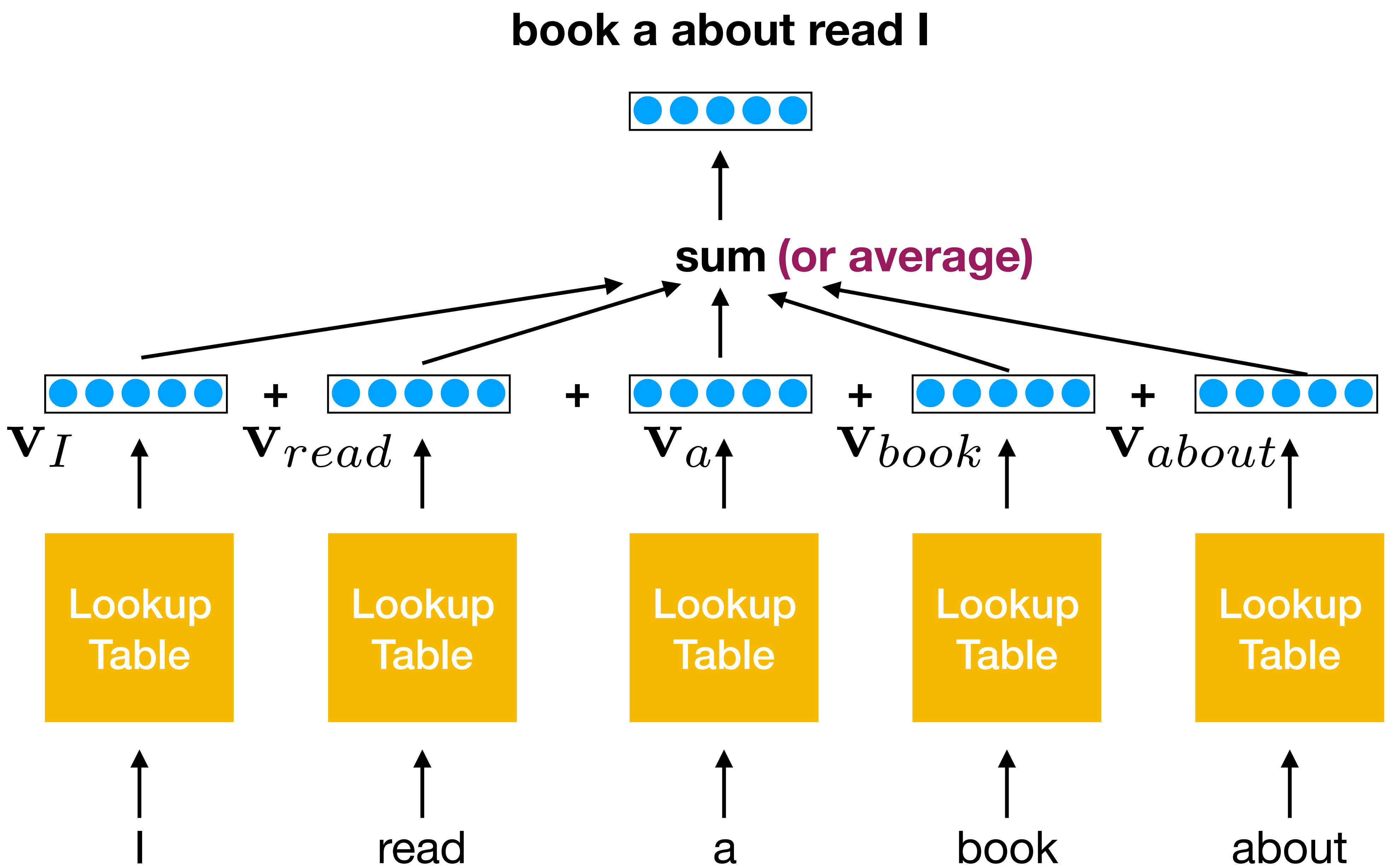
Combining Vectors



Combining Vectors



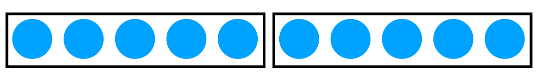
Combining Vectors



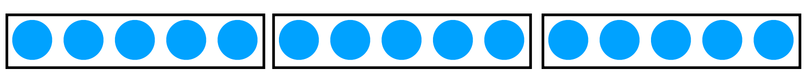
Combining Vectors

Concatenate

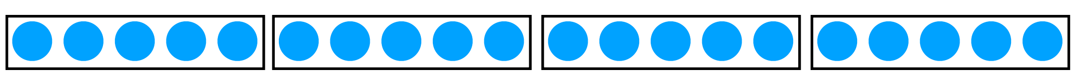
I read



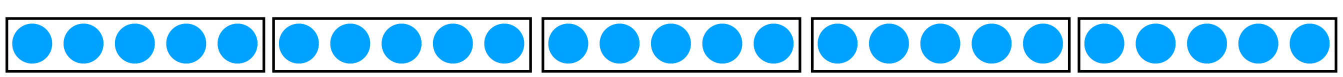
I read a



I read a book



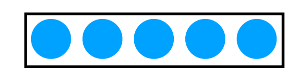
I read a book about



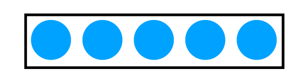
more words = longer vectors

Sum (or average)

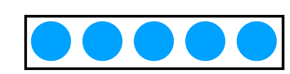
I read



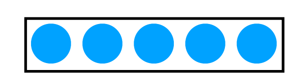
I read a



I read a book



I read a book about

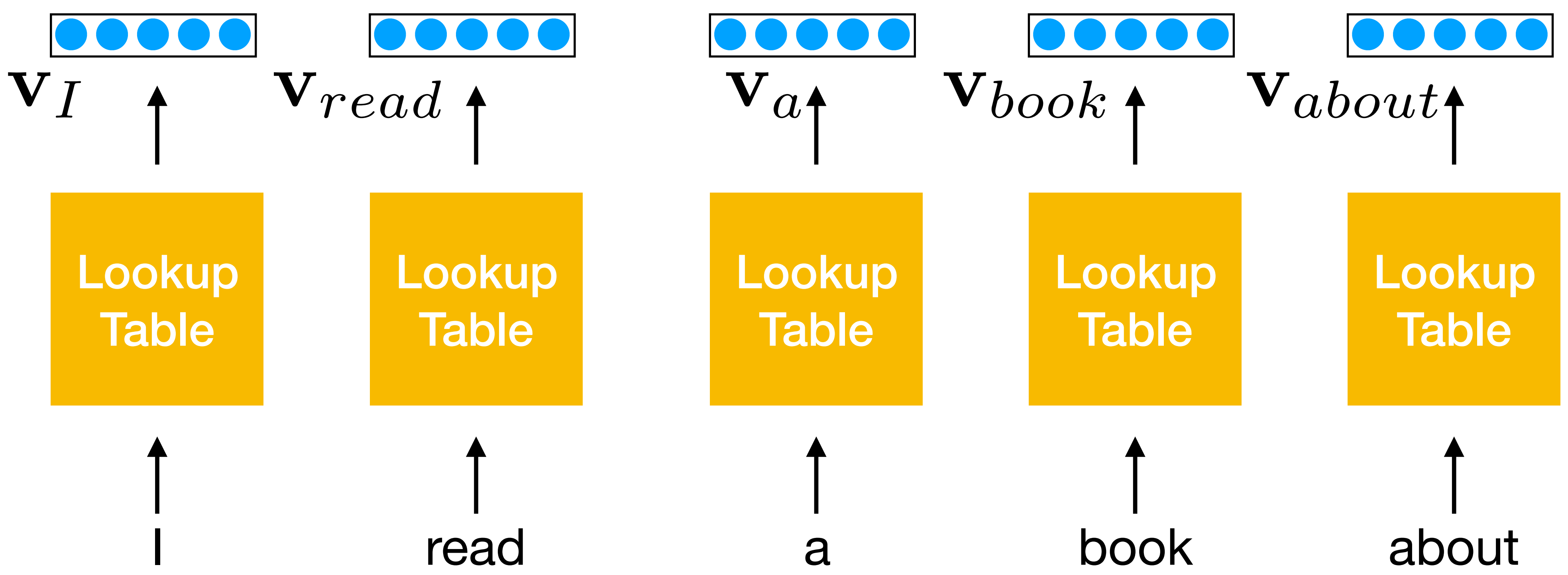


I book a read about
book about read I a
I a about book read
a read about book I
...

order invariant

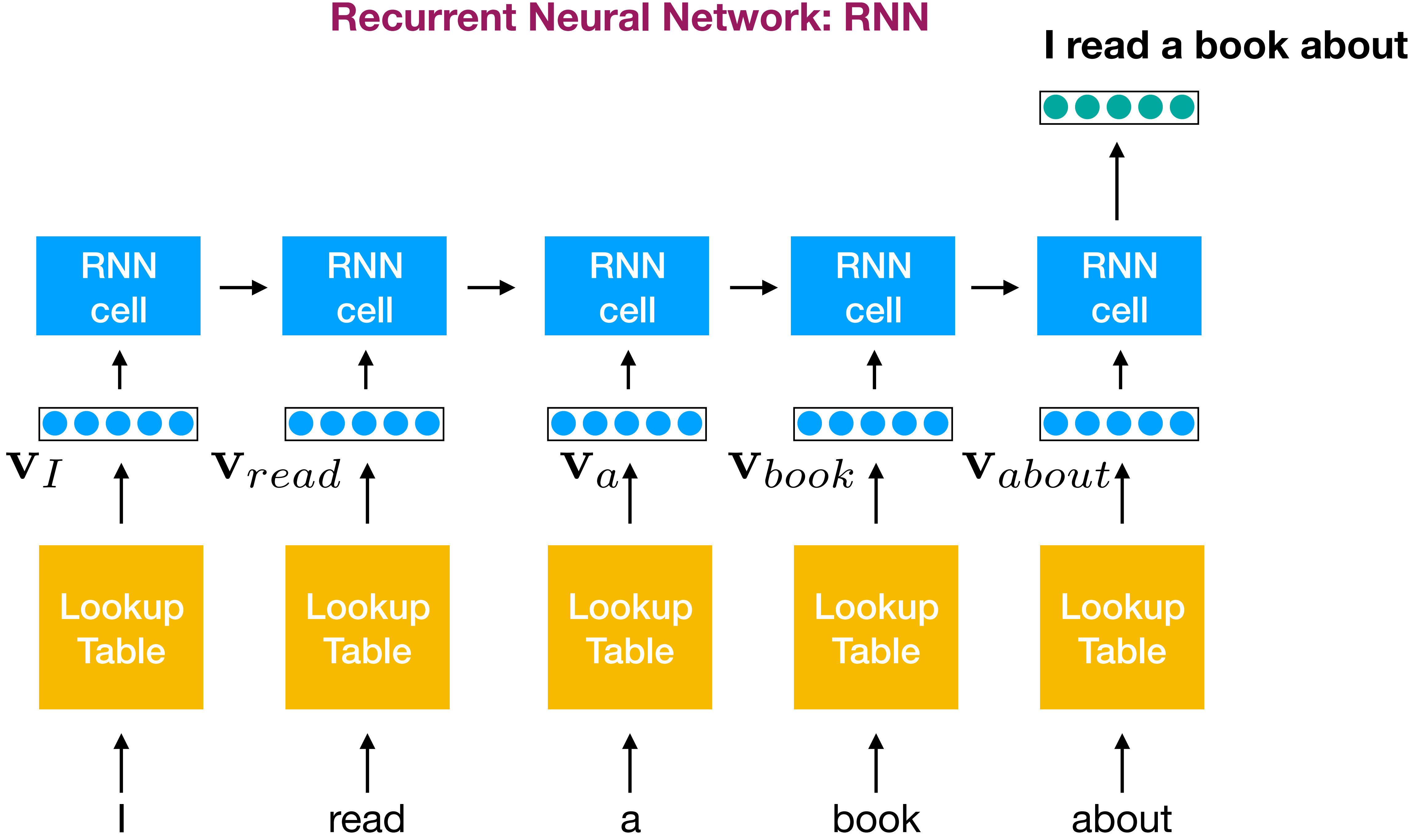
Combining Vectors

Recurrent Neural Network: RNN

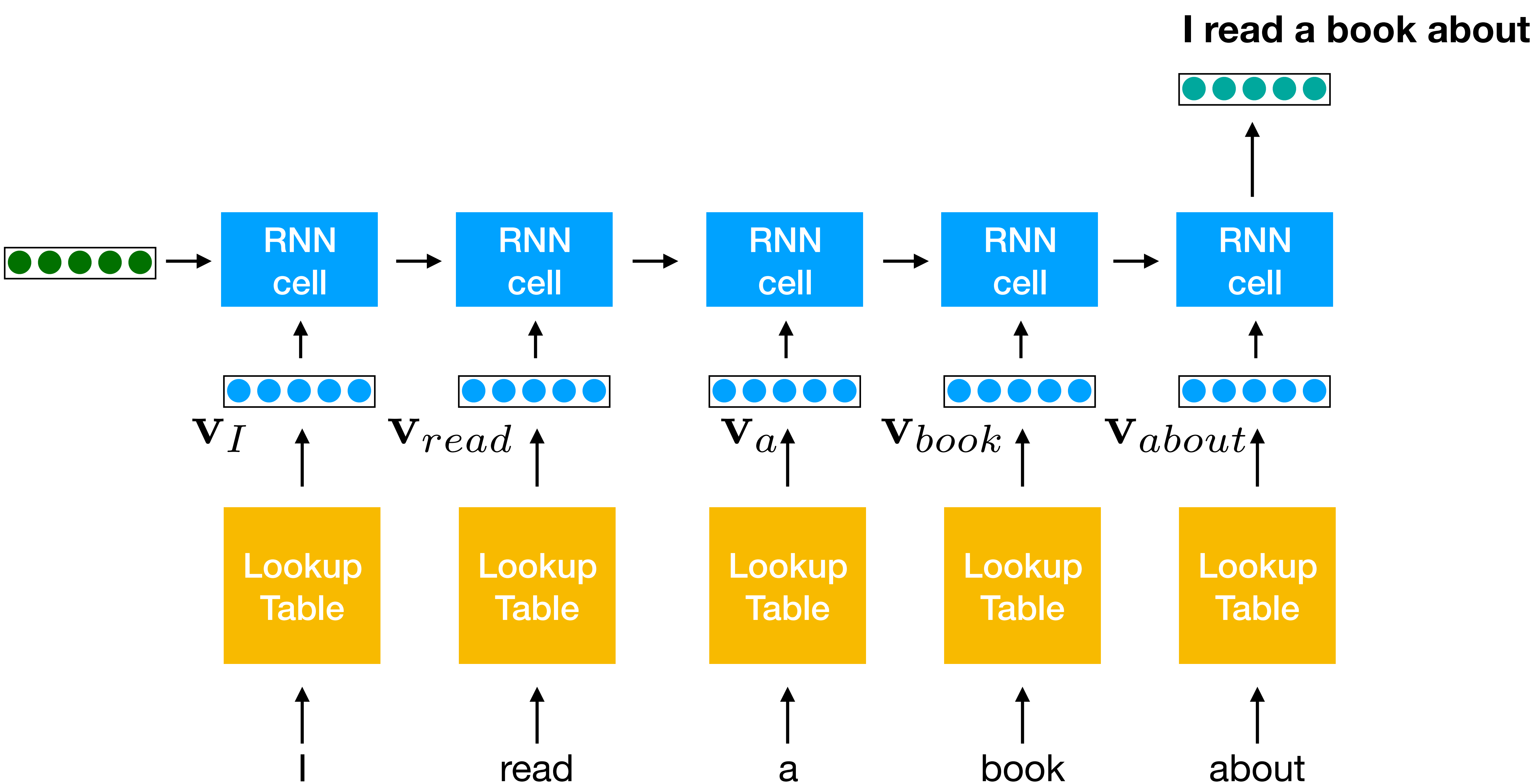


Combining Vectors

Recurrent Neural Network: RNN

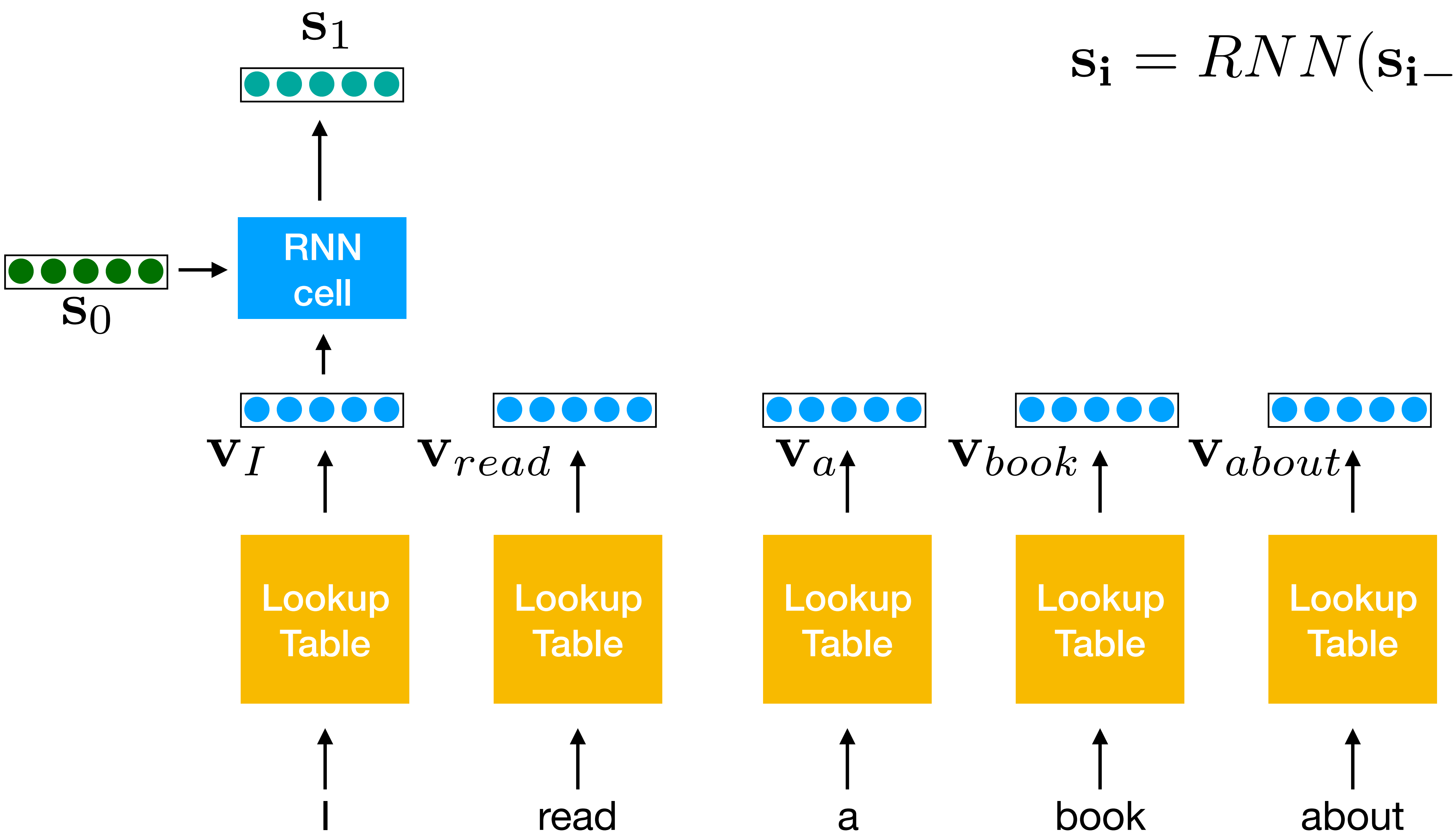


Combining Vectors

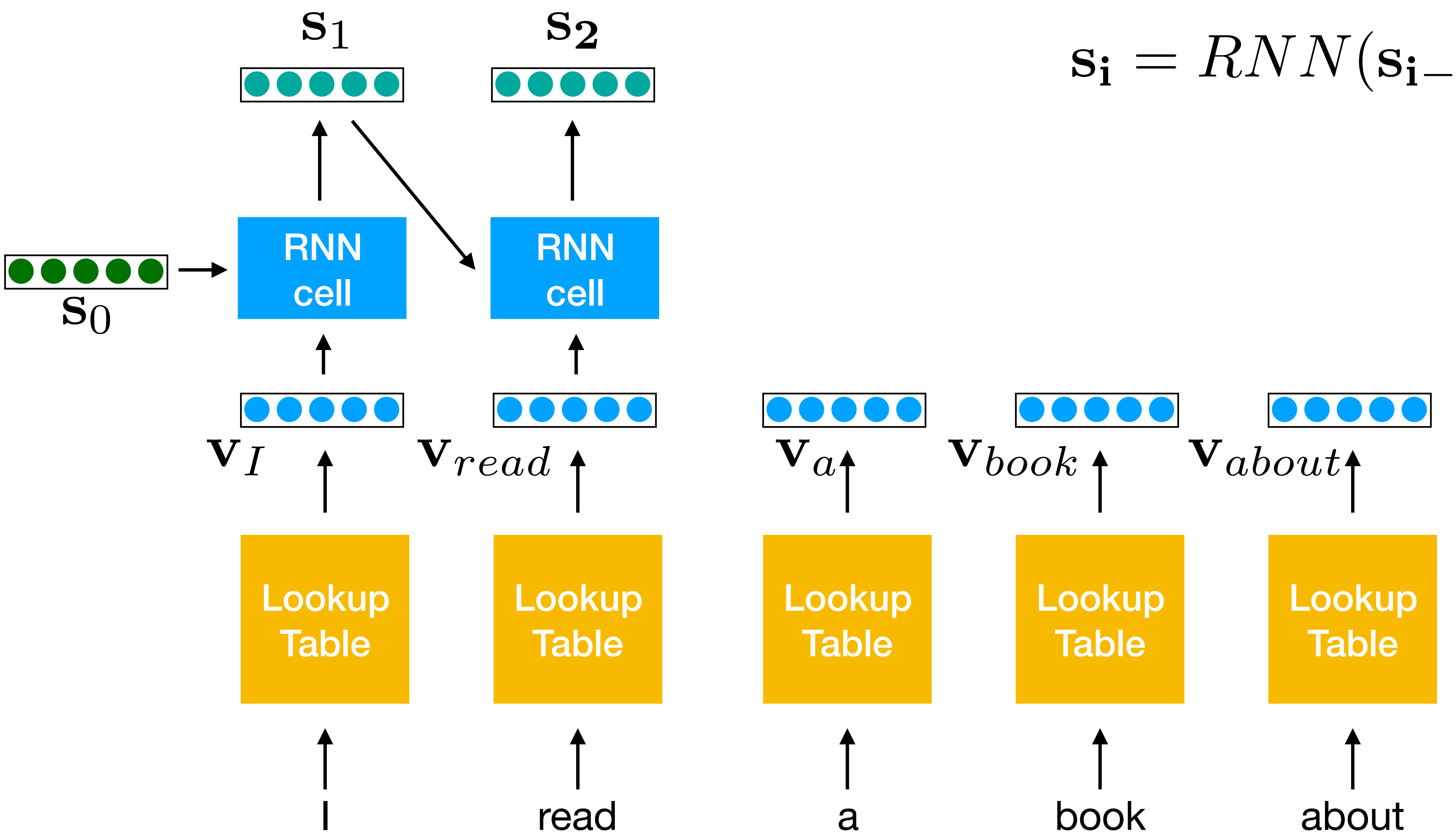


Combining Vectors

$$\mathbf{s_i} = RNN(\mathbf{s_{i-1}}, \mathbf{x_i})$$

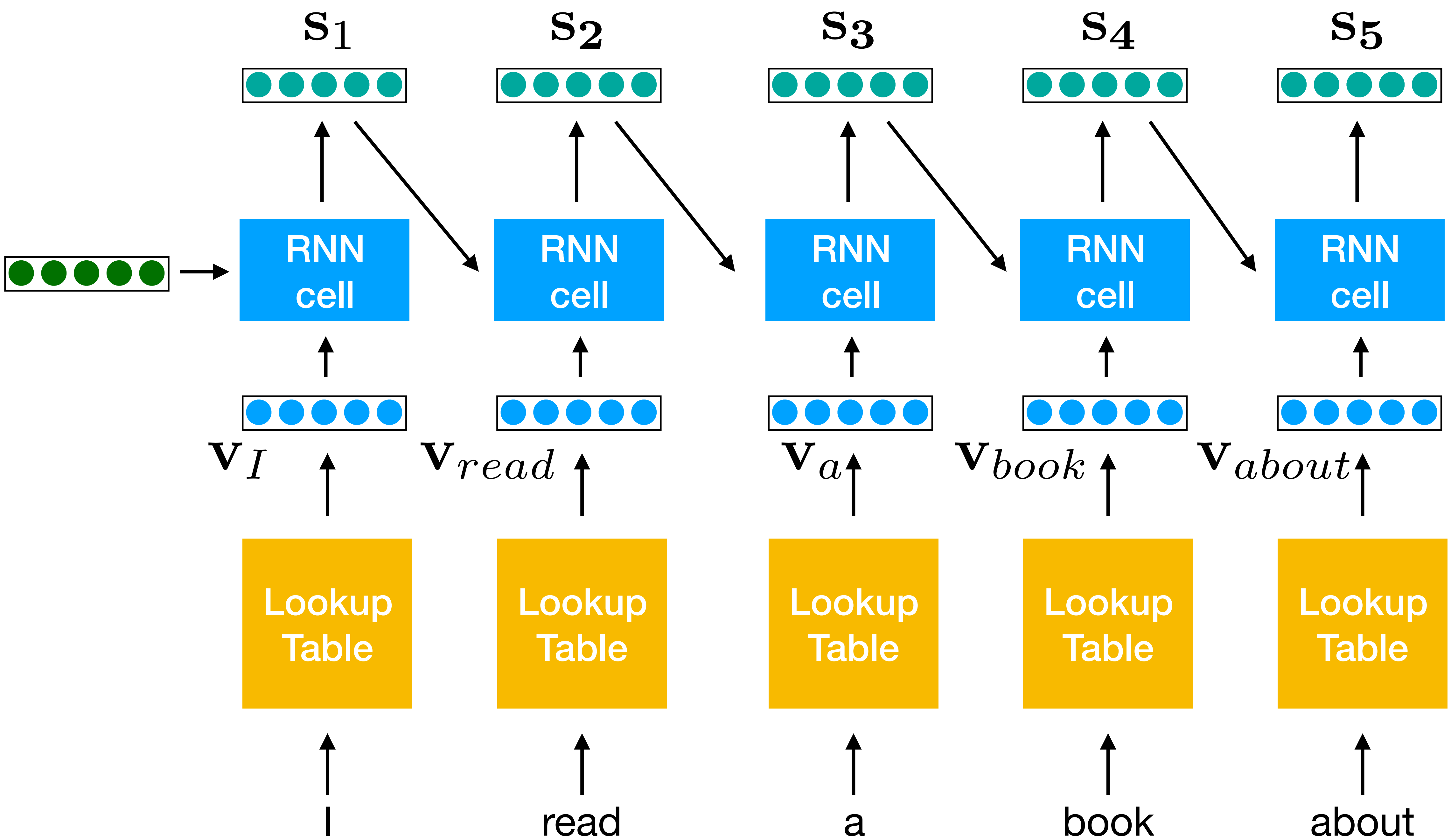


Combining Vectors

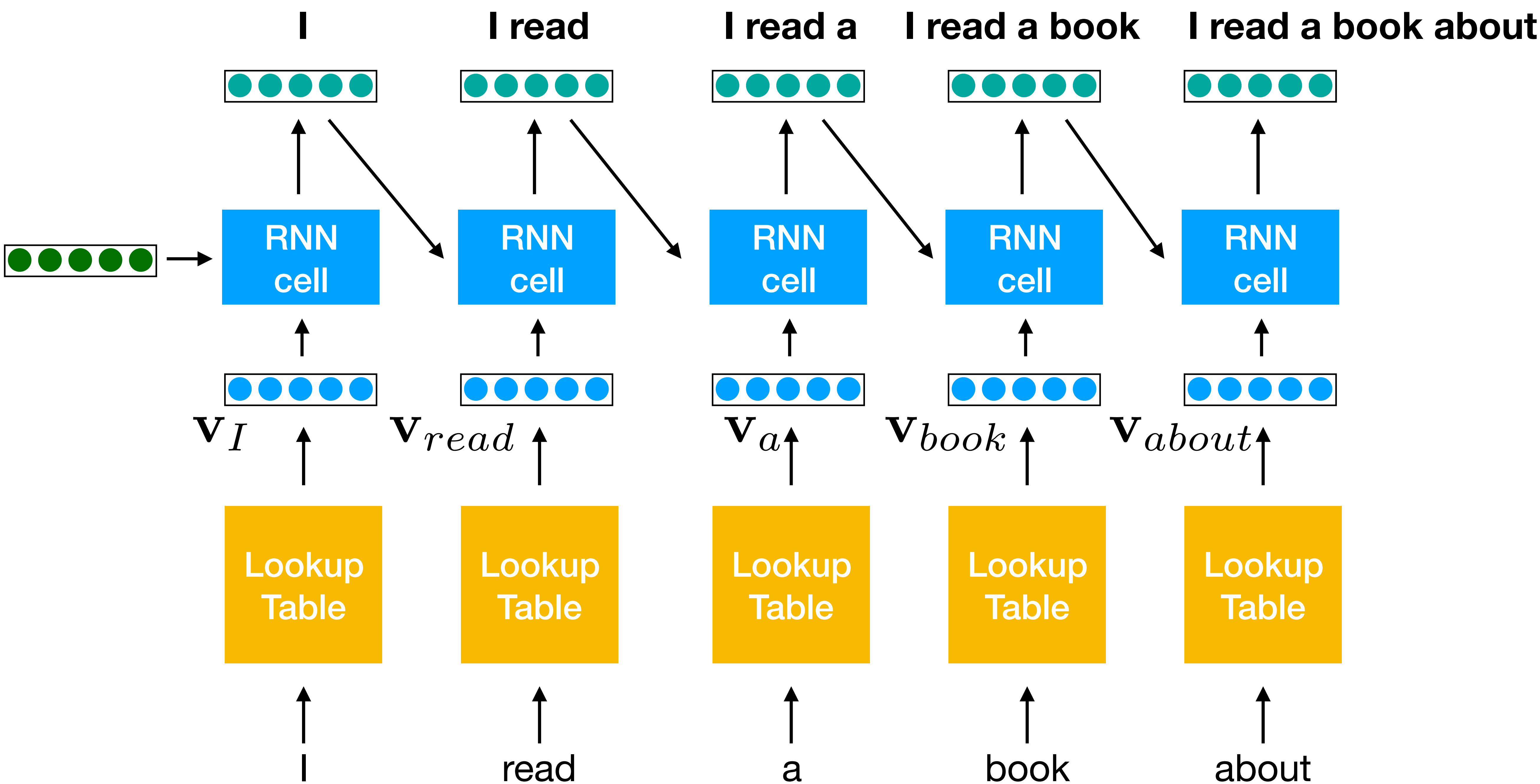


$$\mathbf{s_i} = RNN(\mathbf{s_{i-1}}, \mathbf{x_i})$$

Combining Vectors



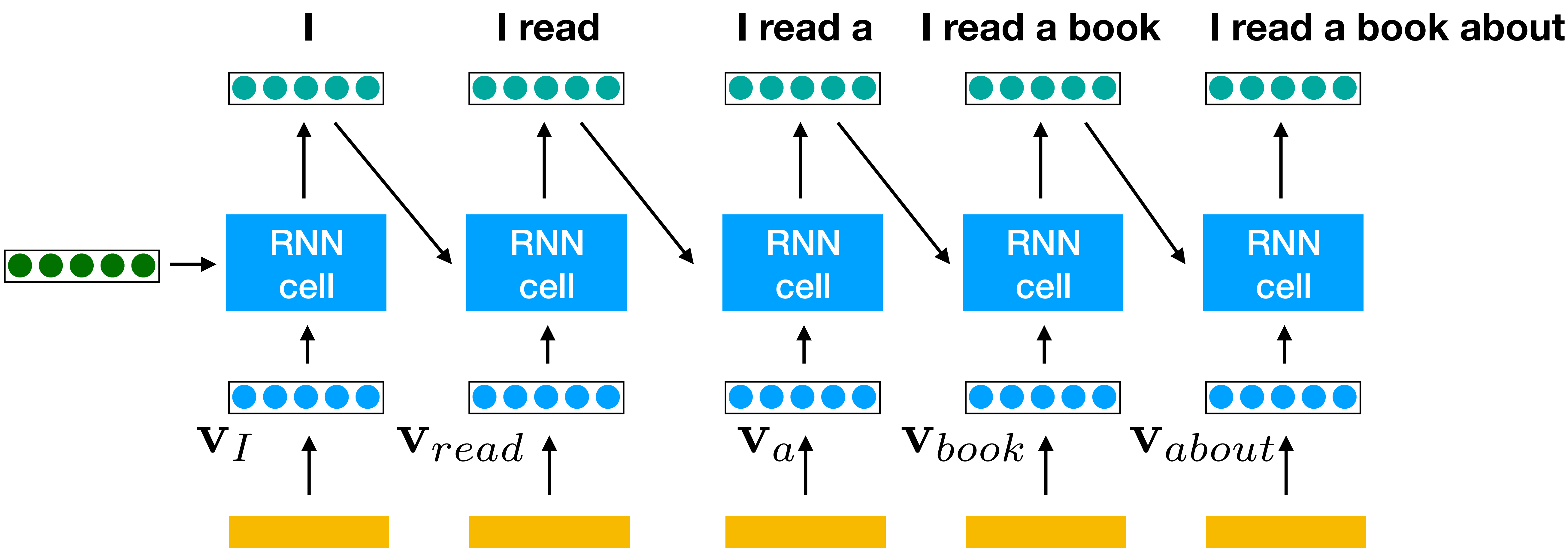
Combining Vectors



Combining Vectors

Recurrent Neural Network: RNN

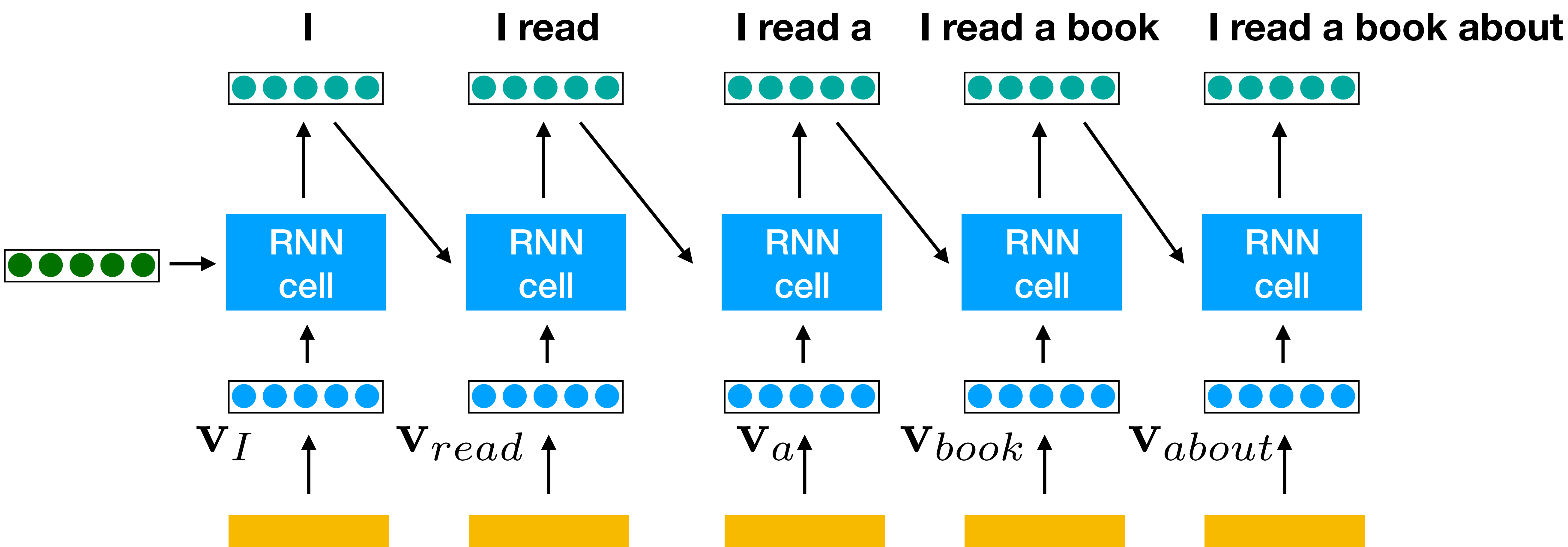
$$\mathbf{s_i} = RNN(\mathbf{s_{i-1}}, \mathbf{x_i})$$



Combining Vectors

Recurrent Neural Network: RNN

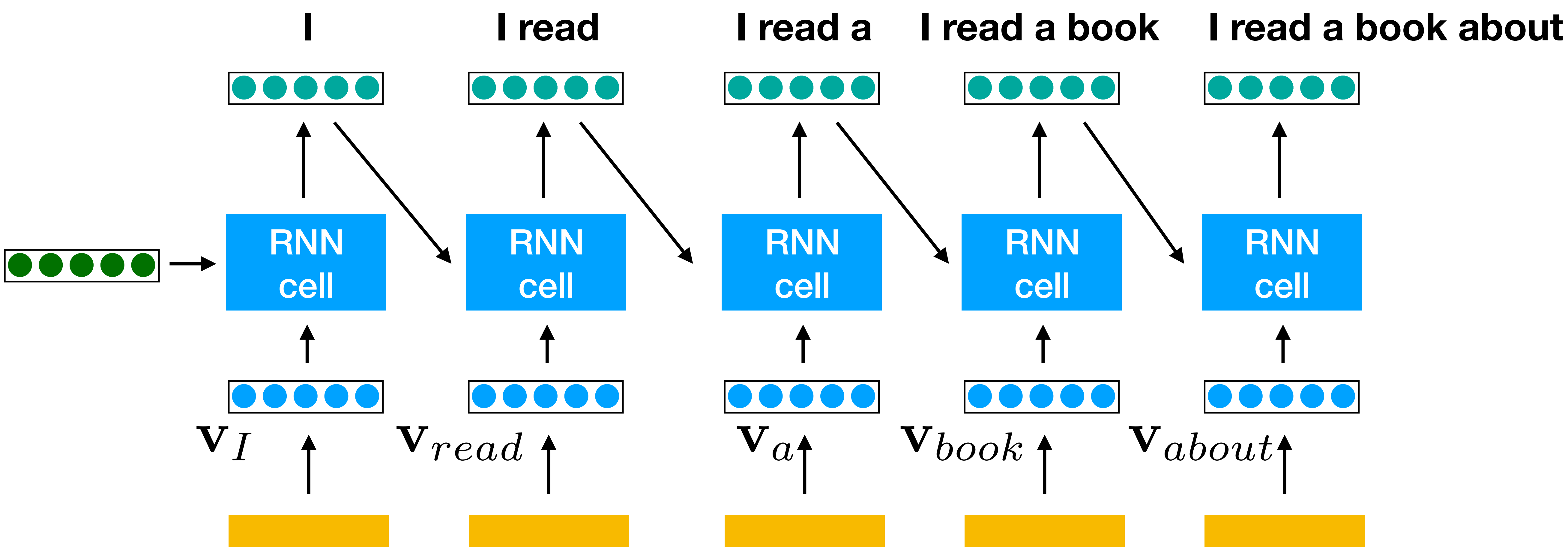
$$R_{SRNN}(\mathbf{s_{i-1}}, \mathbf{x_i}) = tanh(\mathbf{W^s} \cdot \mathbf{s_{i-1}} + \mathbf{W^x} \cdot \mathbf{x_i})$$



Combining Vectors

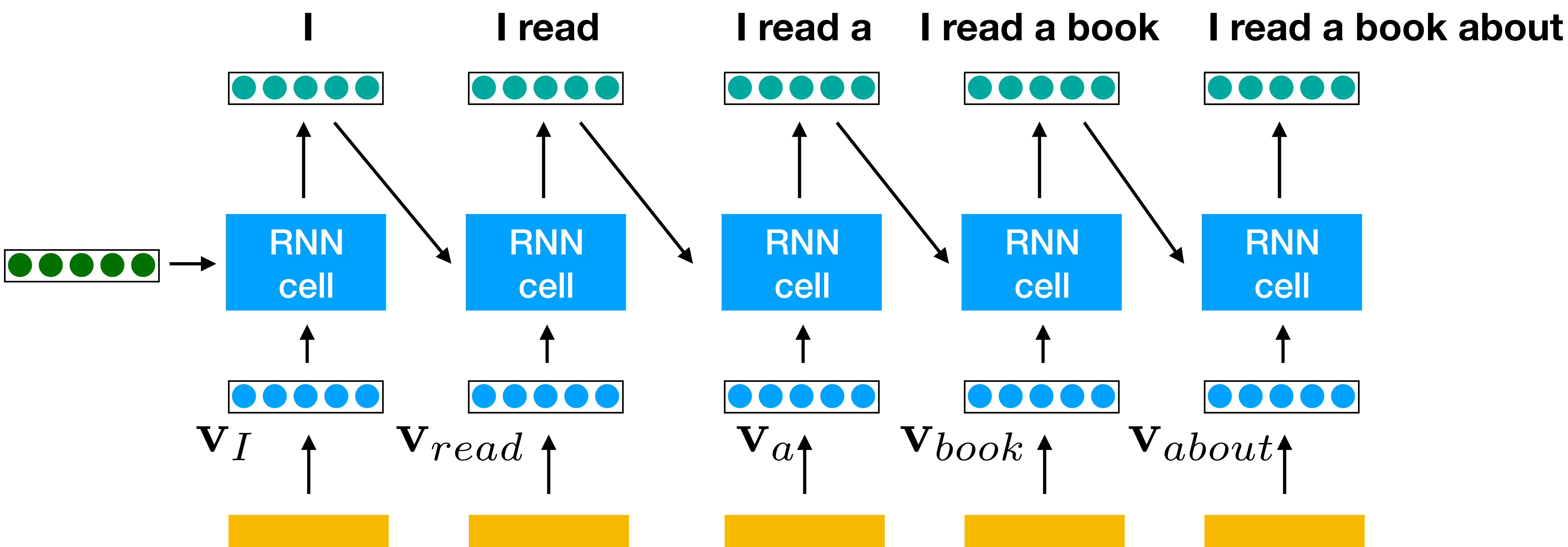
Recurrent Neural Network: RNN

$$
\begin{aligned}
R_{LSTM}(\mathbf{s_{j-1}}, \mathbf{x_j}) =& [\mathbf{c_j}; \mathbf{h_j}] \\
\mathbf{c_j} =& \mathbf{c_{j-1}} \odot \mathbf{f} + \mathbf{g} \odot \mathbf{i} \\
\mathbf{h_j} =& \tanh(\mathbf{c_j}) \odot \mathbf{o} \\
\mathbf{i} =& \sigma(\mathbf{W^{xi}} \cdot \mathbf{x_j} + \mathbf{W^{hi}} \cdot \mathbf{h_{j-1}}) \\
\mathbf{f} =& \sigma(\mathbf{W^{xf}} \cdot \mathbf{x_j} + \mathbf{W^{hf}} \cdot \mathbf{h_{j-1}}) \\
\mathbf{o} =& \sigma(\mathbf{W^{xo}} \cdot \mathbf{x_j} + \mathbf{W^{ho}} \cdot \mathbf{h_{j-1}}) \\
\mathbf{g} =& \tanh(\mathbf{W^{xg}} \cdot \mathbf{x_j} + \mathbf{W^{hg}} \cdot \mathbf{h_{j-1}})
\end{aligned}
$$



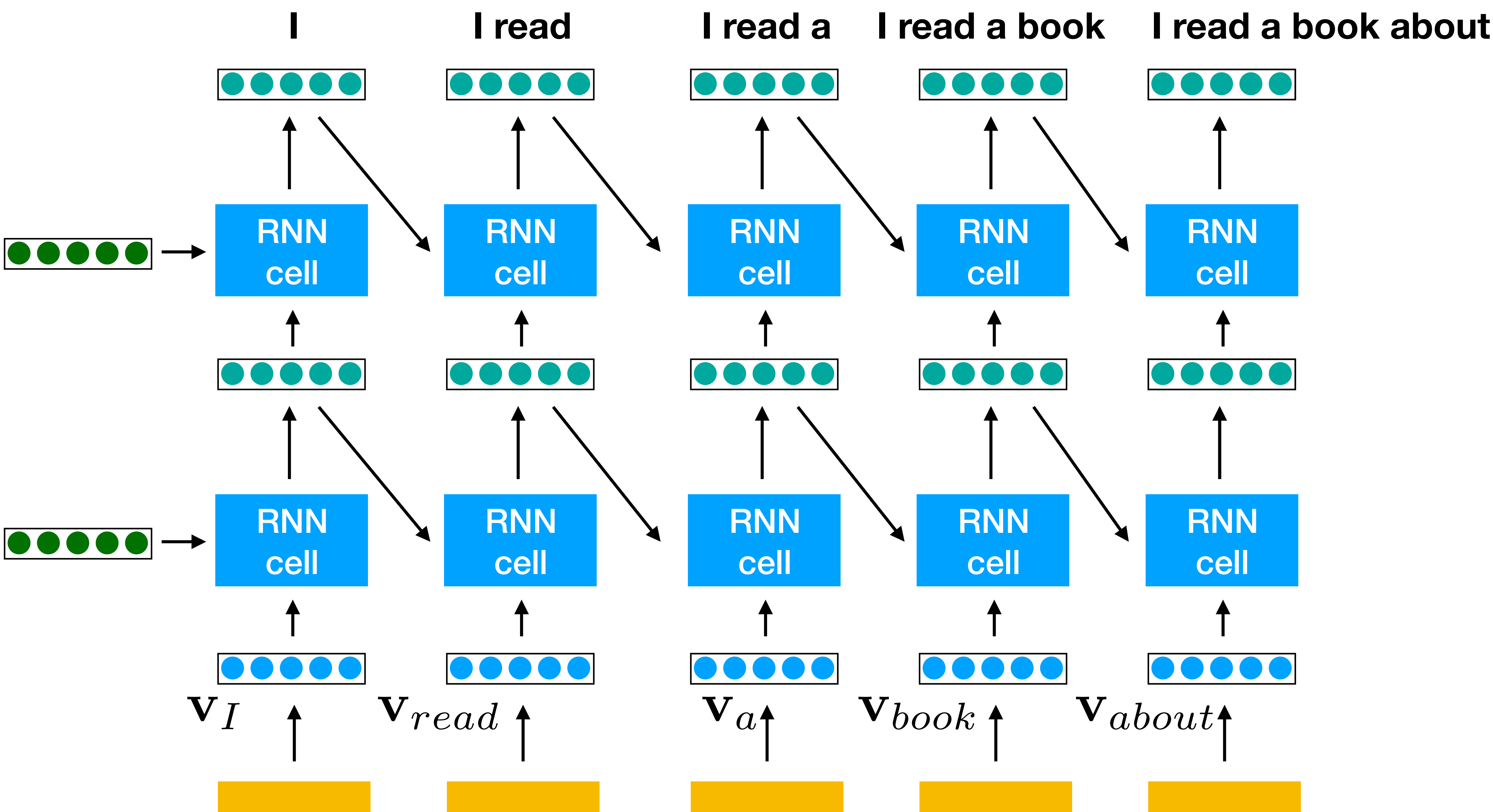
Combining Vectors

Recurrent Neural Network: RNN

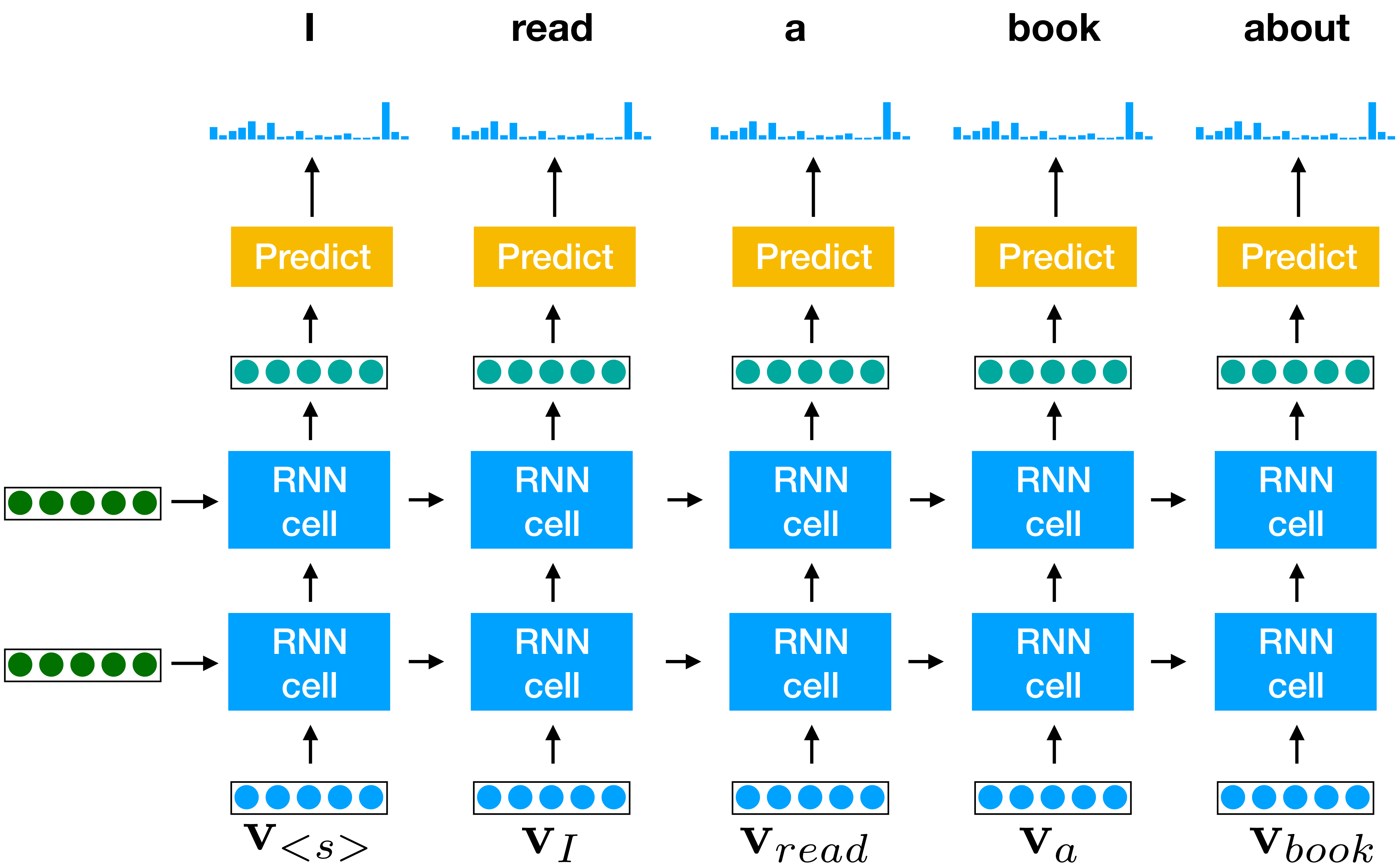


Combining Vectors

multi-layer RNN

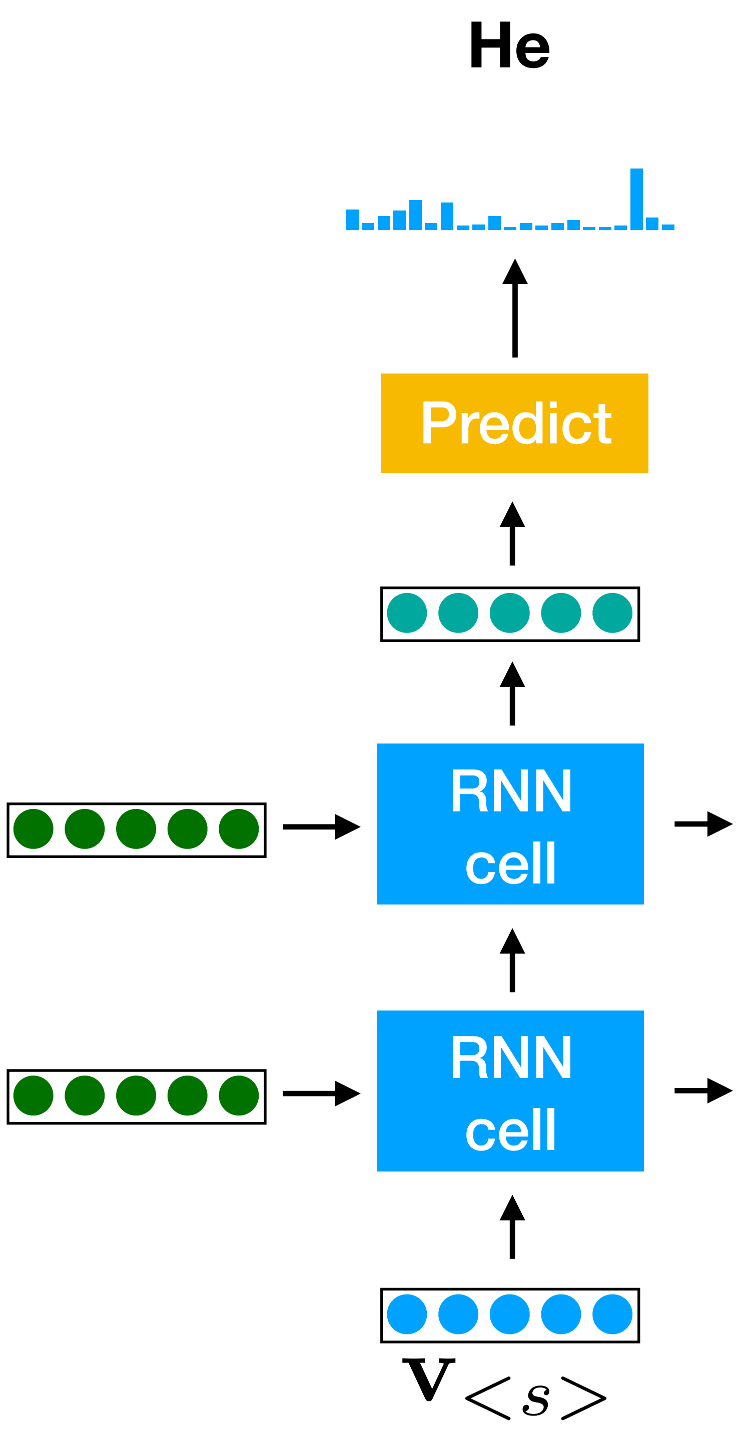


Training

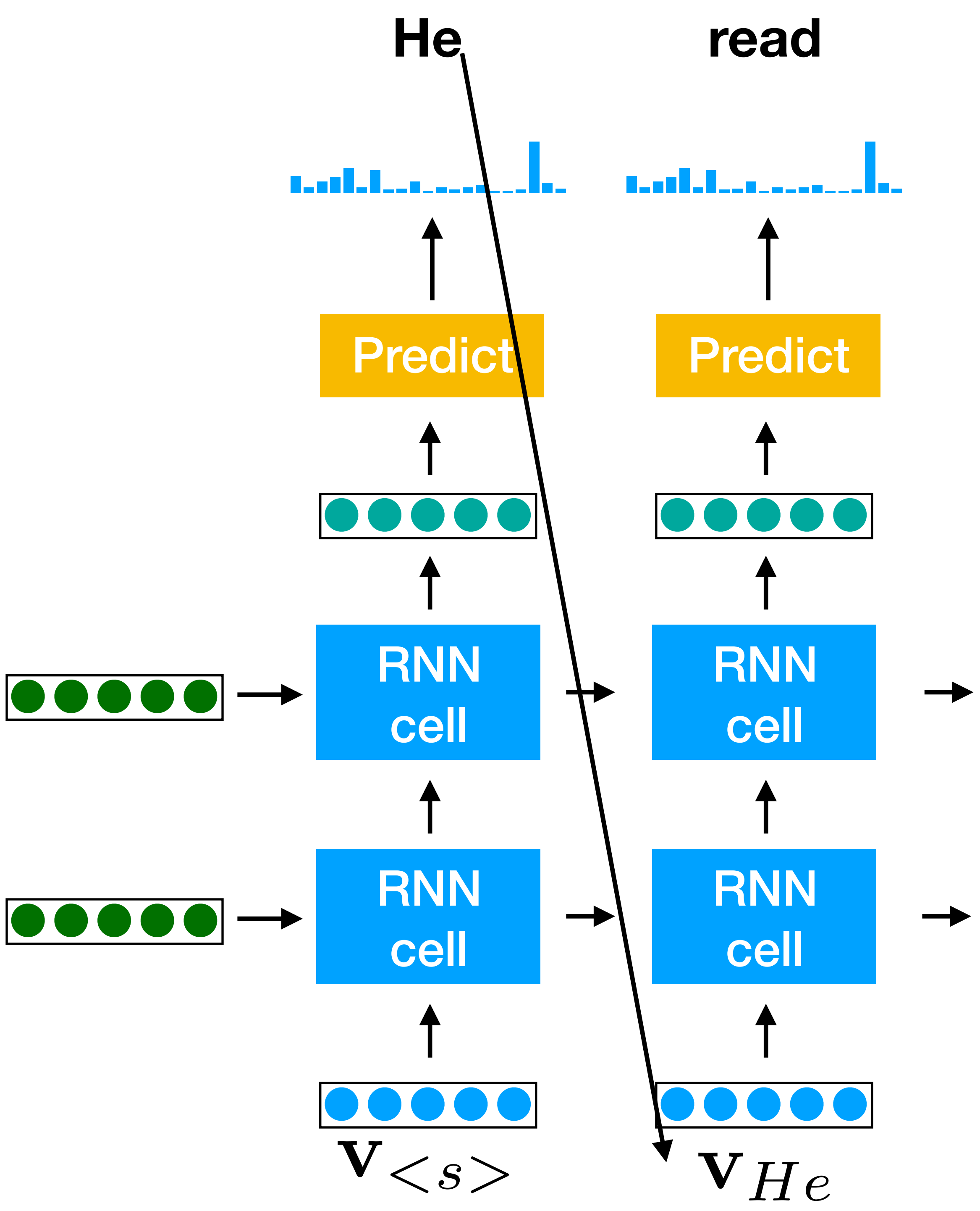


I
read
a
book
about
Predict
Predict
Predict
Predict
Predict
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
$\mathbf{v}_{<s>}$
$\mathbf{v}_{I}$
$\mathbf{v}_{read}$
$\mathbf{v}_{a}$
$\mathbf{v}_{book}$

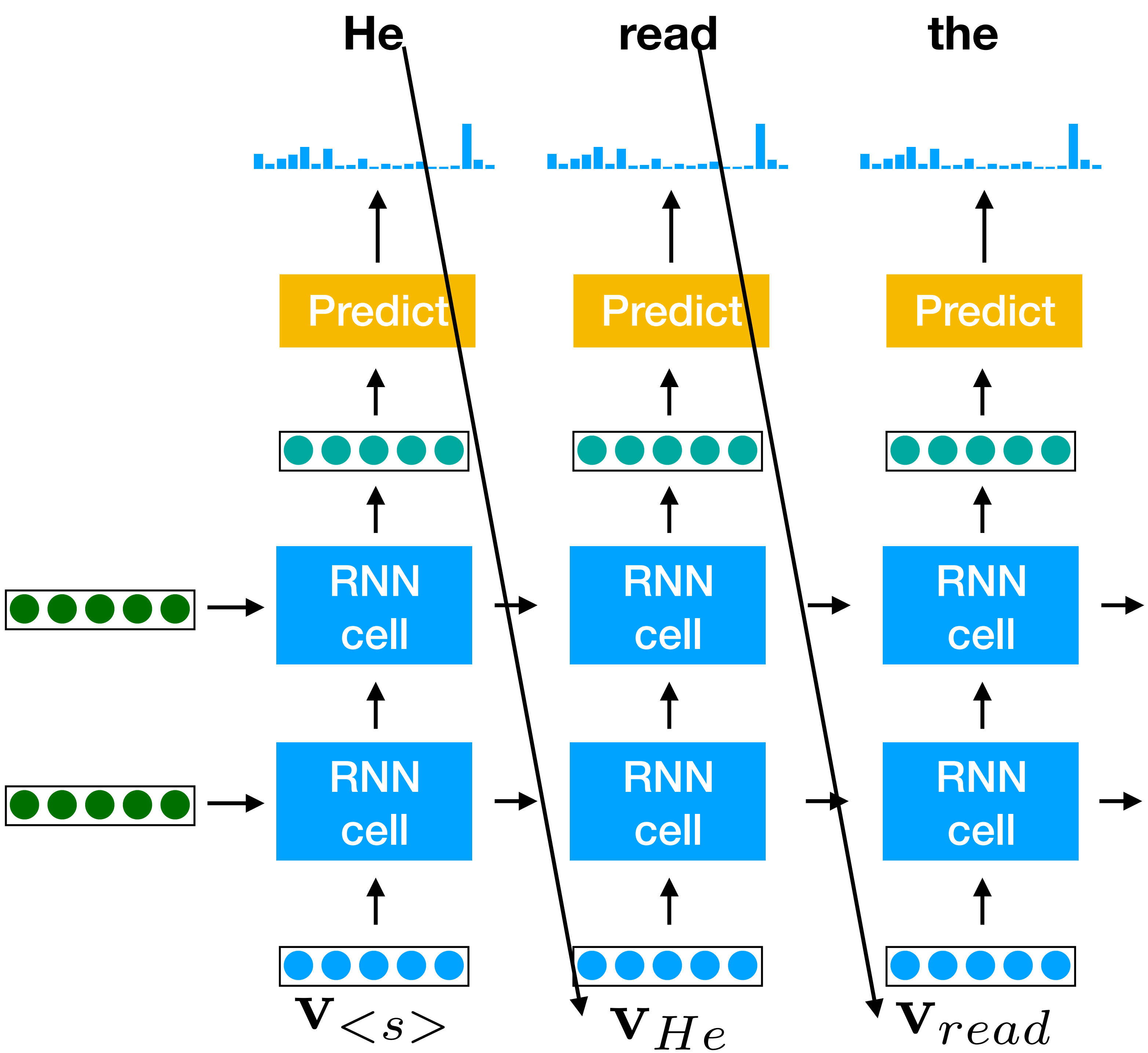
Generation



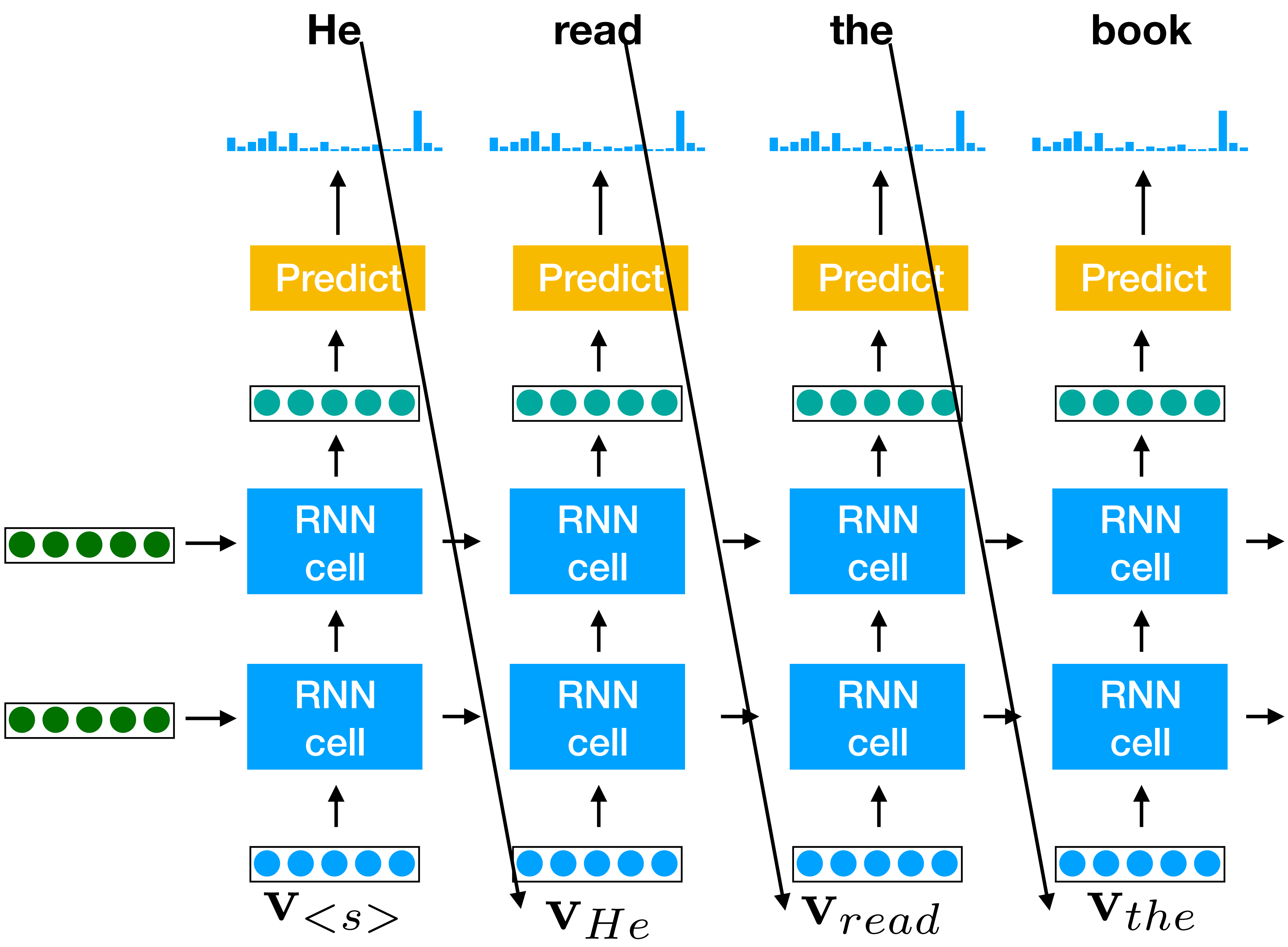
Generation


He
read
Predict
Predict
RNN cell
RNN cell
RNN cell
RNN cell
$\mathbf{v}_{<s>}$
$\mathbf{v}_{He}$

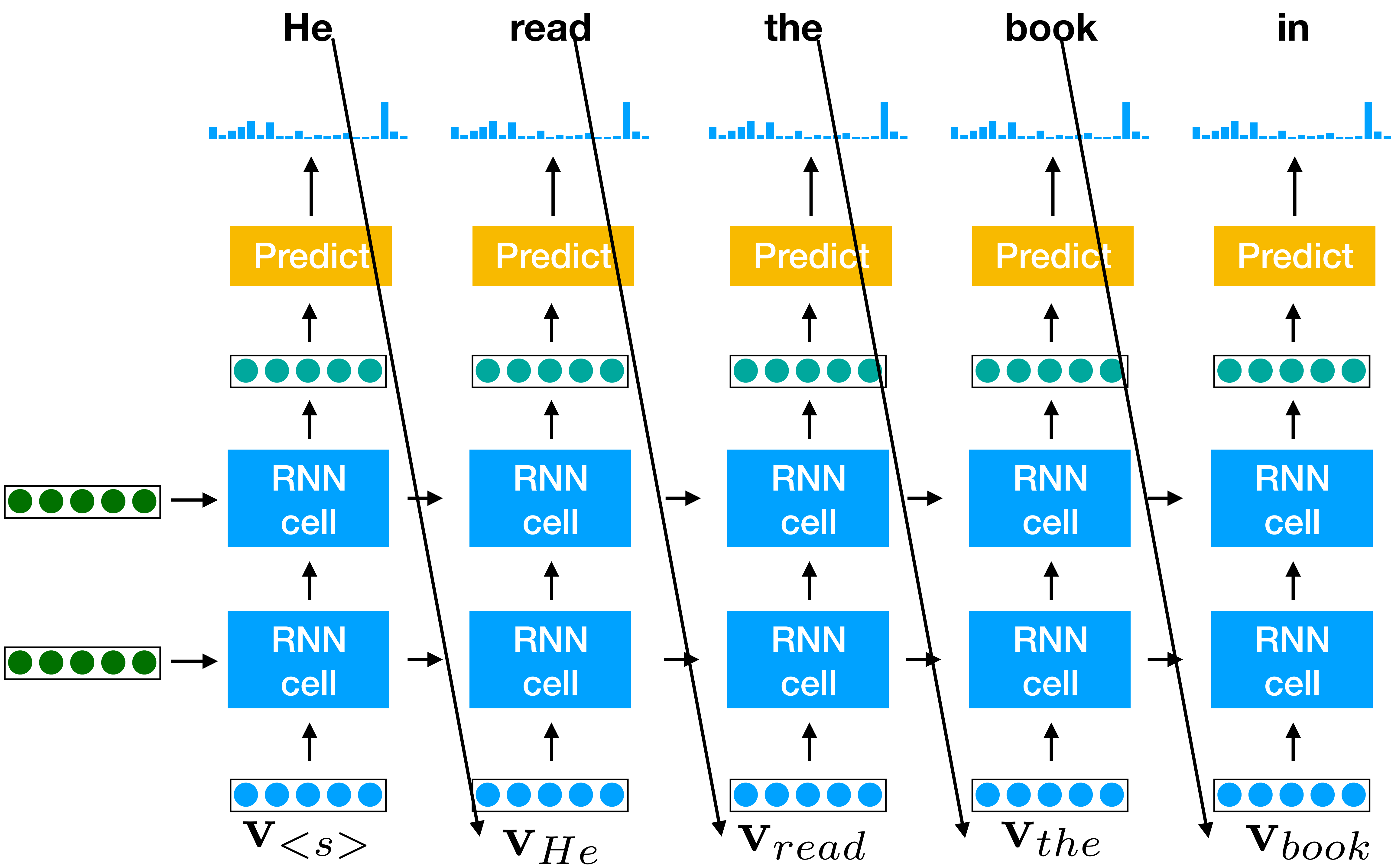
Generation


He
read
the
Predict
Predict
Predict
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
$\mathbf{v}_{<s>}$
$\mathbf{v}_{He}$
$\mathbf{v}_{read}$

Generation


He
read
the
book
Predict
Predict
Predict
Predict
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
$\mathbf{v}_{<s>}$
$\mathbf{v}_{He}$
$\mathbf{v}_{read}$
$\mathbf{v}_{the}$

Generation



He
read
the
book
in
Predict
Predict
Predict
Predict
Predict
RNN cell
$\mathbf{v}_{<s>}$
$\mathbf{v}_{He}$
$\mathbf{v}_{read}$
$\mathbf{v}_{the}$
$\mathbf{v}_{book}$

Generation Algorithm

- Find a corpus.
- Train a language model on a corpus
- Sample from the language model
- "Control" the generation by training on different corpora.

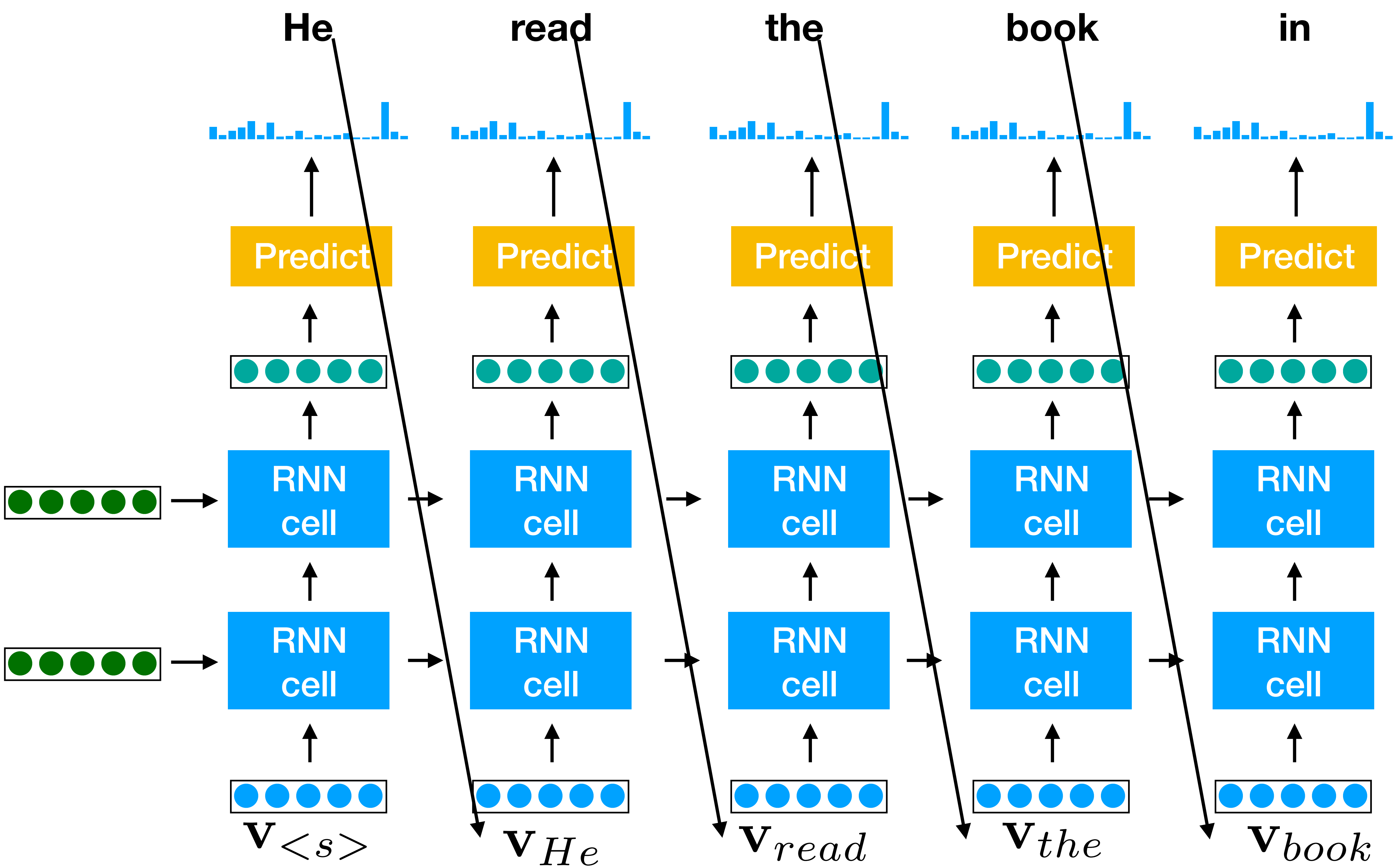
Generation Algorithm

- Considerations:
  - What is the vocabulary?
    - Words?
    - Characters?
    - In between characters and words?
    - If words/parts, which ones?
  - **Why do we want to do this?**

Conditioned Generation

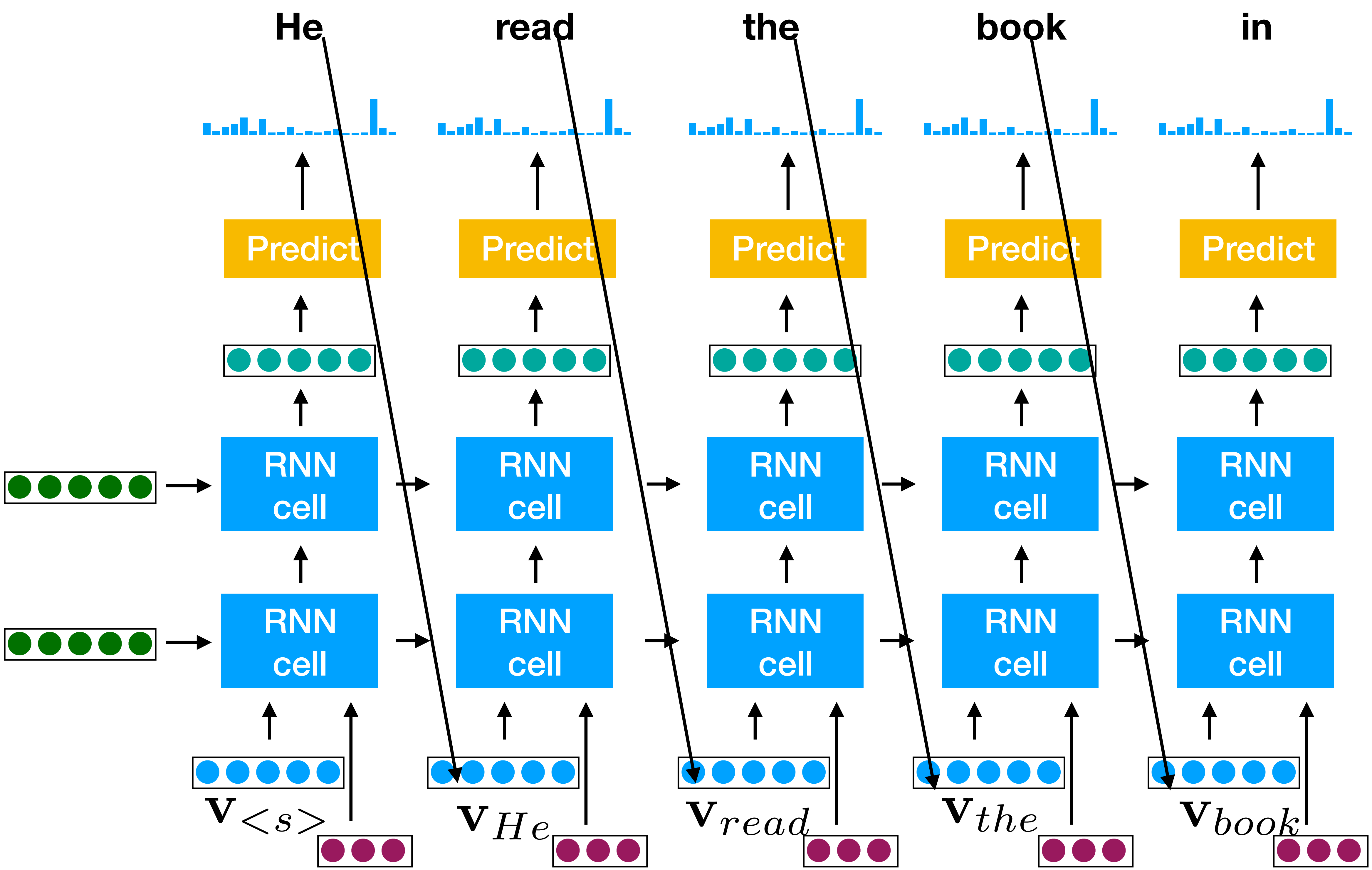
- `gen_so_far` = ["<s>"]
- **cond**
- while True:
    - `next_word_distribution` = p(`next` | `gen_so_far`, **cond**)
    - sample `next_word` from `next_word_distribution`
    - if `next_word` == "</s>": break
    - `gen_so_far`.append(`next_word`)

Generation

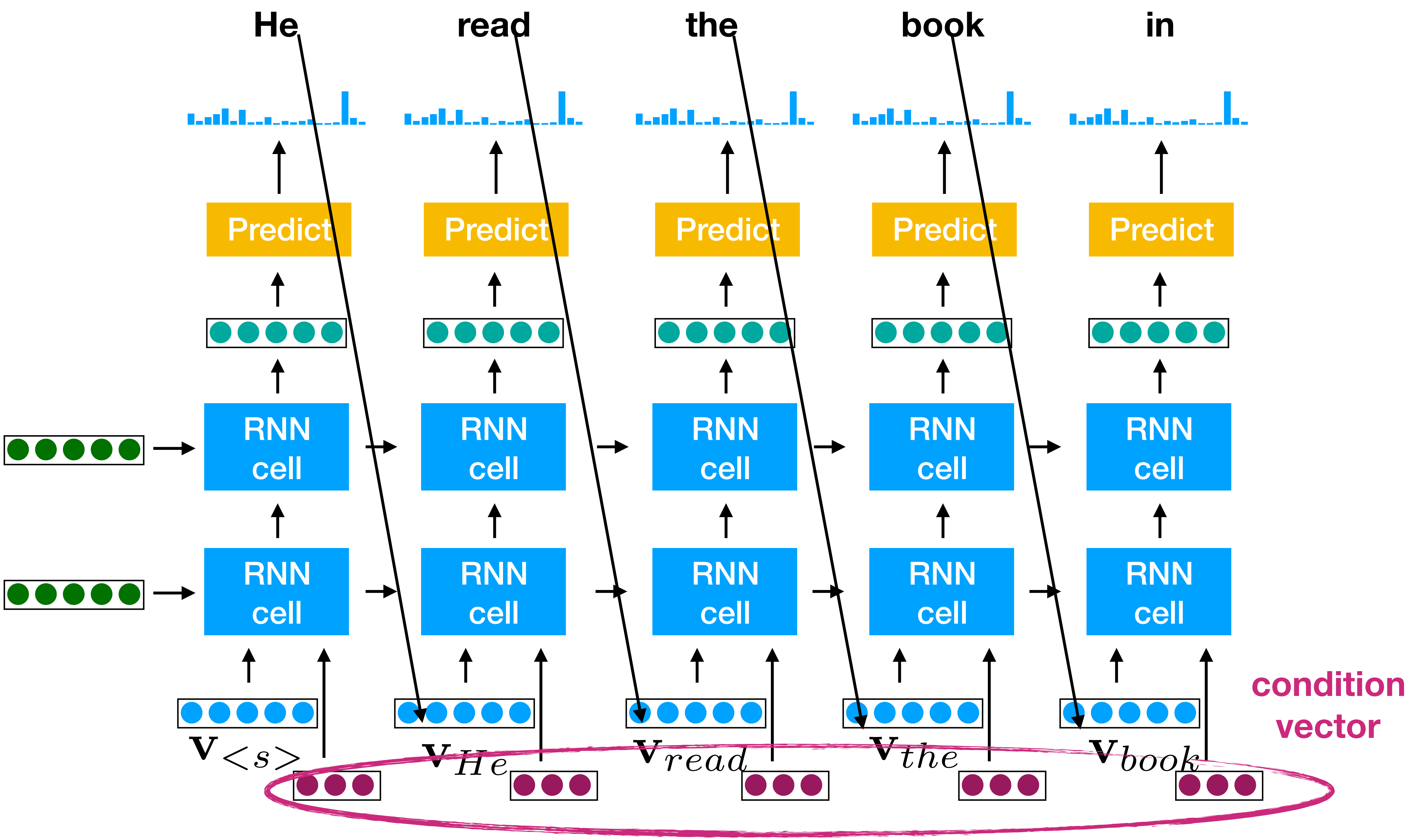


He
read
the
book
in
Predict
RNN cell
$\mathbf{v}_{<s>}$
$\mathbf{v}_{He}$
$\mathbf{v}_{read}$
$\mathbf{v}_{the}$
$\mathbf{v}_{book}$

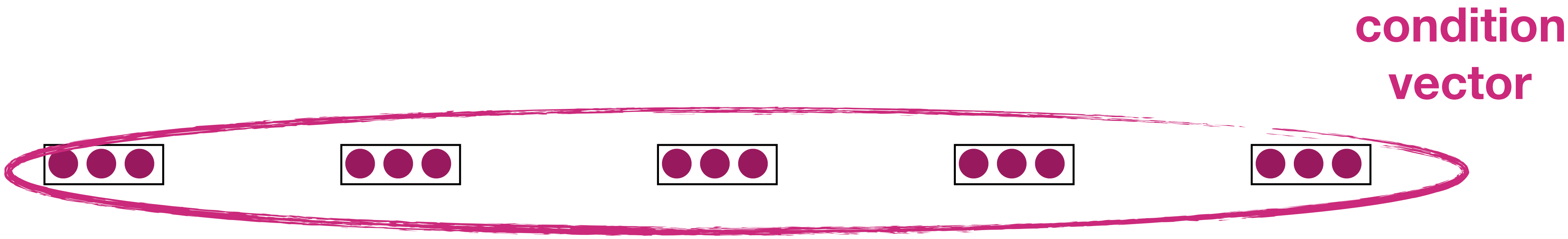
Conditioned Generation



Conditioned Generation

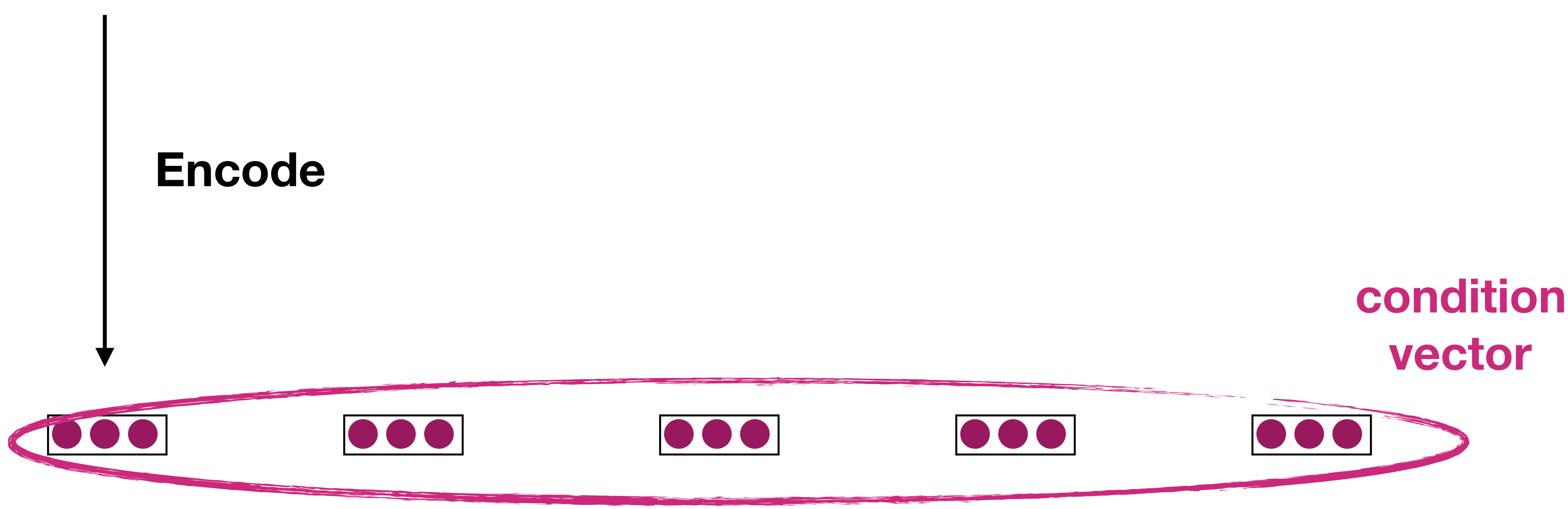


Conditioned Generation

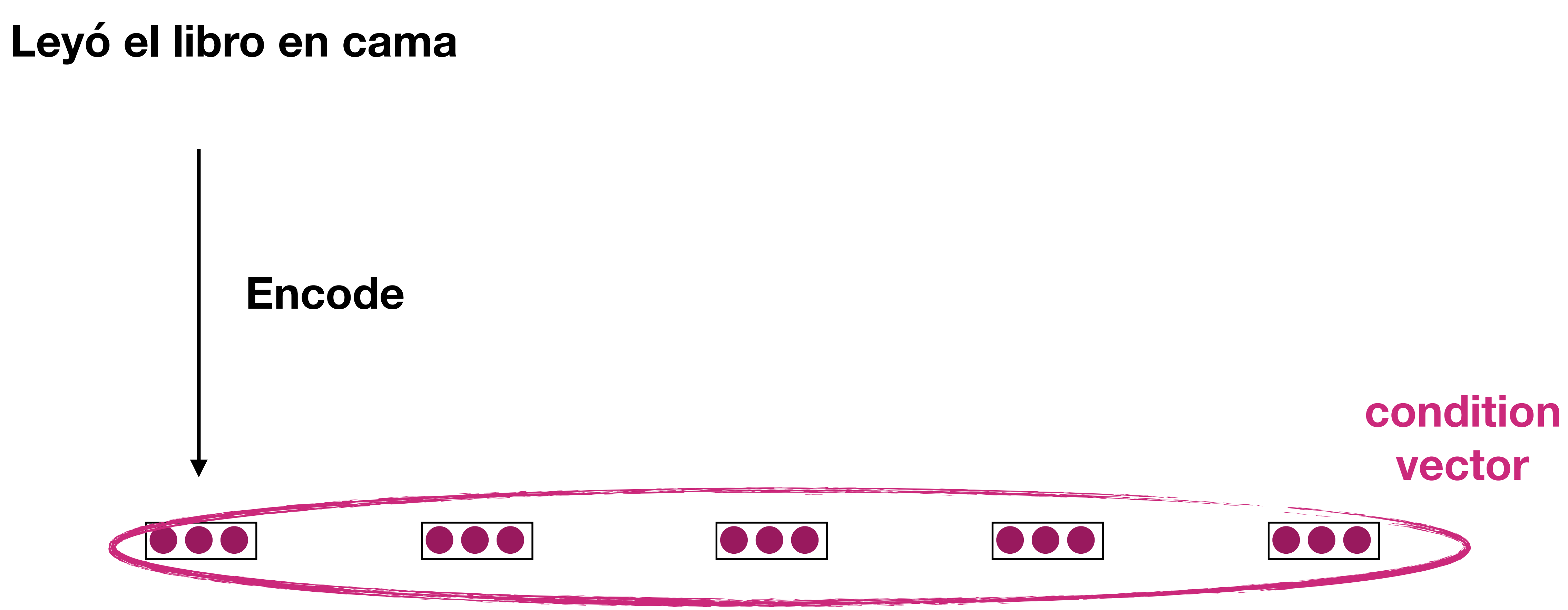


Conditioned Generation

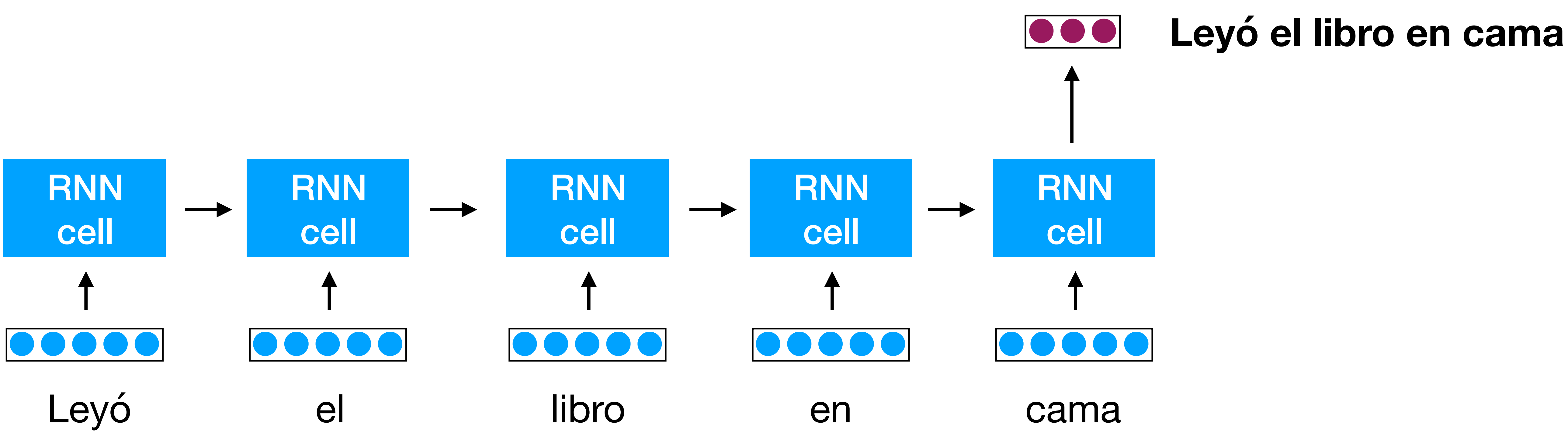
Name	Triton 52
EcoRating	A+
Family	L7



Conditioned Generation

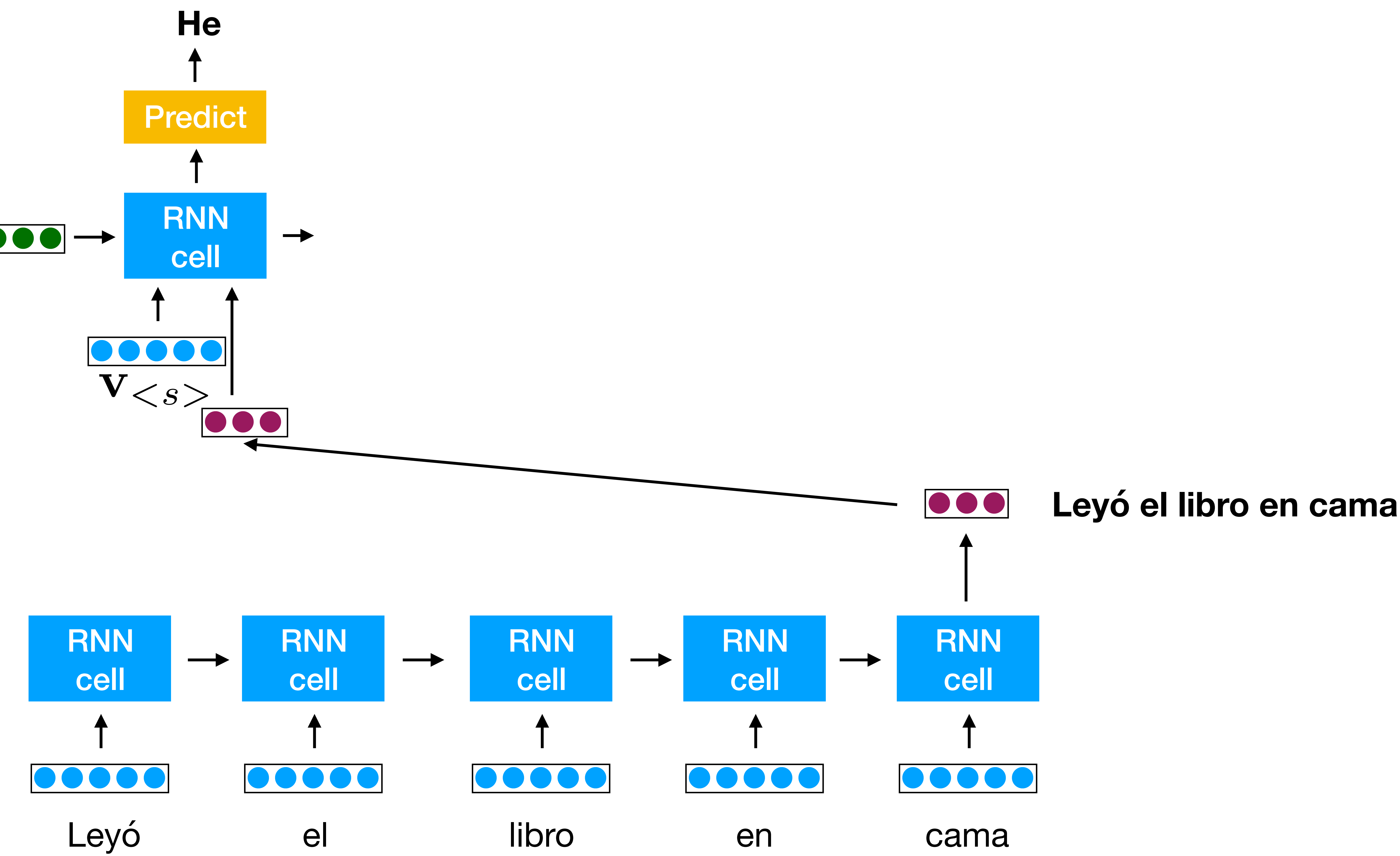


Seq2Seq

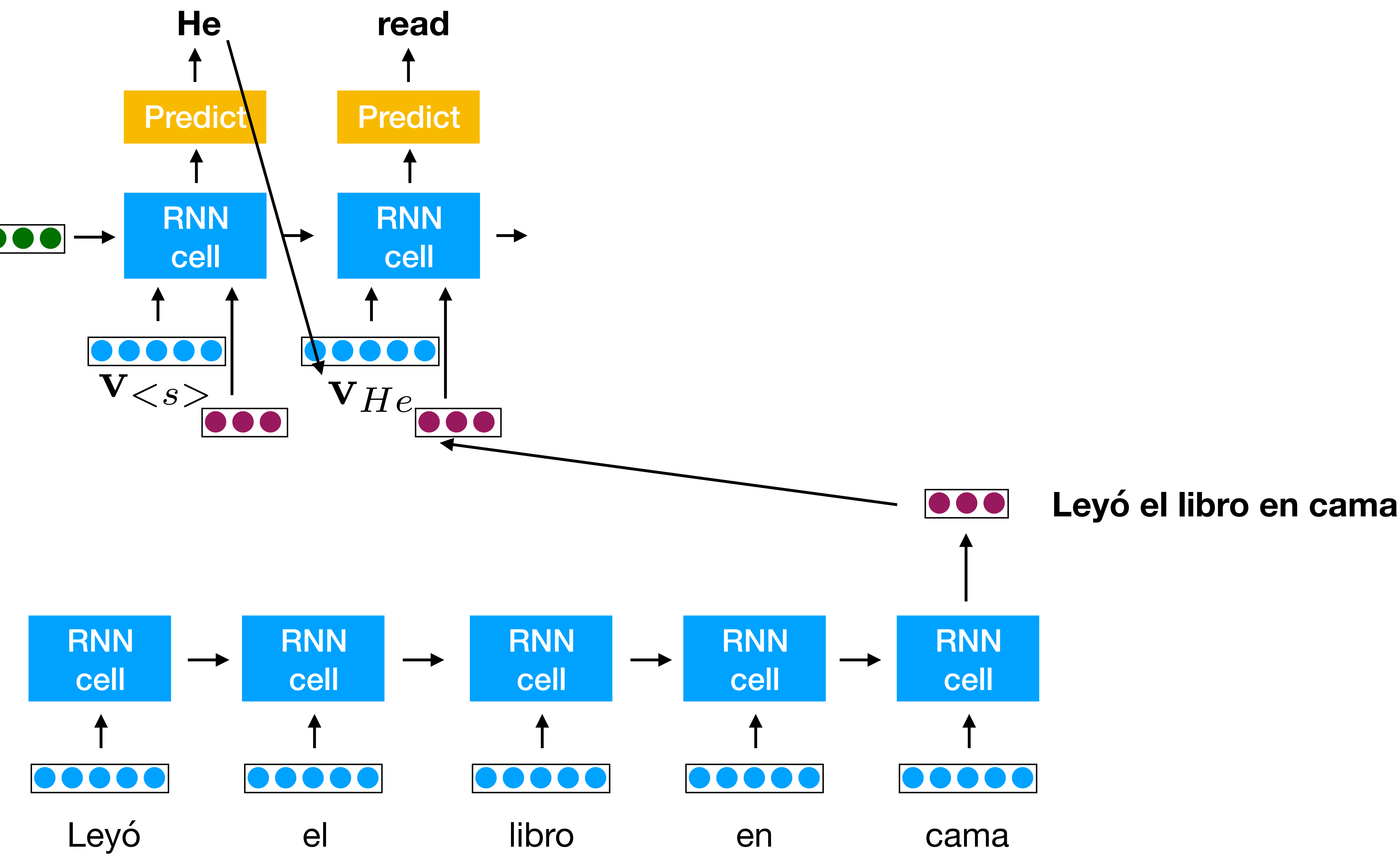


Leyó el libro en cama
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
Leyó
el
libro
en
cama

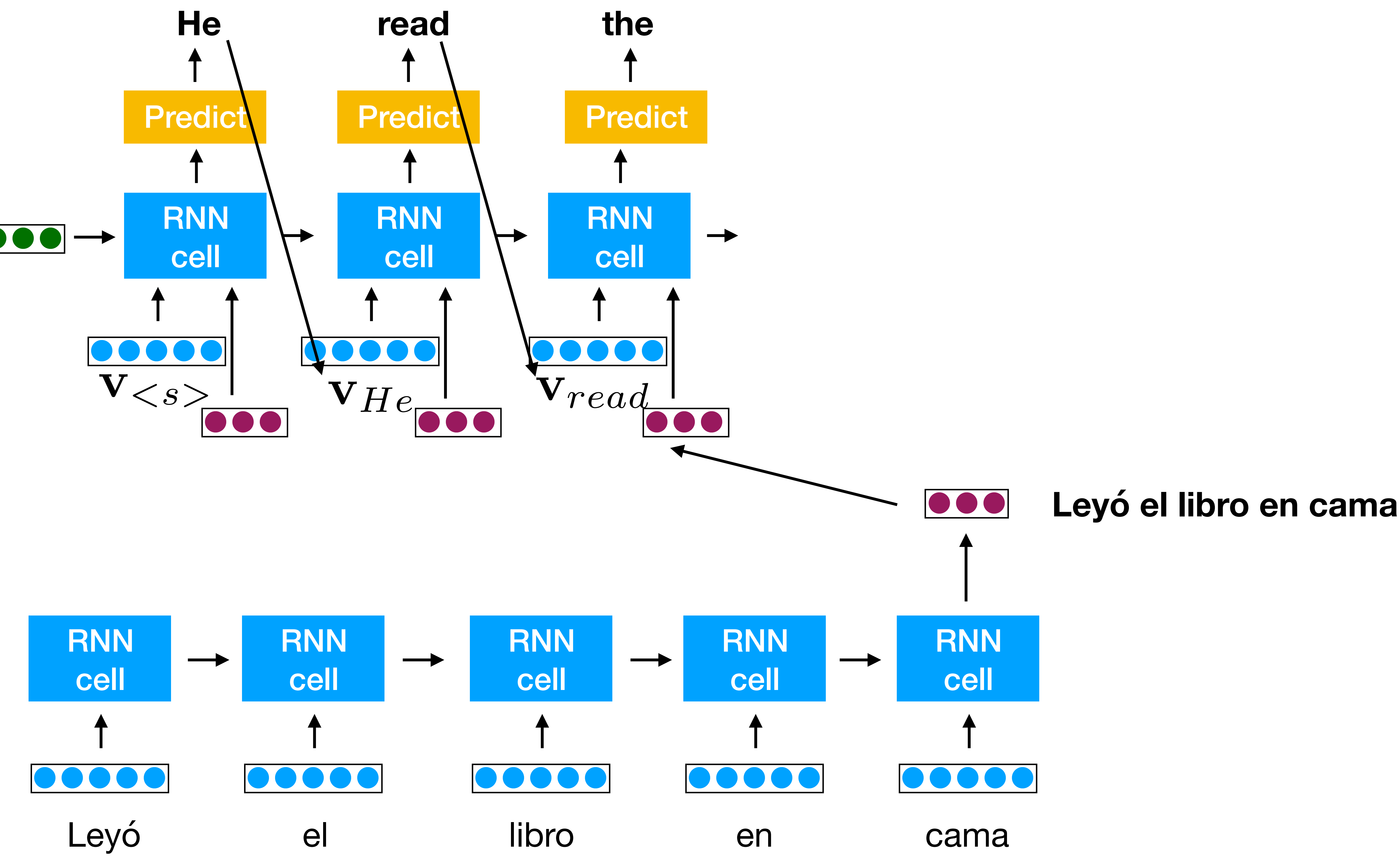
Seq2Seq


He
Predict
RNN cell
$\mathbf{v}_{<s>}$
Leyó el libro en cama
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
Leyó
el
libro
en
cama

Seq2Seq

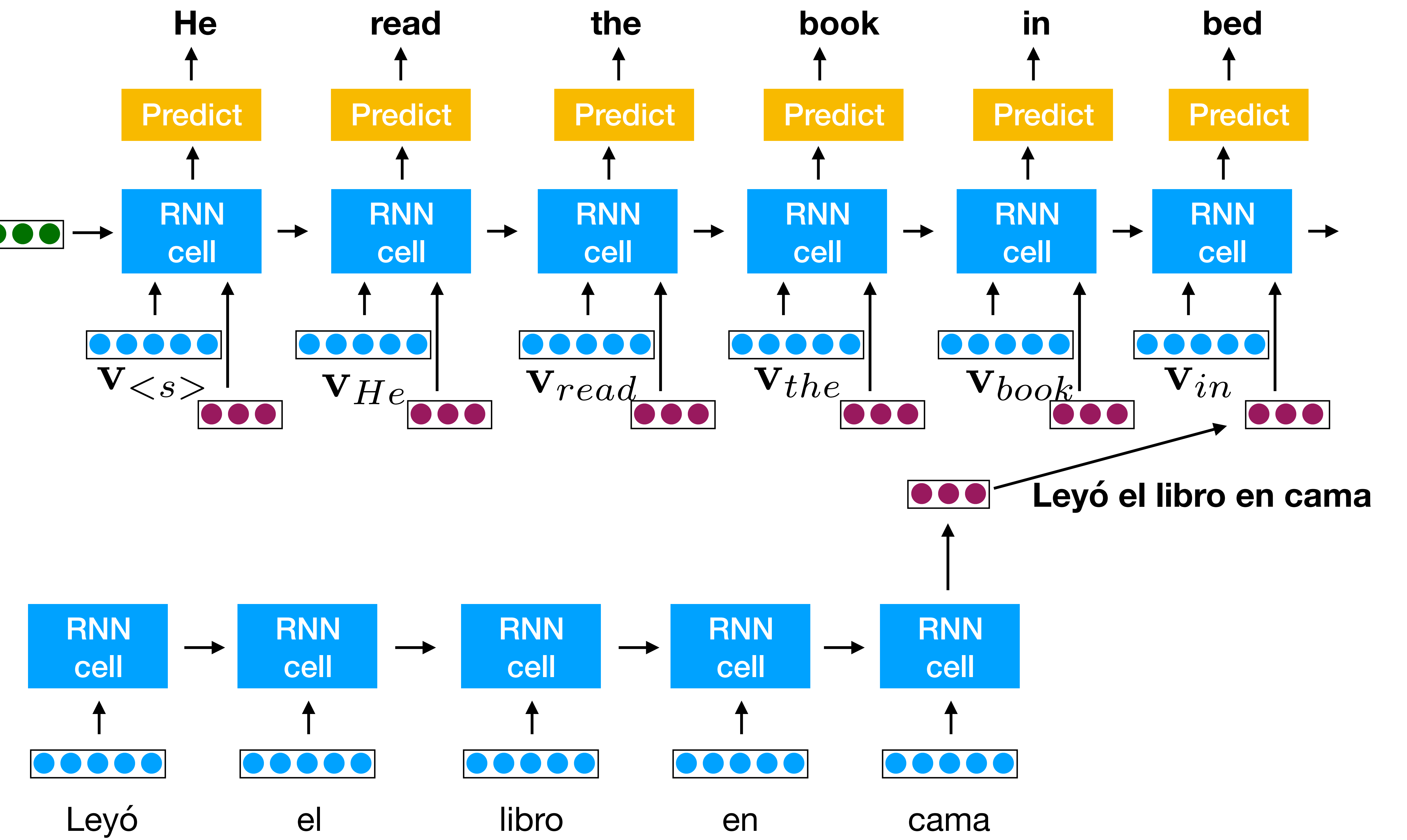

He
read
Predict
Predict
RNN cell
RNN cell
$\mathbf{v}_{<s>}$
$\mathbf{v}_{He}$
Leyó el libro en cama
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
Leyó
el
libro
en
cama

Seq2Seq



He
read
the
Predict
Predict
Predict
RNN cell
RNN cell
RNN cell
$\mathbf{v}_{<s>}$
$\mathbf{v}_{He}$
$\mathbf{v}_{read}$
Leyó el libro en cama
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
Leyó
el
libro
en
cama

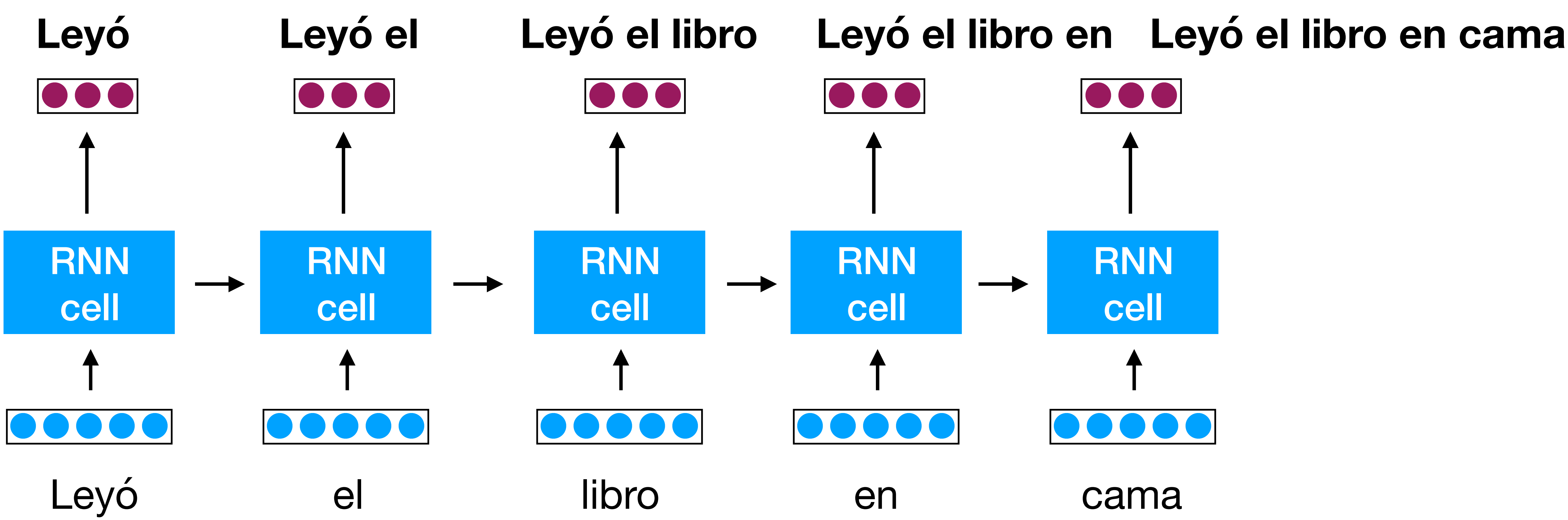
Seq2Seq



He
read
the
book
in
bed
Predict
RNN cell
$\mathbf{v}_{<s>}$
$\mathbf{v}_{He}$
$\mathbf{v}_{read}$
$\mathbf{v}_{the}$
$\mathbf{v}_{book}$
$\mathbf{v}_{in}$
Leyó el libro en cama
Leyó
el
libro
en
cama

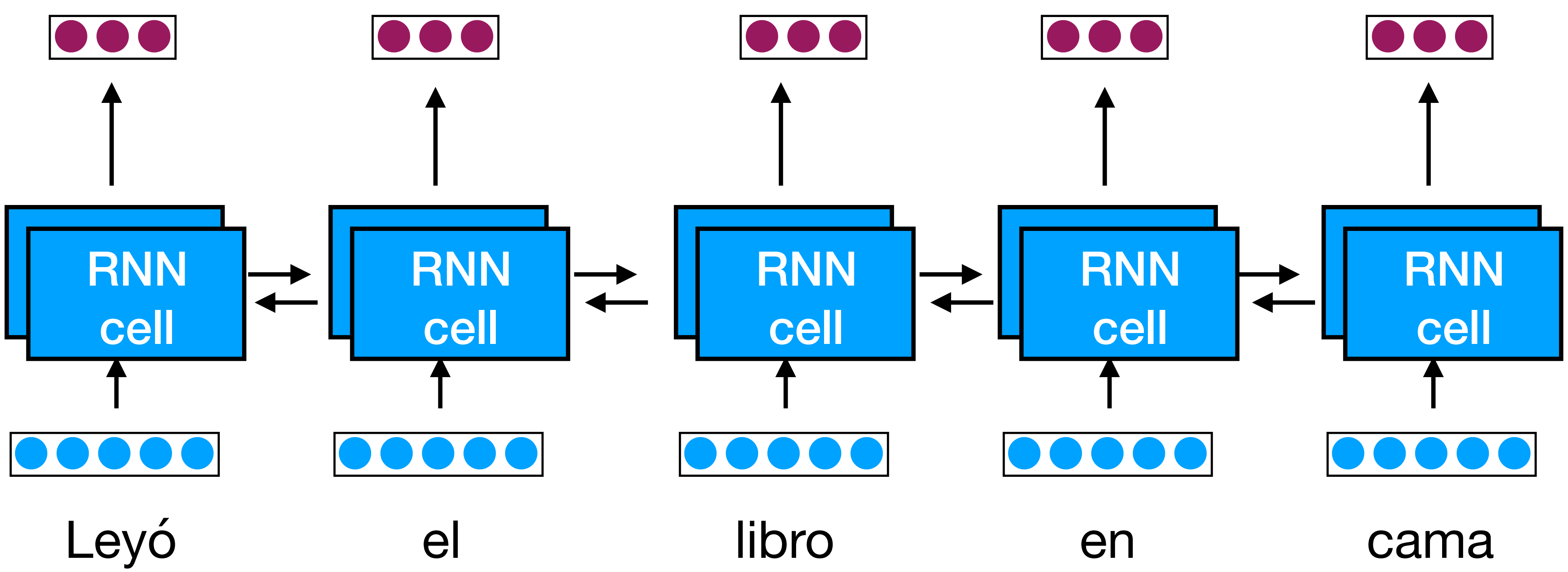
Seq2Seq + Attention

keep intermediate vectors



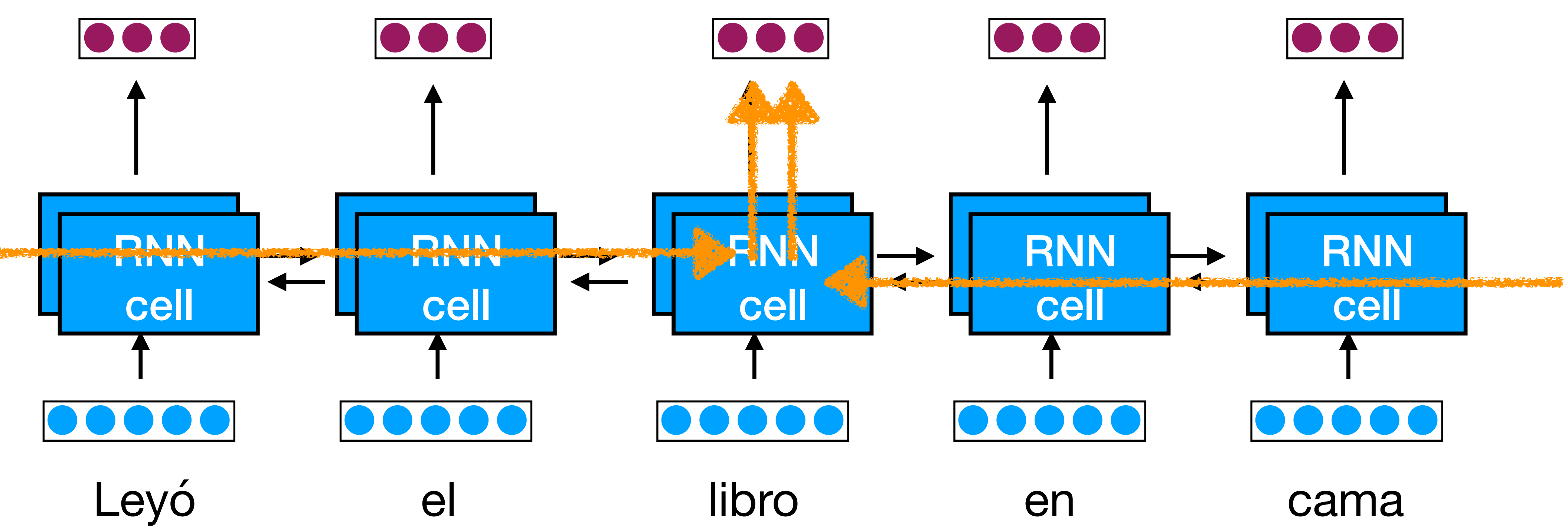
Seq2Seq + Attention

**add right-to-left RNN
(bi-RNN)**



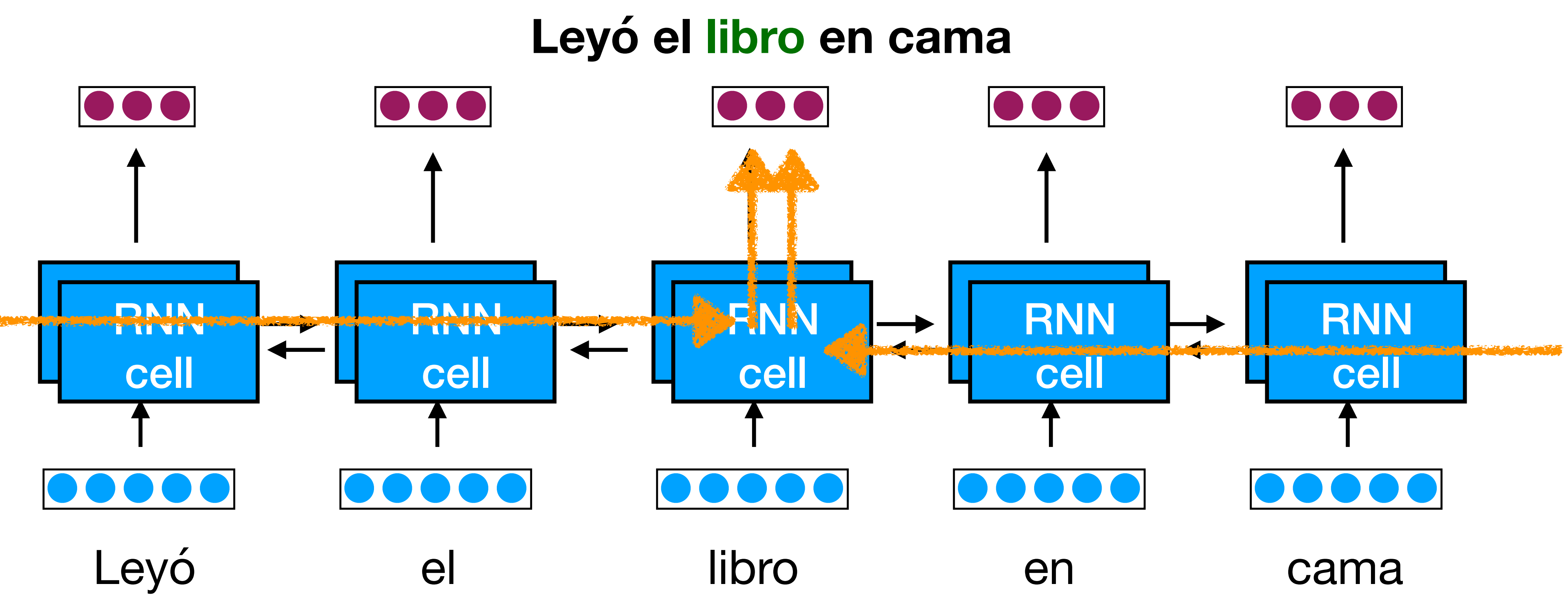
Seq2Seq + Attention

add right-to-left RNN (bi-RNN)



Seq2Seq + Attention

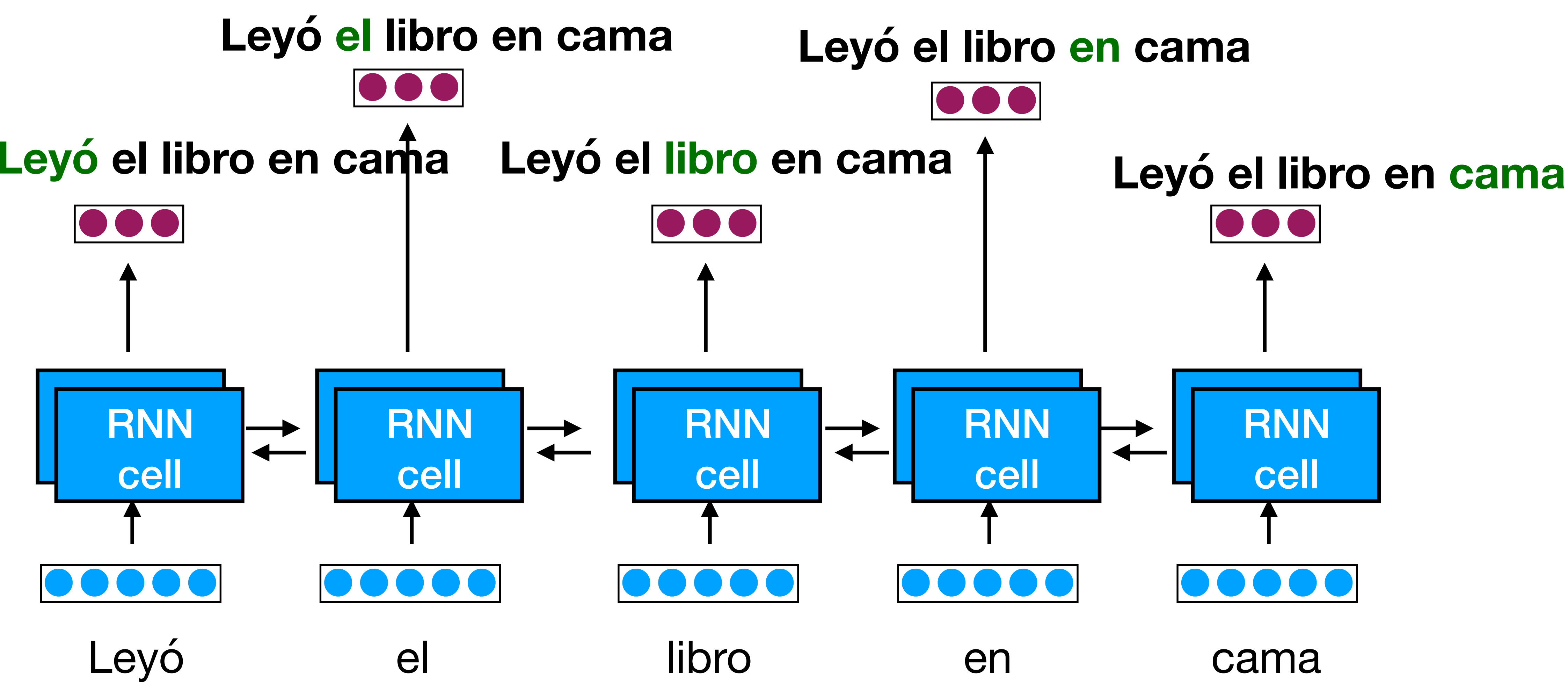
add right-to-left RNN
(bi-RNN)



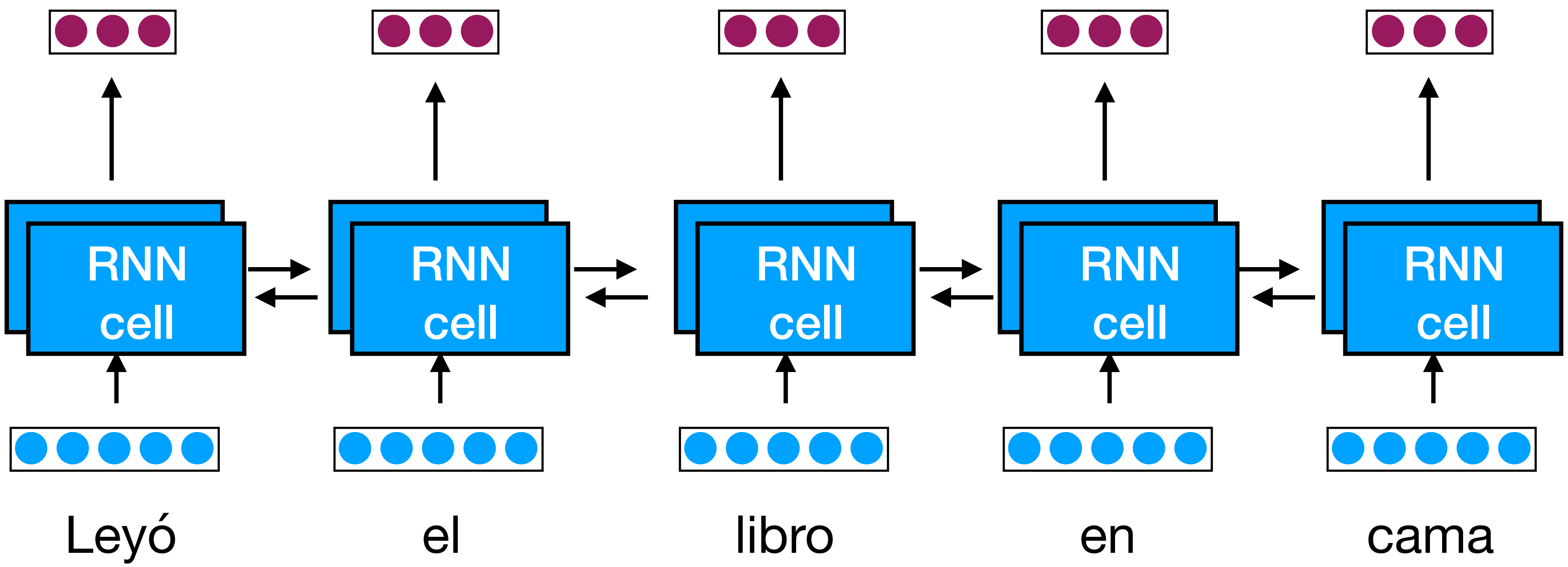
Seq2Seq + Attention

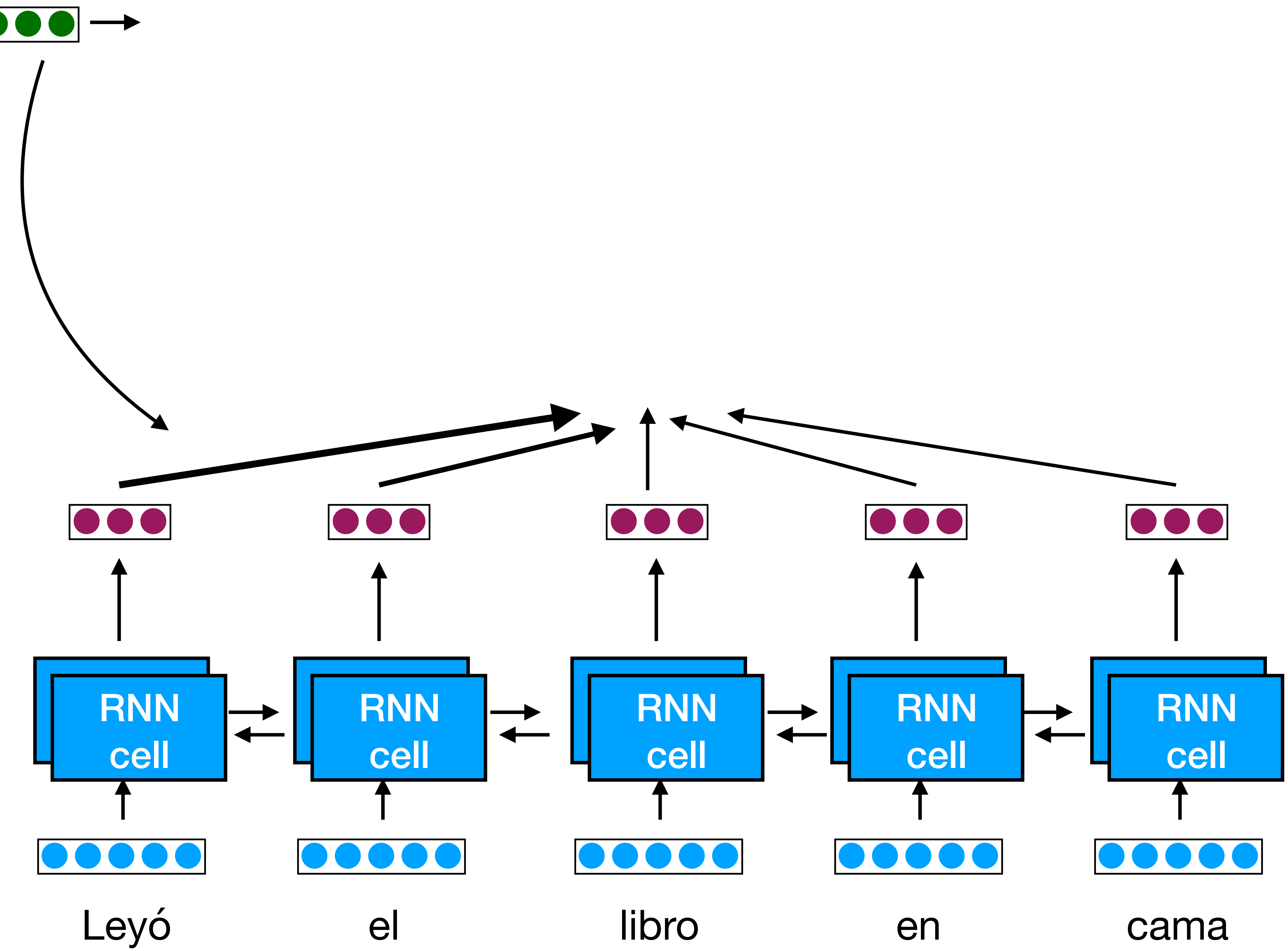
a representation of a word in context.

add right-to-left RNN
(bi-RNN)

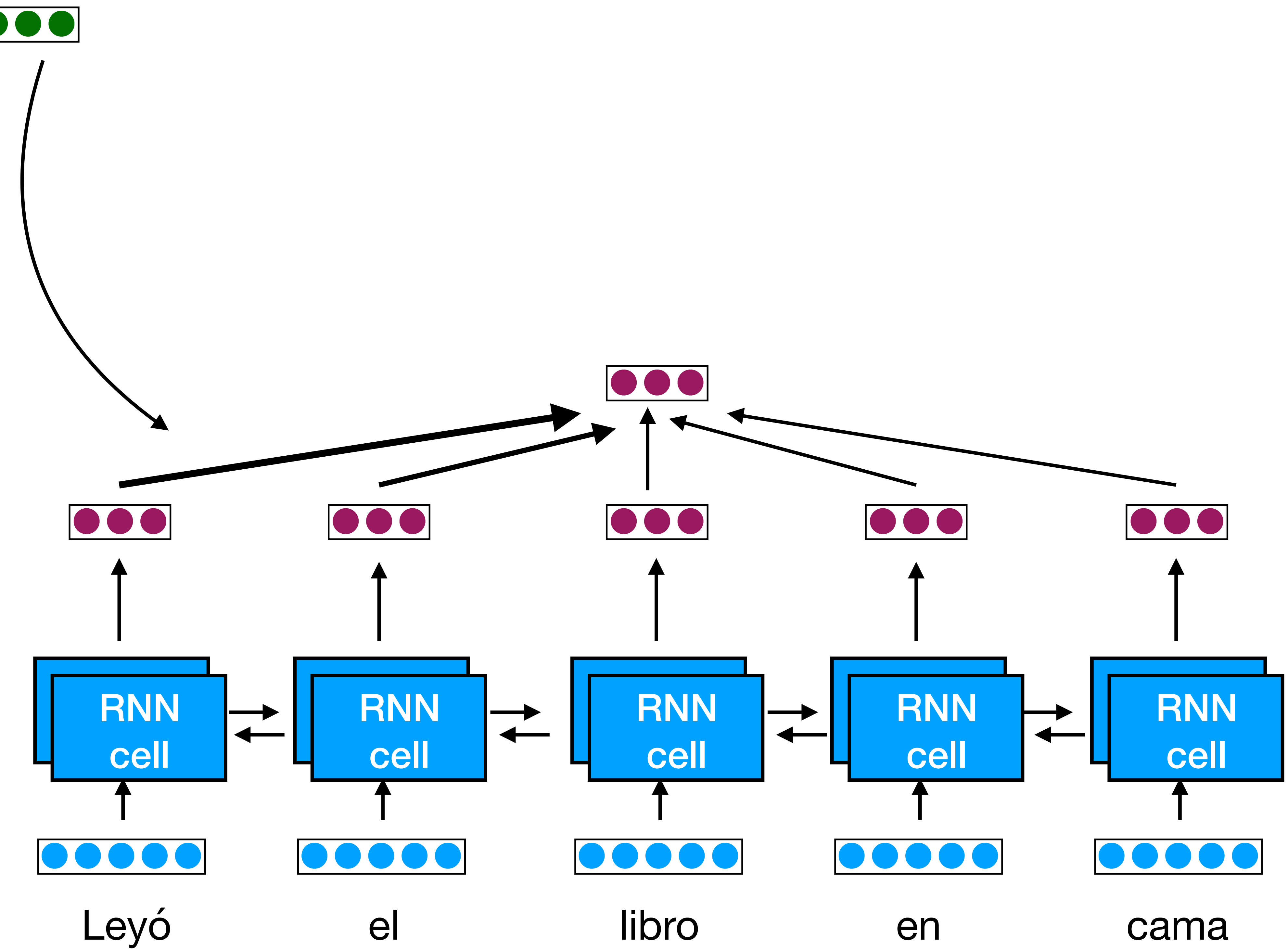


Seq2Seq + Attention

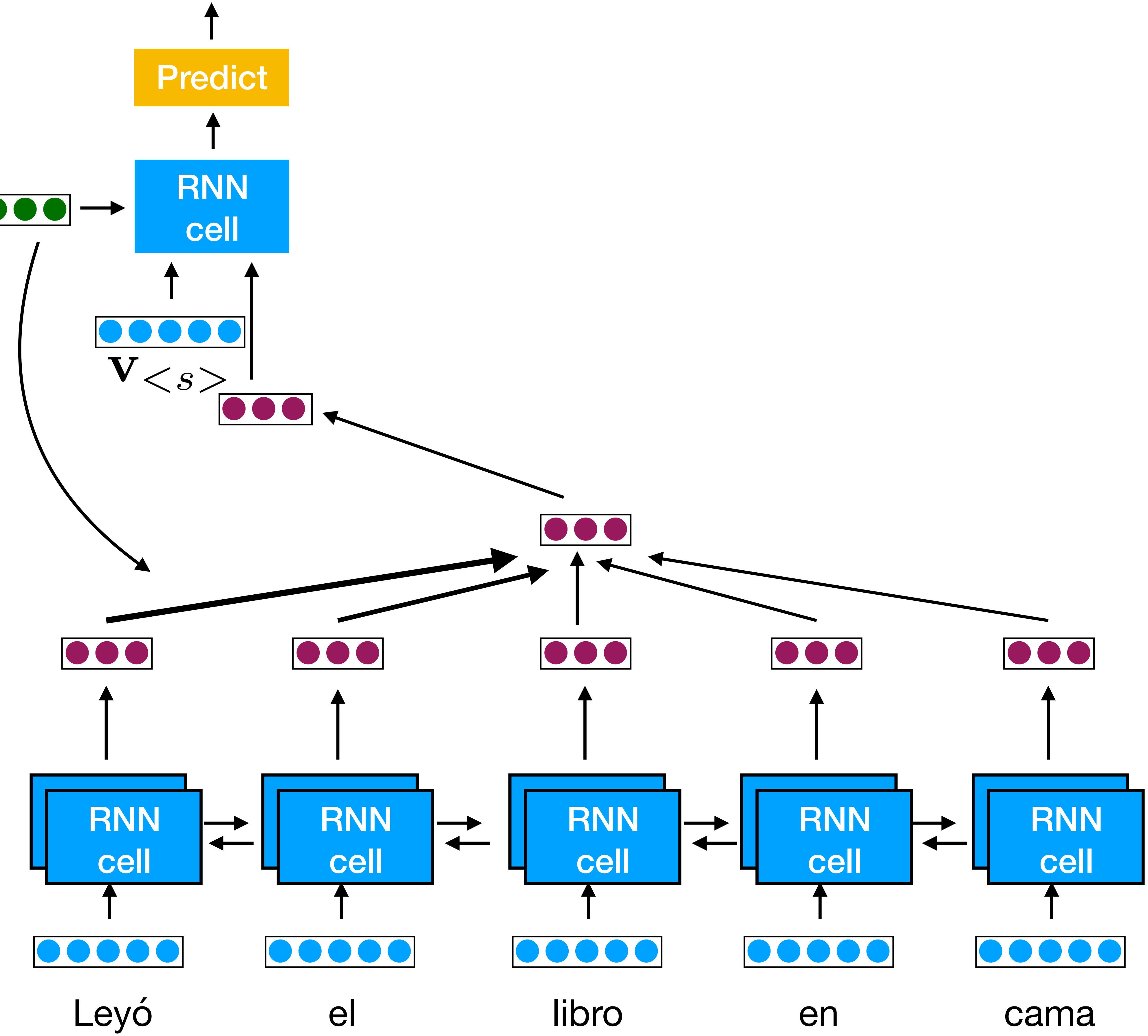




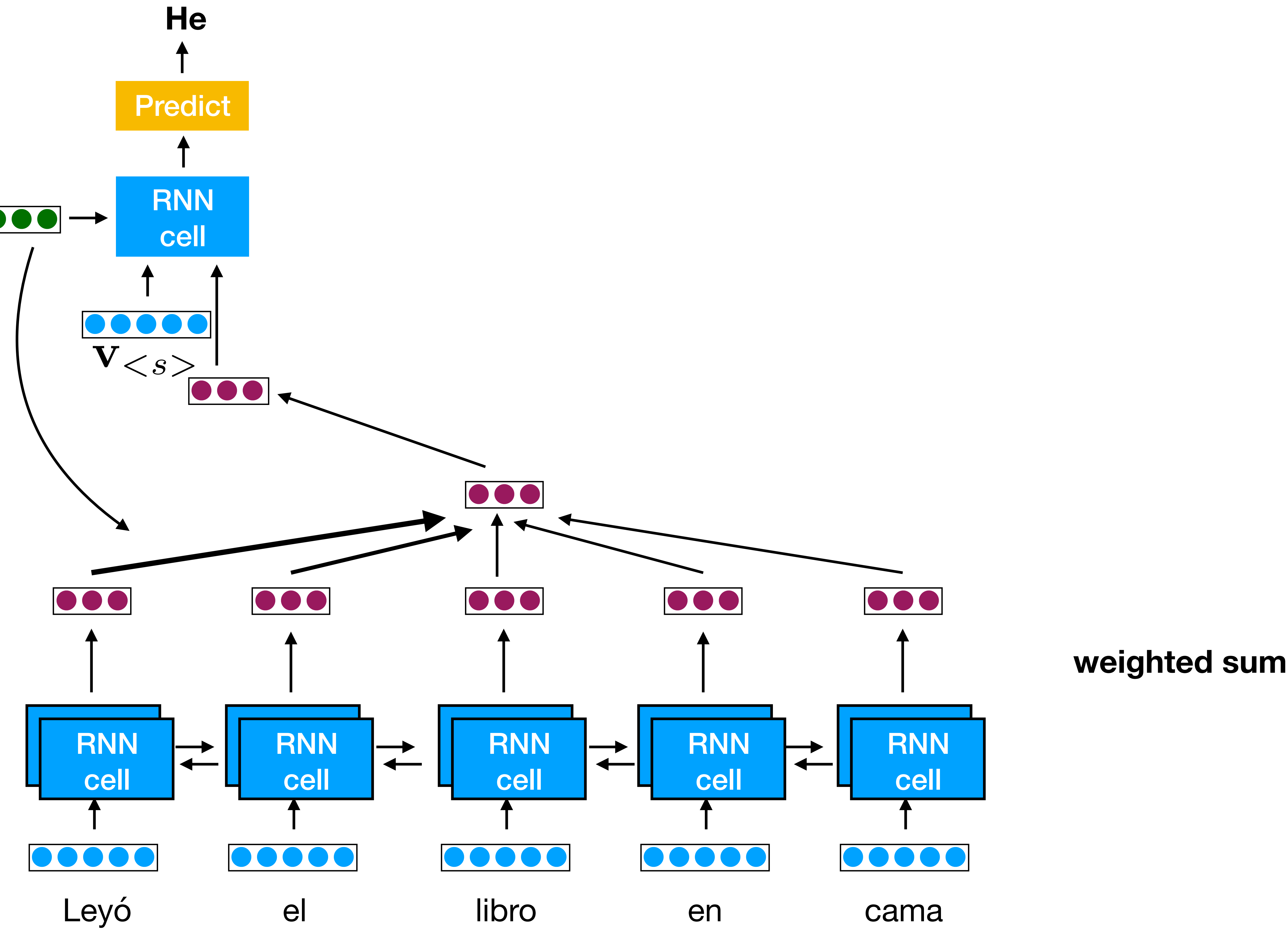
weighted sum

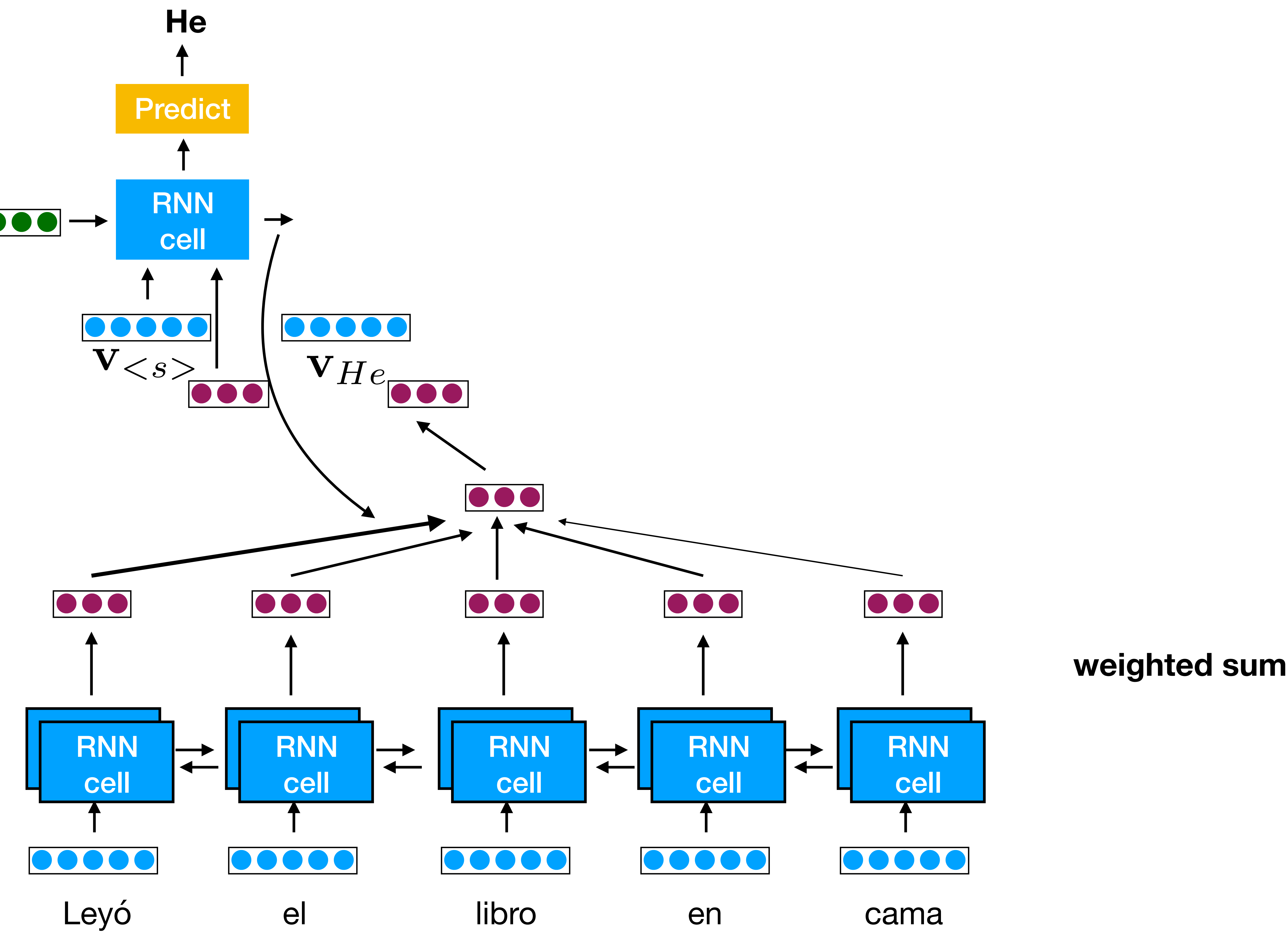


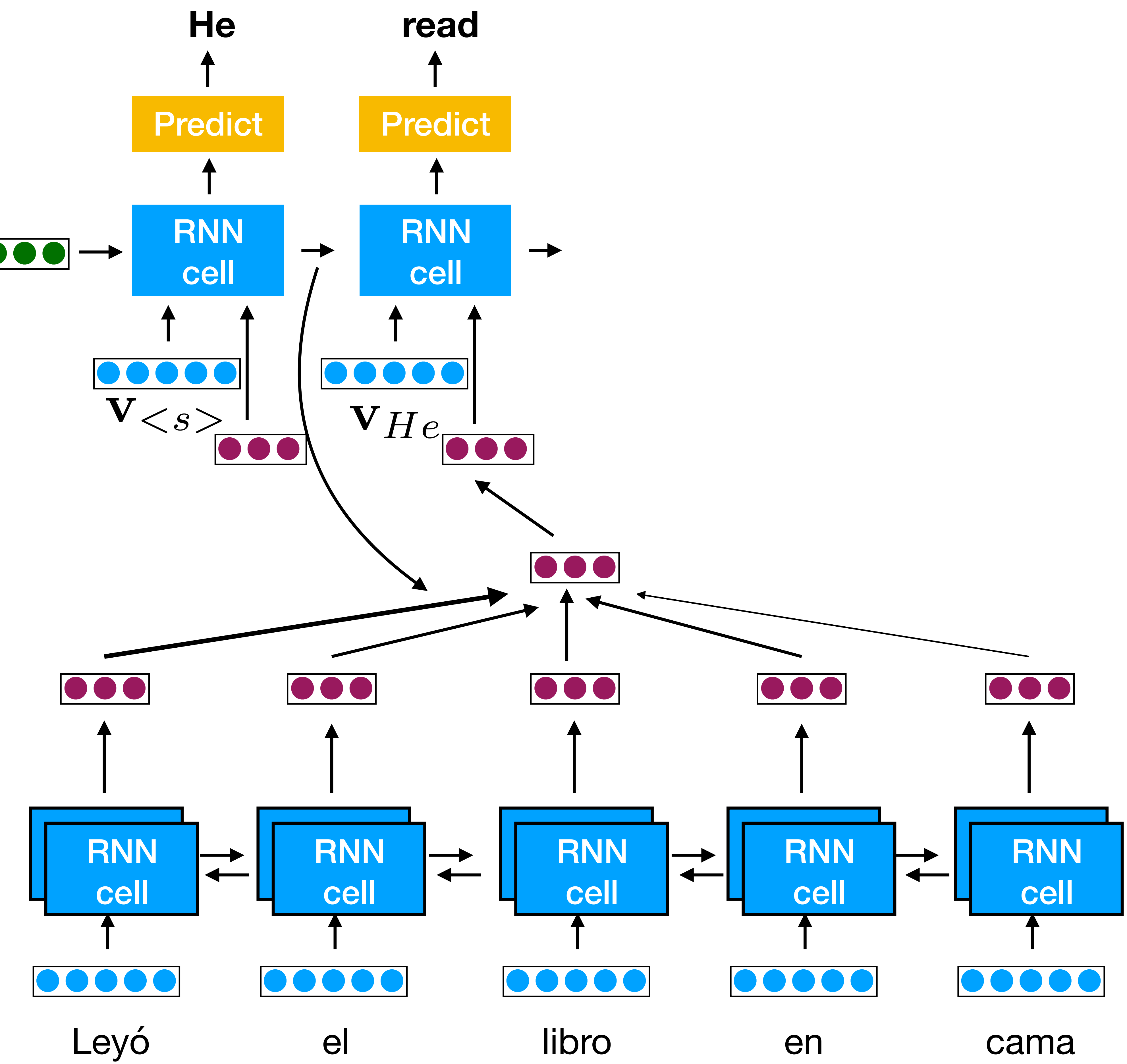
weighted sum



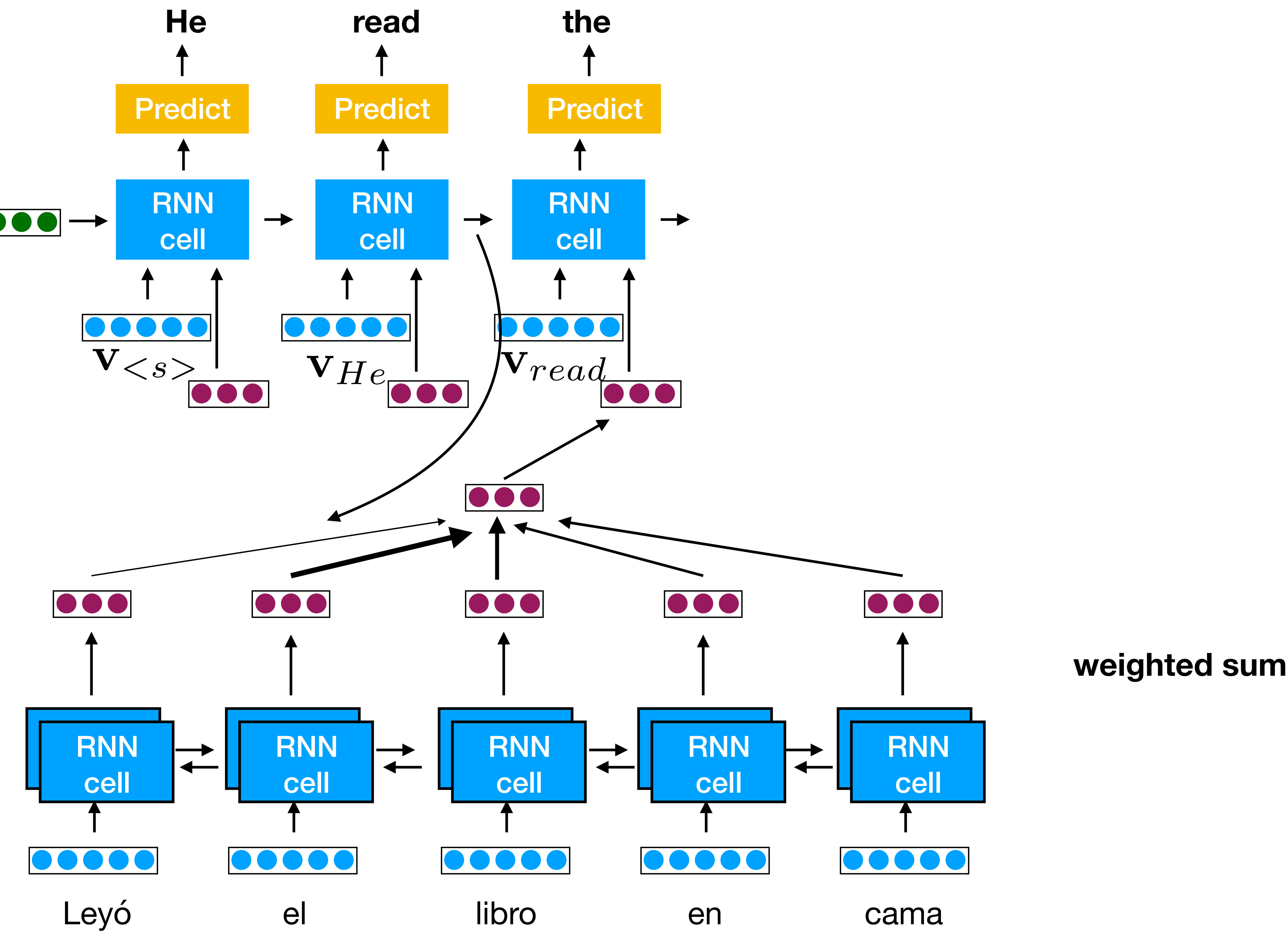
weighted sum

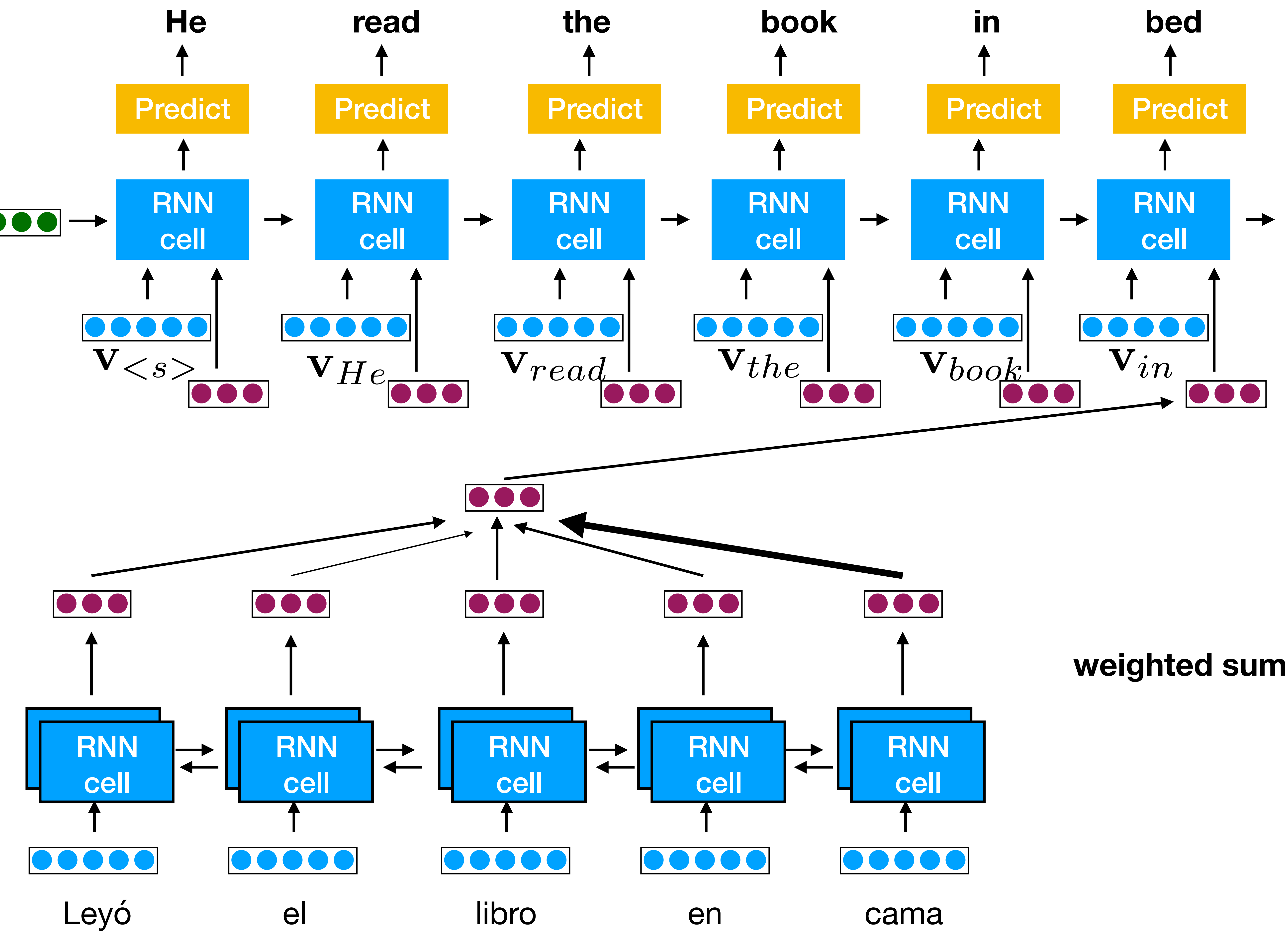

He
Predict
RNN cell
$\mathbf{v}_{<s>}$
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
Leyó
el
libro
en
cama
weighted sum


He
Predict
RNN cell
$\mathbf{v}_{<s>}$
$\mathbf{v}_{He}$
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
Leyó
el
libro
en
cama
weighted sum



weighted sum


He
read
the
Predict
Predict
Predict
RNN cell
RNN cell
RNN cell
$\mathbf{v}_{<s>}$
$\mathbf{v}_{He}$
$\mathbf{v}_{read}$
weighted sum
RNN cell
RNN cell
RNN cell
RNN cell
RNN cell
Leyó
el
libro
en
cama



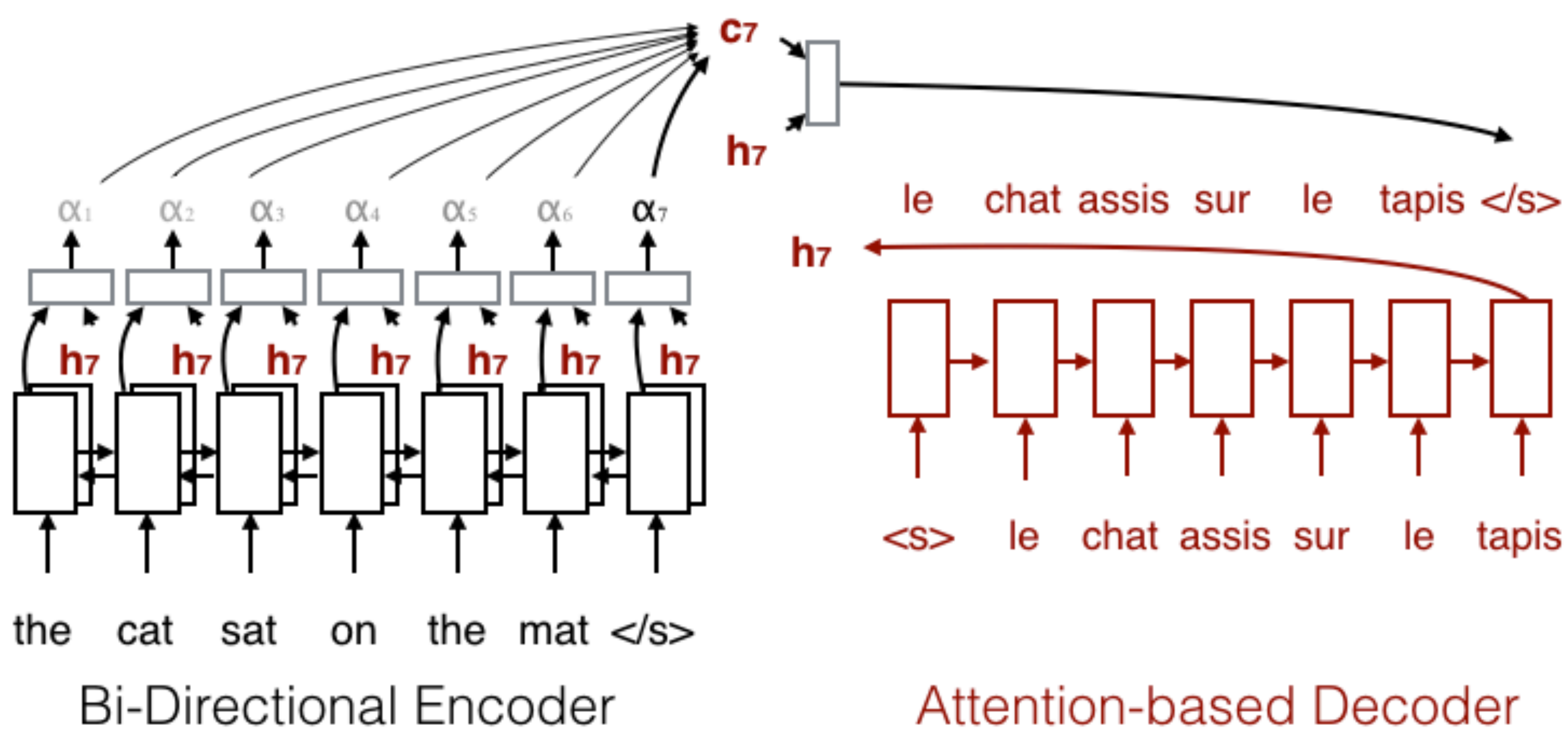
He
read
the
book
in
bed
Predict
RNN cell
$\mathbf{v}_{<s>}$
$\mathbf{v}_{He}$
$\mathbf{v}_{read}$
$\mathbf{v}_{the}$
$\mathbf{v}_{book}$
$\mathbf{v}_{in}$
weighted sum
Leyó
el
libro
en
cama

Seq2Seq + Attention

the conditioning vector is dynamically computed at each stage based on the current decoder hidden state.

Seq2Seq + Attention

the conditioning vector is dynamically computed at each stage based on the current decoder hidden state.



Findings of the E2E NLG Challenge

Ondřej Dušek, Jekaterina Novikova and **Verena Rieser**
The Interaction Lab, School of Mathematical and Computer Sciences
Heriot-Watt University
Edinburgh, Scotland, UK
`{o.dusek, j.novikova, v.t.rieser}@hw.ac.uk`

System	BLEU	NIST	METEOR	ROUGE-L	CIDEr	norm. avg.
♡TGEN baseline (Novikova et al., 2017b): seq2seq with MR classifier reranking	0.6593	8.6094	0.4483	0.6850	2.2338	0.5754
♡SLUG (Juraska et al., 2018): seq2seq-based ensemble (LSTM/CNN encoders, LSTM decoder), heuristic slot aligner reranking, data augmentation	**0.6619**	**8.6130**	0.4454	0.6772	**2.2615**	0.5744
♡TNT1 (Oraby et al., 2018): TGEN with data augmentation	0.6561	8.5105	**0.4517**	0.6839	2.2183	0.5729
♡NLE (Agarwal et al., 2018): fully lexicalised character-based seq2seq with MR classification reranking	0.6534	8.5300	0.4435	0.6829	2.1539	0.5696
♡TNT2 (Tandon et al., 2018): TGEN with data augmentation	0.6502	8.5211	0.4396	**0.6853**	2.1670	0.5688
♡HARV (Gehrmann et al., 2018): fully lexicalised seq2seq with copy mechanism, coverage penalty reranking, diverse ensembling	0.6496	8.5268	0.4386	**0.6872**	2.0850	0.5673
♡ZHANG (Zhang et al., 2018): fully lexicalised seq2seq over subword units, attention memory	0.6545	8.1840	0.4392	**0.7083**	2.1012	0.5661
♡GONG (Gong, 2018): TGEN fine-tuned using reinforcement learning	0.6422	8.3453	0.4469	0.6645	**2.2721**	0.5631
♡TR1 (Schilder et al., 2018): seq2seq with stronger delexicalization (incl. *priceRange* and *customerRating*)	0.6336	8.1848	0.4322	0.6828	2.1425	0.5563
◇SHEFF1 (Chen et al., 2018): 2-level linear classifiers deciding on next slot/token, trained using LOLS, training data filtering	0.6015	8.3075	0.4405	0.6778	2.1775	0.5537
♣DANGNT (Nguyen and Tran, 2018): rule-based two-step approach, selecting phrases for each slot + lexicalising	0.5990	7.9277	0.4346	0.6634	2.0783	0.5395
♡SLUG-ALT (*late submission*, Juraska et al., 2018): SLUG trained only using complex sentences from the training data	0.6035	8.3954	0.4369	0.5991	2.1019	0.5378
◇ZHAW2 (Deriu and Cieliebak, 2018): semantically conditioned LSTM RNN language model (Wen et al., 2015b) + controlling the first generated word	0.6004	8.1394	0.4388	0.6119	1.9188	0.5314
♠TUDA (Puzikov and Gurevych, 2018): handcrafted templates	0.5657	7.4544	**0.4529**	0.6614	1.8206	0.5215
◇ZHAW1 (Deriu and Cieliebak, 2018): ZHAW2 with MR classification loss + reranking	0.5864	8.0212	0.4322	0.5998	1.8173	0.5205
♡ADAPT (Elder et al., 2018): seq2seq with preprocessing that enriches the MR with desired target words	0.5092	7.1954	0.4025	0.5872	1.5039	0.4738
♡CHEN (Chen, 2018): fully lexicalised seq2seq with copy mechanism and attention memory	0.5859	5.4383	0.3836	0.6714	1.5790	0.4685
♠FORGE3 (Mille and Dasiopoulou, 2018): templates mined from training data	0.4599	7.1092	0.3858	0.5611	1.5586	0.4547
♡SHEFF2 (Chen et al., 2018): vanilla seq2seq	0.5436	5.7462	0.3561	0.6152	1.4130	0.4462
♠TR2 (Schilder et al., 2018): templates mined from training data	0.4202	6.7686	0.3968	0.5481	1.4389	0.4372
♣FORGE1 (Mille and Dasiopoulou, 2018): grammar-based	0.4207	6.5139	0.3685	0.5437	1.3106	0.4231

System	BLEU	NIST	METEOR	ROUGE-L	CIDEr	norm. avg.
♡TGEN baseline (Novikova et al., 2017b): seq2seq with MR classifier reranking	0.6593	8.6094	0.4483	0.6850	2.2338	0.5754
♡SLUG (Juraska et al., 2018): seq2seq-based ensemble (LSTM/CNN encoders, LSTM decoder), heuristic slot aligner reranking, data augmentation	**0.6619**	**8.6130**	0.4454	0.6772	**2.2615**	0.5744
♡TNT1 (Oraby et al., 2018): TGEN with data augmentation	0.6561	8.5105	**0.4517**	0.6839	2.2183	0.5729
♡NLE (Agarwal et al., 2018): fully lexicalised character-based seq2seq with MR classification reranking	0.6534	8.5300	0.4435	0.6829	2.1539	0.5696
♡TNT2 (Tandon et al., 2018): TGEN with data augmentation	0.6502	8.5211	0.4396	**0.6853**	2.1670	0.5688
♡HARV (Gehrmann et al., 2018): fully lexicalised seq2seq with copy mechanism, coverage penalty reranking, diverse ensembling	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
♡ZHANG (Zhang et al., 2018): fully lexicalised seq2seq over subword units, attention memory						
♡GONG (Gong, 2018): TGEN fine-tuned using reinforcement learning						
♡TR1 (Schilder et al., 2018): seq2seq with stronger delexicalization (incl. *priceRange* and *customerRating*)						
◇SHEFF1 (Chen et al., 2018): 2-level linear classifiers deciding on next slot/token, trained using LOLS, training data filtering						
♣DANGNT (Nguyen and Tran, 2018): rule-based two-step approach, selecting phrases for each slot + lexicalising						
♡SLUG-ALT (*late submission*, Juraska et al., 2018): SLUG trained only using complex sentences from the training data						
◇ZHAW2 (Deriu and Cieliebak, 2018): semantically conditioned LSTM RNN language model (Wen et al., 2015b) + controlling the first generated word	0.6004	8.1394	0.4388	0.6119	1.9188	0.5314
♠TUDA (Puzikov and Gurevych, 2018): handcrafted templates	0.5657	7.4544	**0.4529**	0.6614	1.8206	0.5215
◇ZHAW1 (Deriu and Cieliebak, 2018): ZHAW2 with MR classification loss + reranking	0.5864	8.0212	0.4322	0.5998	1.8173	0.5205
♡ADAPT (Elder et al., 2018): seq2seq with preprocessing that enriches the MR with desired target words	0.5092	7.1954	0.4025	0.5872	1.5039	0.4738
♡CHEN (Chen, 2018): fully lexicalised seq2seq with copy mechanism and attention memory	0.5859	5.4383	0.3836	0.6714	1.5790	0.4685
♠FORGE3 (Mille and Dasiopoulou, 2018): templates mined from training data	0.4599	7.1092	0.3858	0.5611	1.5586	0.4547
♡SHEFF2 (Chen et al., 2018): vanilla seq2seq	0.5436	5.7462	0.3561	0.6152	1.4130	0.4462
♠TR2 (Schilder et al., 2018): templates mined from training data	0.4202	6.7686	0.3968	0.5481	1.4389	0.4372
♣FORGE1 (Mille and Dasiopoulou, 2018): grammar-based	0.4207	6.5139	0.3685	0.5437	1.3106	0.4231

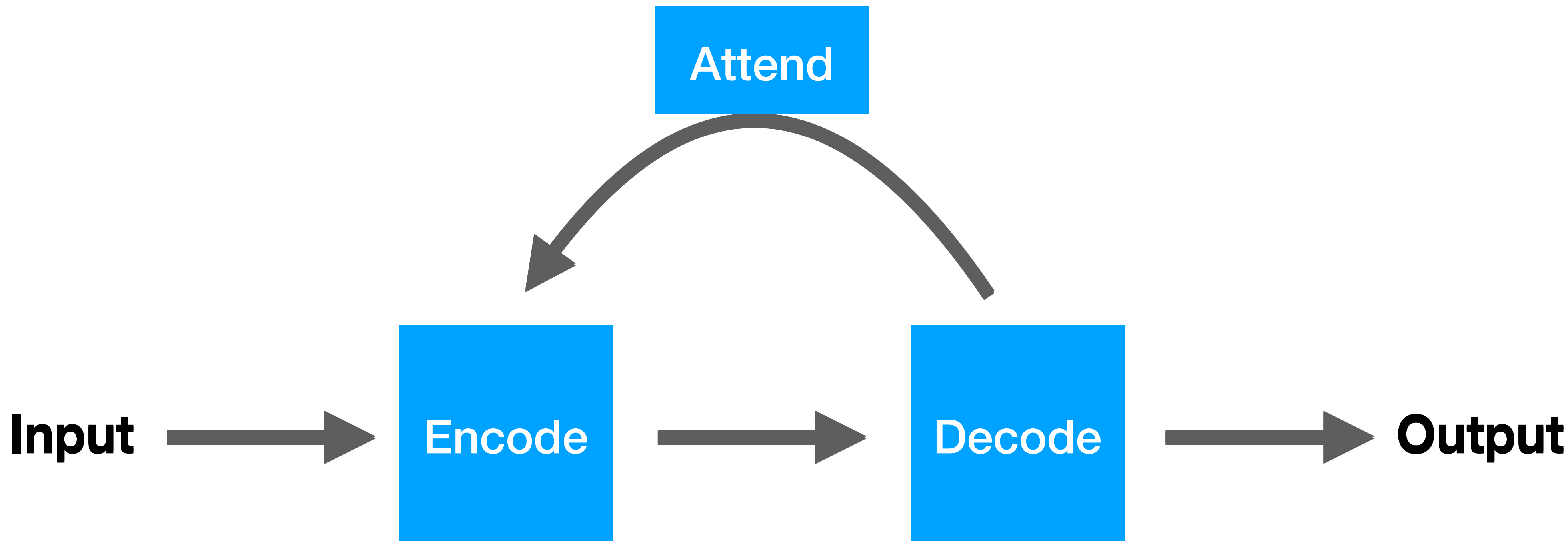
- Red: seq2seq+att
- Orange: almost seq2seq+att
- Green/Blue: patterns

3. The BiLSTM Hegemony

To a first approximation,
the de facto consensus in NLP in 2017 is
that no matter what the task,
you throw a BiLSTM at it, with
attention if you need information flow

Chris Manning
April 2017

The basic abstraction



sequence of symbols
from alphabet A

sequence of symbols
from alphabet B

The basic abstraction

Input → off-the-shelf Software → Output

sequence of symbols
from alphabet A

sequence of symbols
from alphabet B

Software

MARIANNMT

Fast Neural Machine Translation in C++

NMT

An open source neural machine translation system.

OpenNMT

pytorch / fairseq

What's not supported

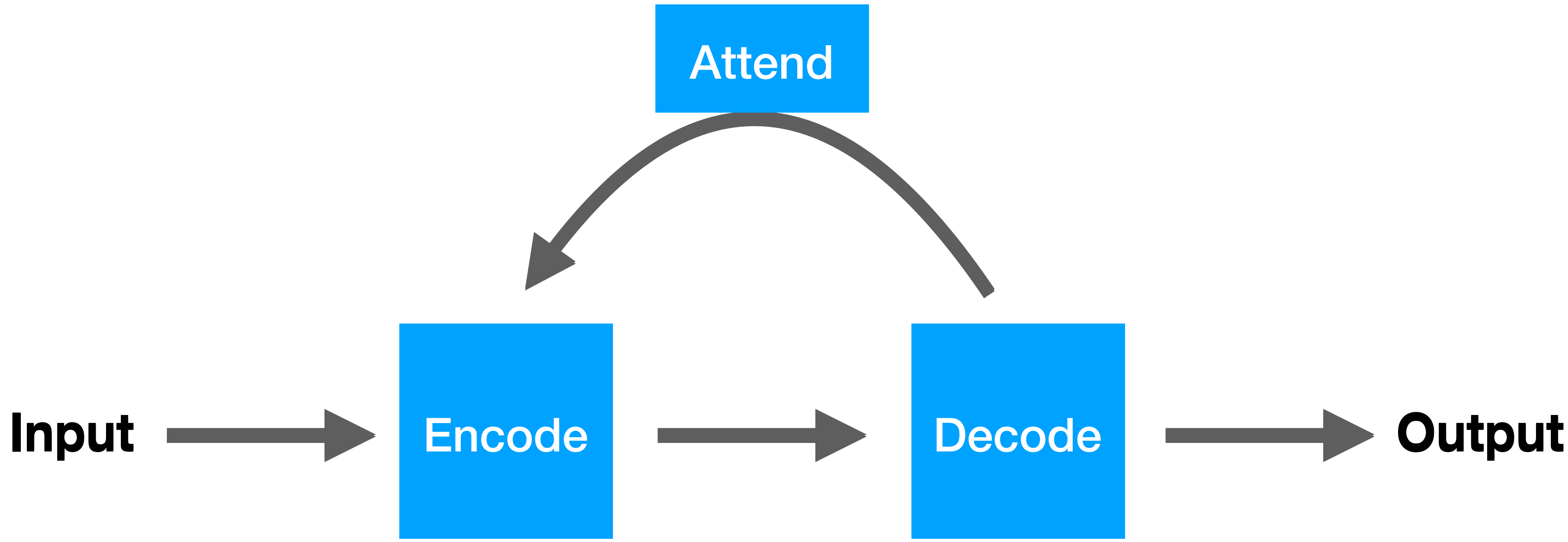
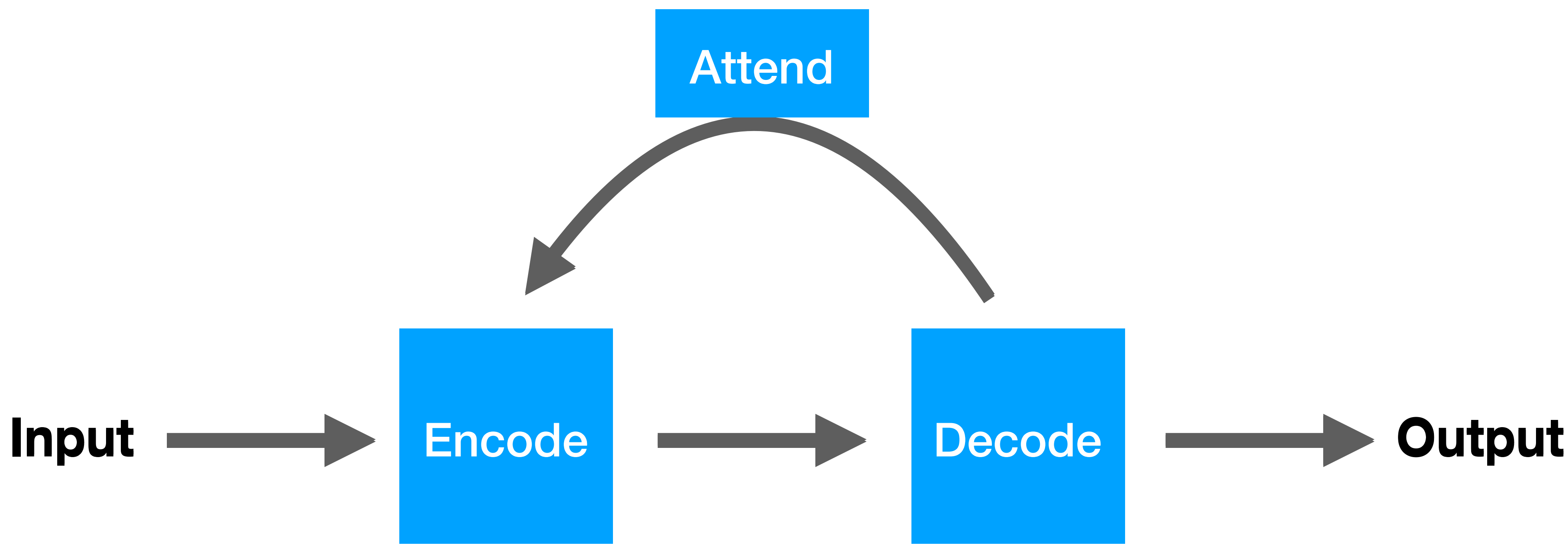


Image
Tree
Graph
Set of items
Key=value pairs
....

sequence of symbols
from alphabet B

Tree / Graph

What's not supported



Key=value pairs

sequence of symbols from alphabet B

Name	Triton 52
EcoRating	A+
Family	L7

→ **the Triton 52 has an A+ echo rating and is in the L7 product family**

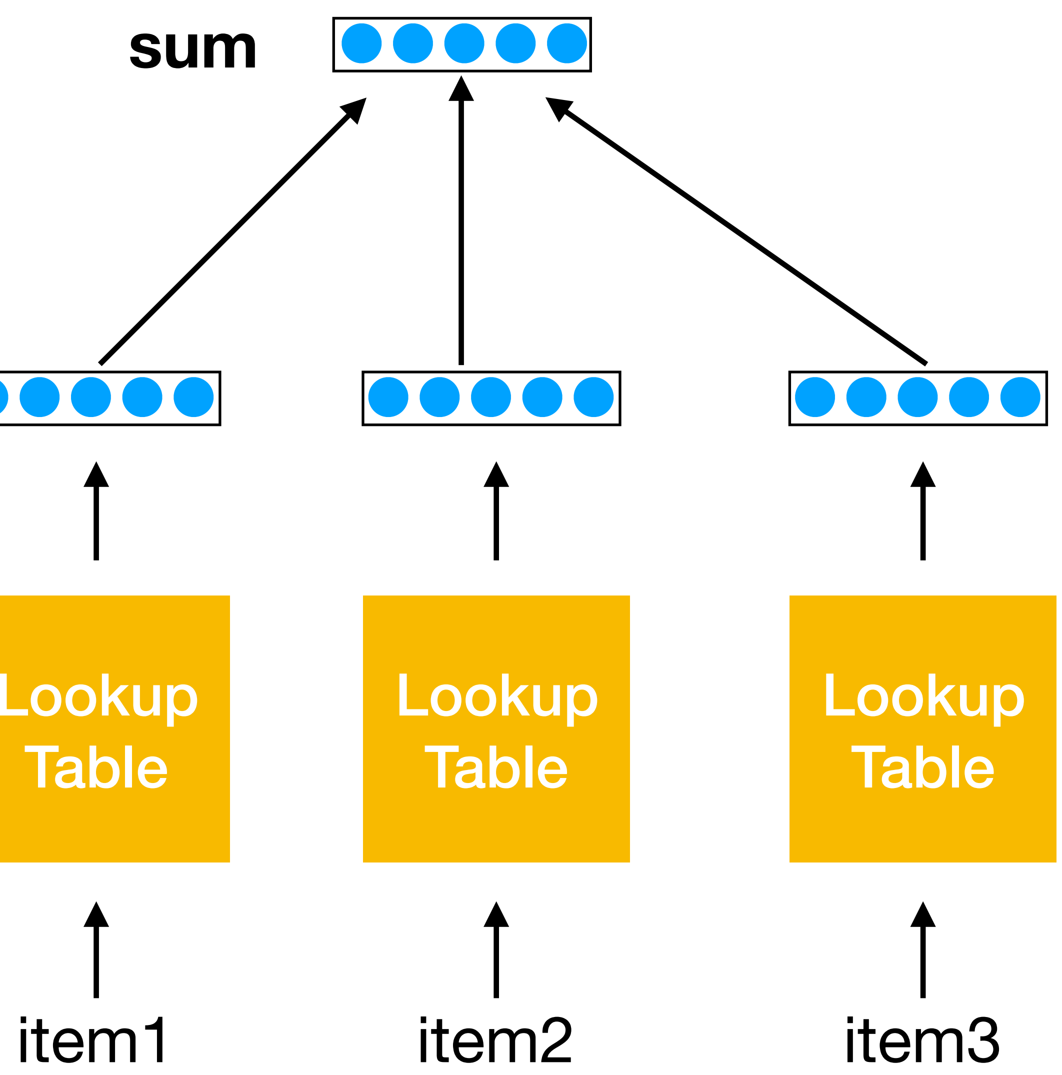
Roll your own: neural network software toolkits

PYTORCH

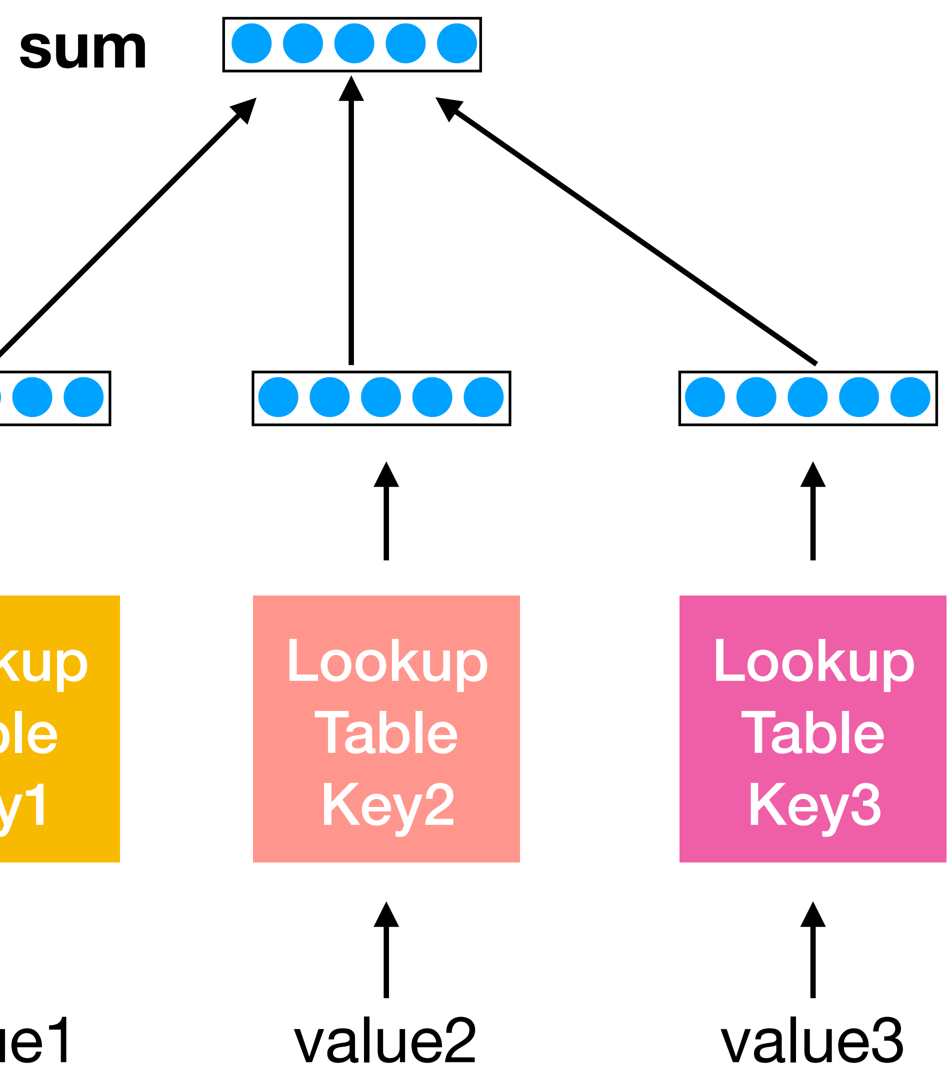
∂y/net

(don't use TensorFlow)

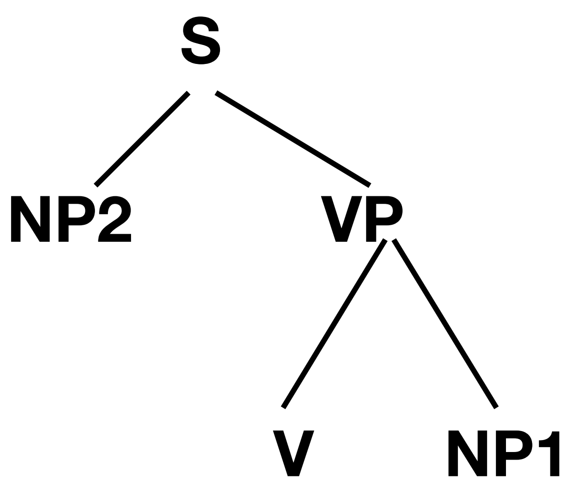
Encoding sets



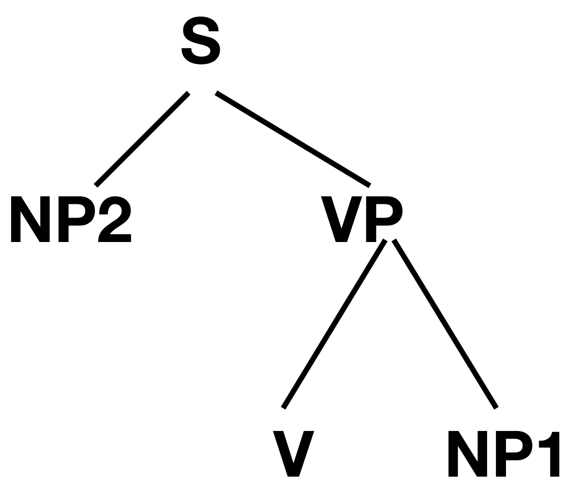
Encoding key=value pairs

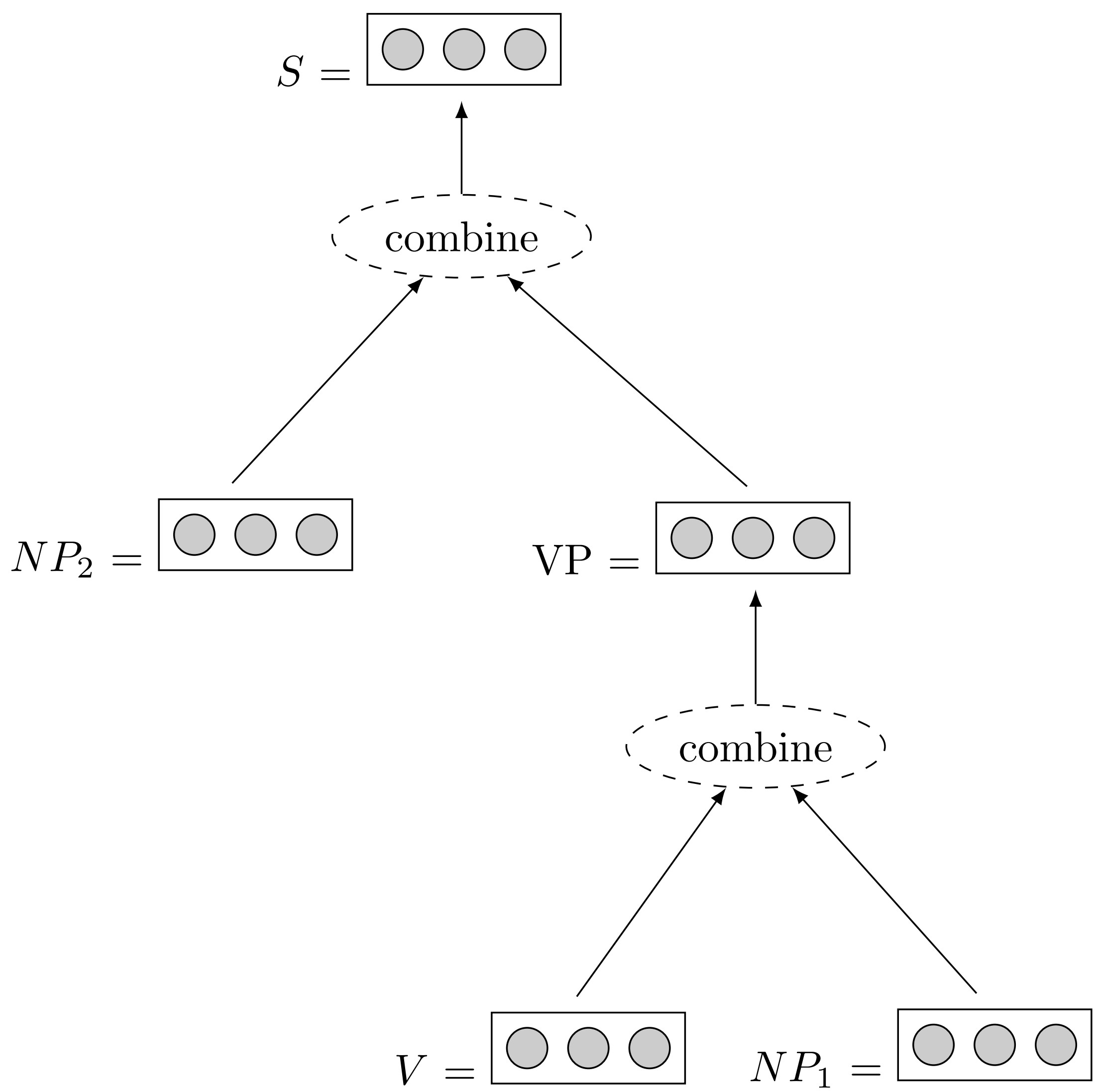


Encoding trees

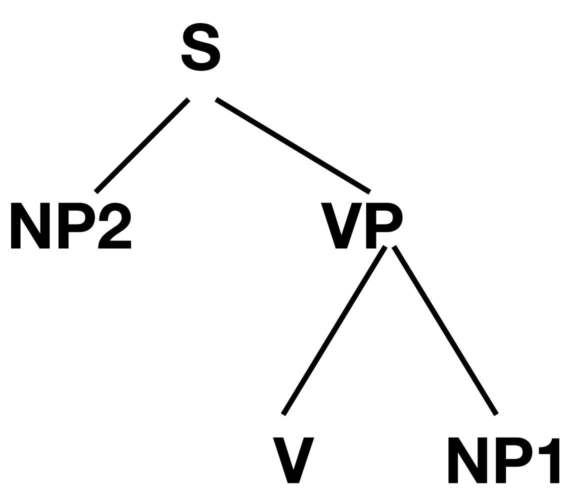


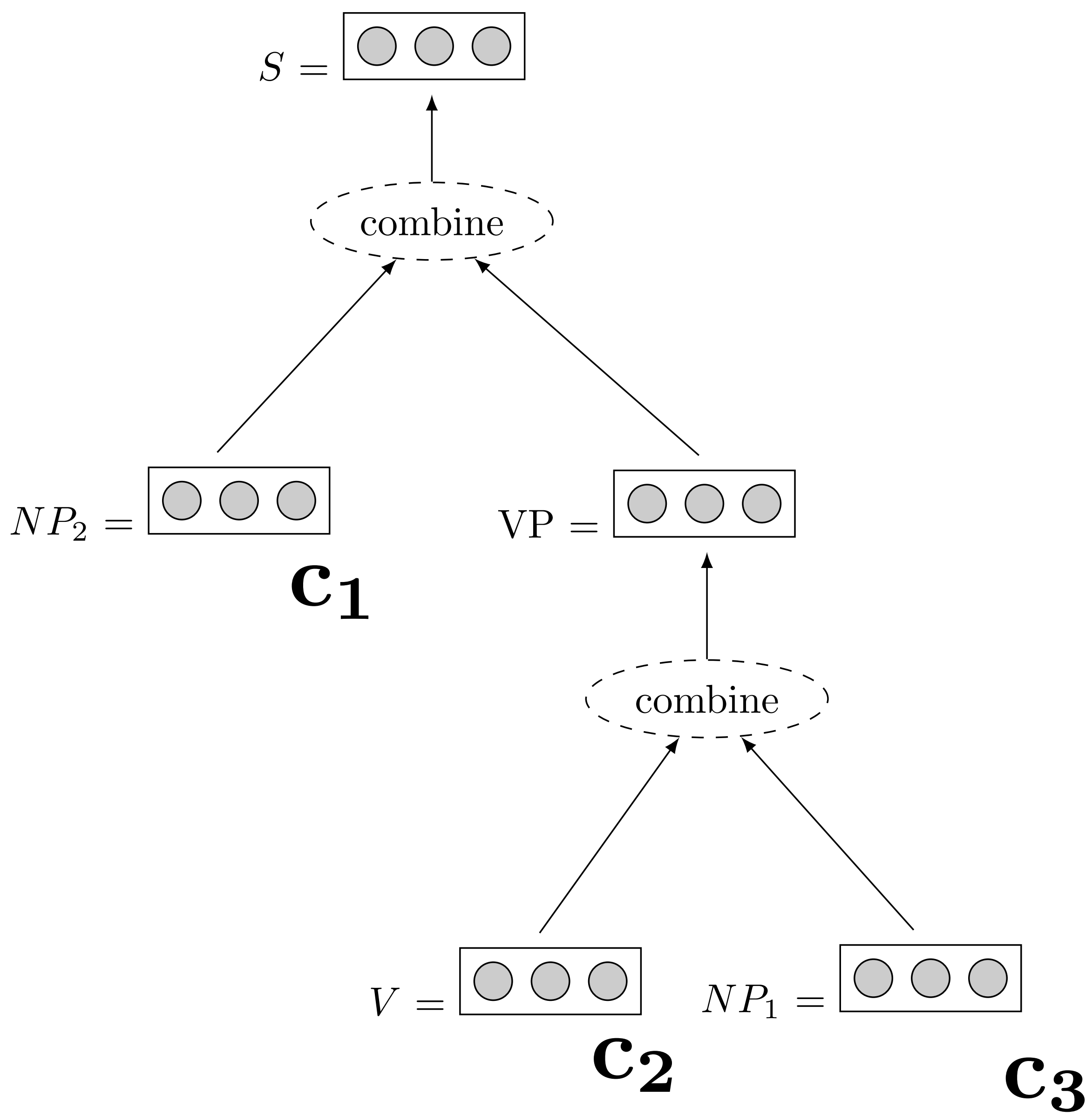
Encoding trees



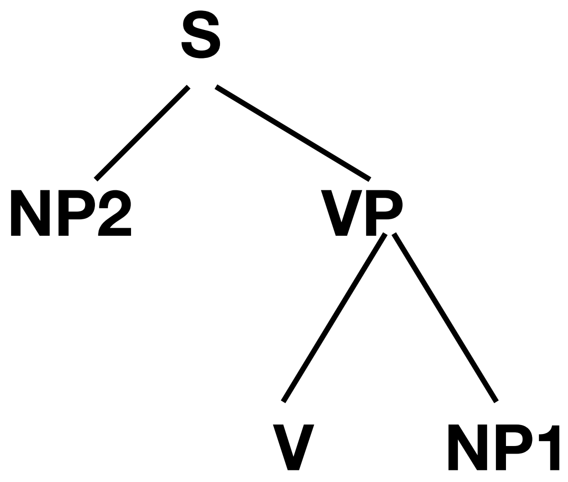


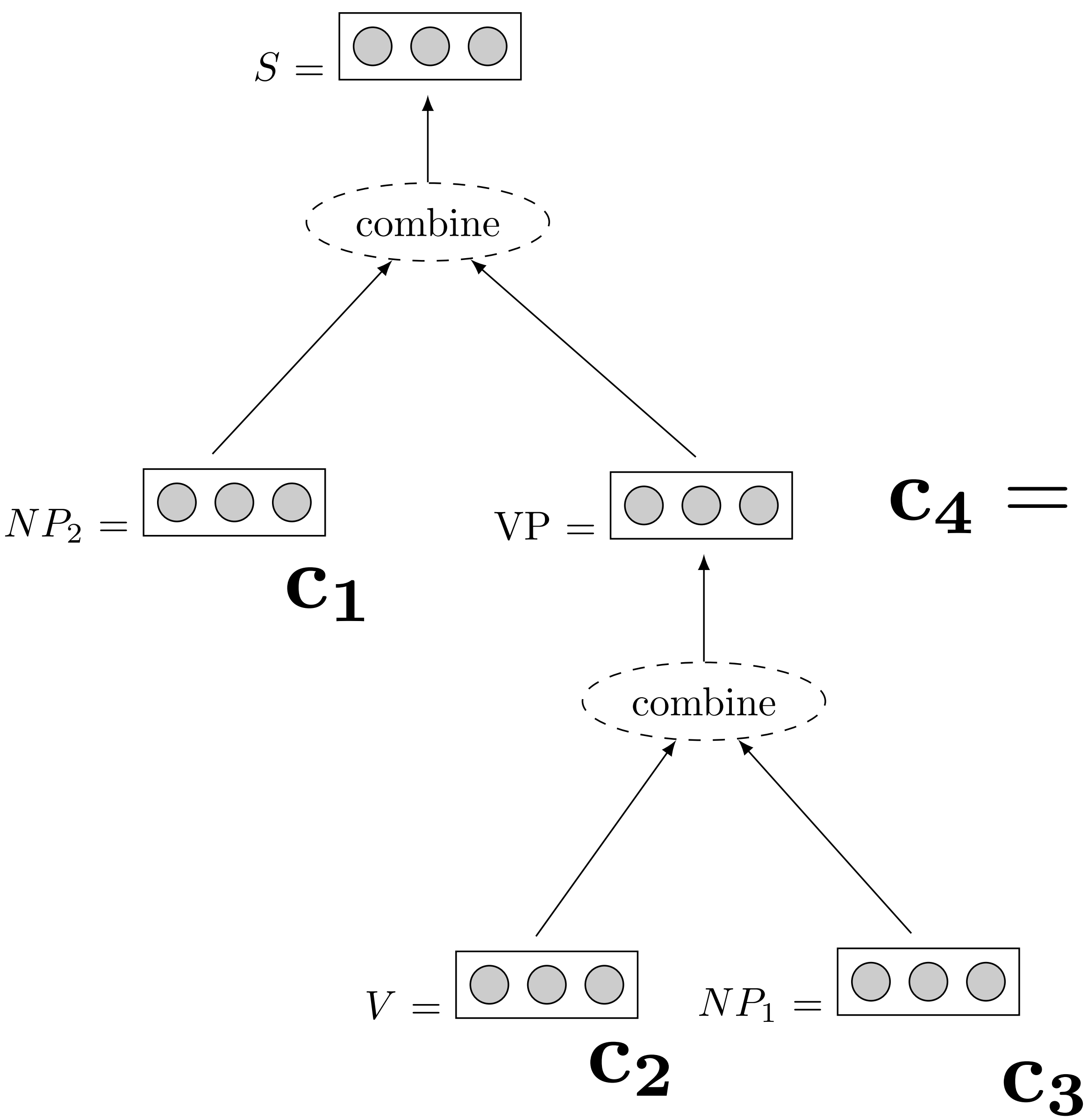
Encoding trees





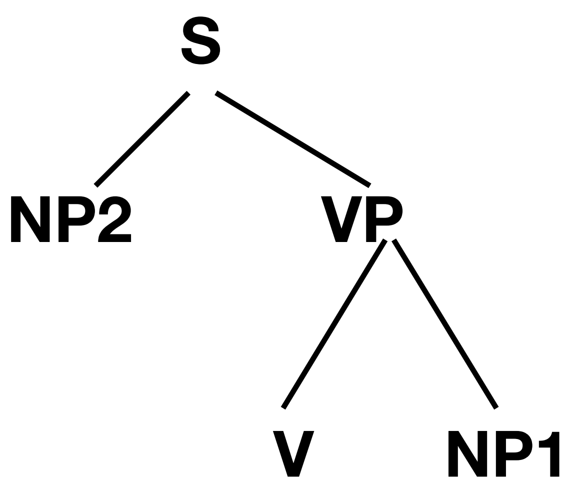
Encoding trees

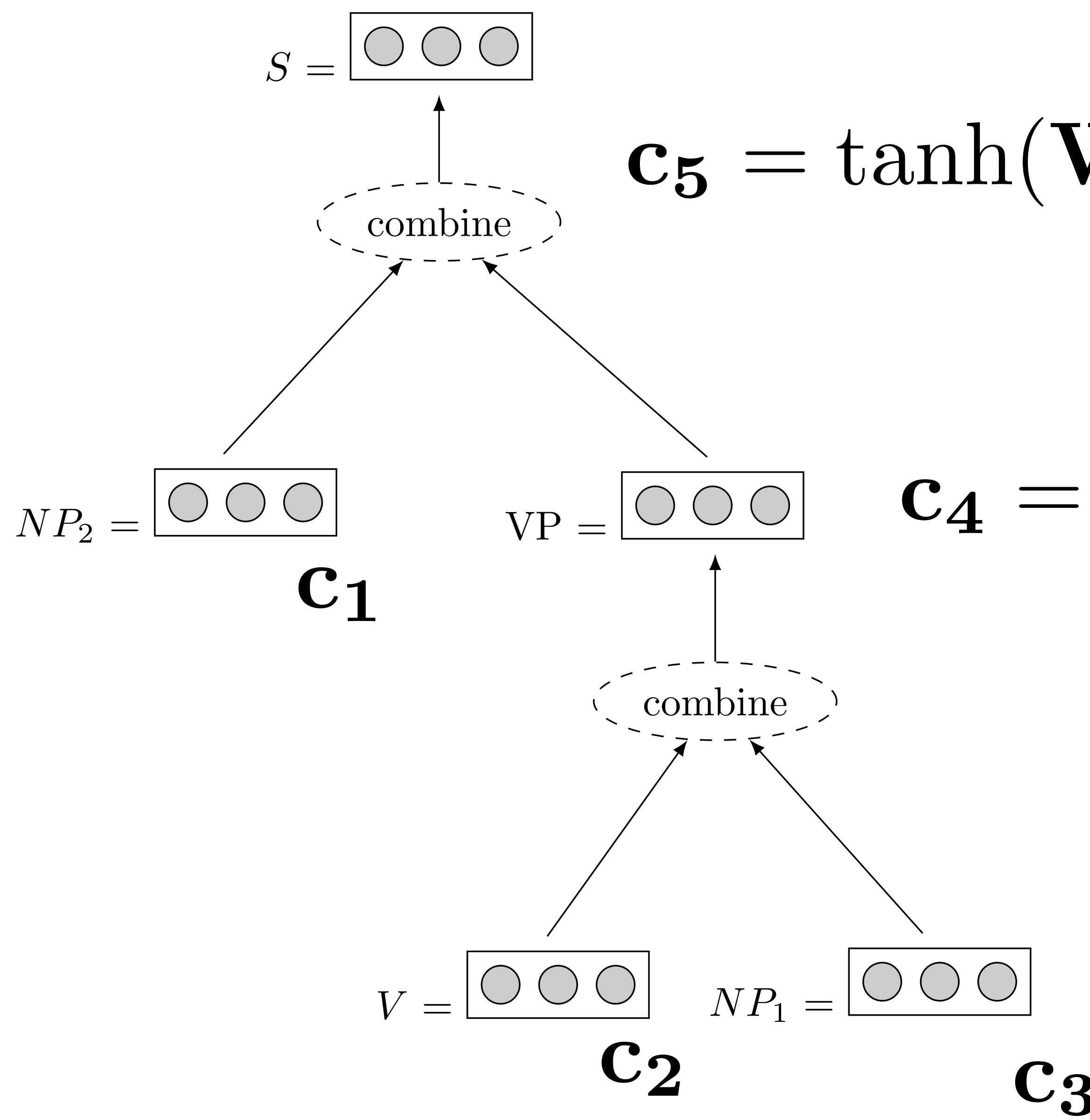




$$\mathbf{c_4} = \tanh(\mathbf{W}[\mathbf{c_2}; \mathbf{c_3}])$$

Encoding trees





$$\mathbf{c_5} = \tanh(\mathbf{W}[\mathbf{c_1}; \mathbf{c_4}])$$

$$\mathbf{c_4} = \tanh(\mathbf{W}[\mathbf{c_2}; \mathbf{c_3}])$$

But you can also do a lot with seq2seq

Just encode things as strings!

Key-value pairs

Name	Triton 52
EcoRating	A+
Family	L7

→ **the Triton 52 has an A+ echo rating and is in the L7 product family**

@ N: Triton 52 @ EC: A+ @ F: L7 @ → **the Triton 52 has an A+ echo rating and is in the L7 product family**

Key-value pairs

Learning to generate one-sentence biographies from Wikidata

Andrew Chisholm
University of Sydney
Sydney, Australia
andy.chisholm.89@gmail.com

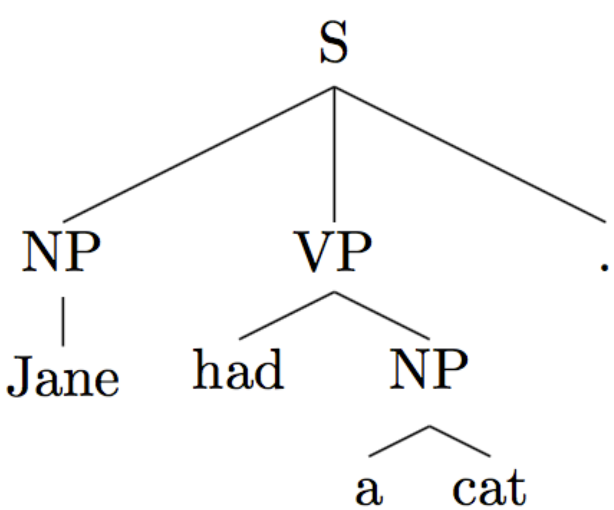
Will Radford
Hugo Australia
Sydney, Australia
wradford@hugo.ai

Ben Hachey
Hugo Australia
Sydney, Australia
bhachey@hugo.ai


```
TITLE mathias tuomi SEX_OR_GENDER
male DATE_OF_BIRTH 1985-09-03
OCCUPATION squash player
CITIZENSHIP finland
```

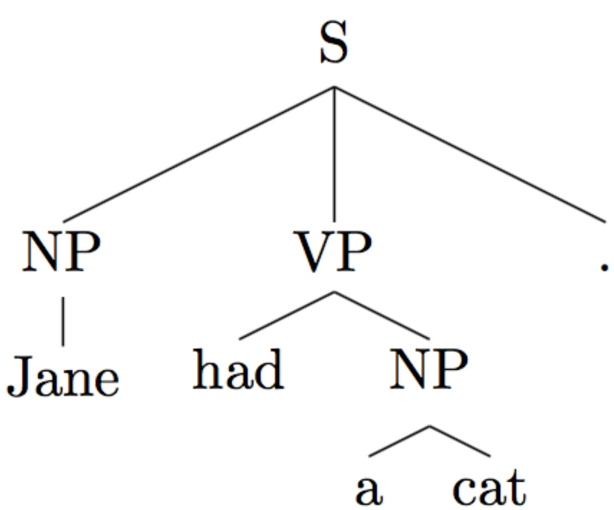
(Task: generate first sentence of wikipedia biography)

Linearized Trees



$$(_S\ (_{NP}\ \textbf{Jane}\ )_{NP}\ (_{VP}\ \textbf{had}\ (_{NP}\ \textbf{a cat}\ )_{NP}\ )_{VP}\ \textbf{.}\ )_S$$

Linearized Trees



$\rightarrow$

$(_S\ (_{NP}\ \textbf{Jane}\ )_{NP}\ (_{VP}\ \textbf{had}\ (_{NP}\ \textbf{a cat}\ )_{NP}\ )_{VP}\ \textbf{.}\ )_S$

feed the tree as a bracketed string into your encoder
or output it as string from the decoder

For many cases, encoding the input and output as linear strings and relying on the attention mechanism and neural-net training work sufficiently well.

NLG (for the rest of this tutorial):

1) Define task.
2) Obtain input/output example.
3) Represent input and output as strings.
4) Train a seq2seq+attention model.

The big challenge: how to get the training data

NLG (for the rest of this tutorial):

1) **Define task.**
2) **Obtain input/output example.**
3) **Represent input and output as strings.**
4) **Train a seq2seq+attention model.**

The big challenge: how to get the training data

NLG (for the rest of this tutorial):

1) Define task.
2) Obtain input/output example.
3) Represent input and output as strings.
4) Train a seq2seq+attention model.

Need MANY examples of input and desired output.

Translating with Politeness

Controlling Politeness in Neural Machine Translation via Side Constraints

Rico Sennrich and **Barry Haddow** and **Alexandra Birch**
School of Informatics, University of Edinburgh
{rico.sennrich,a.birch}@ed.ac.uk, bhaddow@inf.ed.ac.uk

Translating with Politeness

3 NMT with Side Constraints

We are interested in machine translation for language pairs where politeness is not grammatically marked in the source text, but should be predicted in the target text. The basic idea is to provide the neural network with additional input features that mark *side constraints* such as politeness.

Translating with Politeness

At training time, the correct feature is extracted from the sentence pair as described in the following section. At test time, we assume that the side constraint is provided by a user who selects the desired level of politeness of the translation.

Translating with Politeness

We add side constraints as special tokens at the end of the source text, for instance <*T*> or <*V*>. The attentional encoder-decoder framework is then able to learn to pay attention to the side constraints.

Translating with Politeness

We automatically annotate politeness on a sentence level with rules based on a morphosyntactic annotation by ParZu (Sennrich et al., 2013). Sentences containing imperative verbs are labelled informal. Sentences containing an informal or polite pronoun from Table 1 are labelled with the corresponding class.

Generating with Style

Controlling Linguistic Style Aspects in Neural Language Generation

Jessica Ficler and **Yoav Goldberg**
Computer Science Department
Bar-Ilan University
Israel
{jessica.ficler, yoav.goldberg}@gmail.com

Generating with Style

[Jessica's slides]

Our goal is to generate text…

…while allowing control of its style.

Style

The same message (e.g. expressing a positive sentiment towards a movie) can be conveyed in different ways.

Style Aspects (Example)

“OMG... This movie actually made me cry a little bit because I laughed so hard at some parts.“

Style Aspects (Example)

"OMG... This movie actually made me cry a little bit because I laughed so hard at some parts."

Colloquial style

Style Aspects (Example)

"OMG... This movie actually made me cry a little bit because I laughed so hard at some parts.“

Colloquial style

Personal voice

Style Aspects (Example)

"OMG... This movie actually made me cry a little bit because I laughed so hard at some parts."

Colloquial style

Personal voice

Few adjectives

Style Aspects (Example)

“OMG... This movie actually made me cry a little bit because I laughed so hard at some parts.“

Colloquial style

Personal voice

Few adjectives

“A genuinely unique, full-on sensory experience that treads its own path between narrative clarity and pure visual expression.”

Style Aspects (Example)

“OMG... This movie actually made me cry a little bit because I laughed so hard at some parts.“

Colloquial style

Personal voice

Few adjectives

“A genuinely unique, full-on sensory experience that treads its own path between narrative clarity and pure visual expression.”

Professional critic

Style Aspects (Example)

"OMG... This movie actually made me cry a little bit because I laughed so hard at some parts."

Colloquial style

Personal voice

Few adjectives

"A genuinely unique, full-on sensory experience that treads its own path between narrative clarity and pure visual expression."

Professional critic

Impersonal voice

Style Aspects (Example)

"OMG... This movie actually made me cry a little bit because I laughed so hard at some parts."

"A genuinely unique, full-on sensory experience that treads its own path between narrative clarity and pure visual expression."

Colloquial style	Professional critic
Personal voice	Impersonal voice
Few adjectives	Many adjectives

The challenge

Generate text that conforms to a set of content-based and stylistic requirements.

The challenge

Generate **text** that conforms to a set of content-based and stylistic requirements.

full length, natural sentences

The challenge

more than 2

Generate **text** that conforms to a **set** of <u>content-based</u> and <u>stylistic</u> requirements.

full length, natural sentences

Example

Theme: Acting
Descriptive: True

Example

Theme: Acting
Descriptive: True

“A wholly original, well-acted, romantic comedy that's elevated by the modest talents of a lesser known cast.”

Example

Theme: Acting
Descriptive: True

“A wholly original, **well-acted**, romantic comedy that's elevated by the modest talents of a lesser known **cast**.”

Example

Theme: Acting
Descriptive: True

“A wholly original, **well-acted**, romantic comedy that's elevated by the modest talents of a lesser known **cast**.”

Example

Theme: Acting
Descriptive: True

“A wholly <u>original</u>, <u>**well-acted**</u>, <u>romantic</u> comedy that's elevated by the <u>modest</u> talents of a <u>lesser</u> <u>known</u> **cast**.”

Theme: Plot
Descriptive: False

“I think the <u>poor</u> **writing** and **script** are what caused this movie to bomb.”

Formal Definition

- We assume a set of k parameters $p_1 \ldots p_k$, each parameter p_i with a set of possible values V^{p_i}

Formal Definition

- We assume a set of k parameters $p_1 \ldots p_k$, each parameter p_i with a set of possible values V^{p_i}
- Input: specific assignment to these parameters

e.g.

Parameter	**Value**
Professional	False
Personal	True
Length	≤ 10
Descriptive	False
Theme	Other
Sentiment	Positive

Formal Definition

- We assume a set of k parameters $p_1 \ldots p_k$, each parameter p_i with a set of possible values V^{p_i}
- Input: specific assignment to these parameters

e.g.

Parameter	Value
Professional	False
Personal	True
Length	≤ 10
Descriptive	False
Theme	Other
Sentiment	Positive

Output: a text that is compatible with the parameters values

e.g. "I don't understand why it is rated so poorly."

This work

We consider 6 parameters and values from the movie reviews domain

Style	Content
Professional Personal Descriptive Length	Sentiment Theme

Content Parameters

Task Description – Content Parameters

Sentiment - The score that the reviewer gave the movie

Positive

"This movie is so much to keep you on the edge of your seat."

Neutral

"While the film doesn't quite reach the level of sugar fluctuations, it's beautifully animated."

Negative

"It's a very low-budget movie that just seems to be a bunch of fluff."

Task Description – Content Parameters

Theme - Whether the sentence's content is about the Plot, Acting, Production, Effects or none of these (Other)

Task Description – Content Parameters

Theme - Whether the sentence's content is about the Plot, Acting, Production, Effects or none of these (Other)

Plot - “The **storyline** had me laughing out loud.”

Task Description – Content Parameters

Theme - Whether the sentence's content is about the Plot, Acting, Production, Effects or none of these (Other)

Plot - “The **storyline** had me laughing out loud.”

Acting - “The **cast** are all excellent.”

Task Description – Content Parameters

Theme - Whether the sentence's content is about the Plot, Acting, Production, Effects or none of these (Other)

Plot - “The **storyline** had me laughing out loud.”

Acting - “The **cast** are all excellent.”

Production - “The **director**'s magical.”

Task Description – Content Parameters

Theme - Whether the sentence's content is about the Plot, Acting, Production, Effects or none of these (Other)

Plot - “The **storyline** had me laughing out loud.”

Acting - “The **cast** are all excellent.”

Production - “The **director**'s magical.”

Effects - “Only saving grace is the **sound effects**.”

Task Description – Content Parameters

Theme - Whether the sentence's content is about the Plot, Acting, Production, Effects or none of these (Other)

Plot - “The **storyline** had me laughing out loud.”

Acting - “The **cast** are all excellent.”

Production - “The **director**'s magical.”

Effects - “Only saving grace is the **sound effects**.”

Other - “I'm afraid that the movie is aimed at kids and adults weren't sure what to say about it.”

Style Parameters

Task Description – Style Parameters

Length – Number of words

$\leq$ 10 words

11-20 words

21-40 words

> 40 words

Task Description – Style Parameters

Professional - Whether the review is written in the style of a professional critic or not

Task Description – Style Parameters

Professional - Whether the review is written in the style of a professional critic or not

True

“This is a breath of fresh air, it's a welcome return to the franchise's brand of satirical humor.”

Task Description – Style Parameters

Professional - Whether the review is written in the style of a professional critic or not

True	False
“This is a breath of fresh air, it's a welcome return to the franchise's brand of satirical humor.”	“So glad to see this movie !!”

Task Description – Style Parameters

Personal - Whether the review describes subjective experience (written in personal voice) or not

Task Description – Style Parameters

Personal - Whether the review describes subjective experience (written in personal voice) or not

True

“I could see the movie again”

Task Description – Style Parameters

Personal - Whether the review describes subjective experience (written in personal voice) or not

True	False
“I could see the movie again”	“Very similar to the book.”

Task Description – Style Parameters

Descriptive - Whether the review is in descriptive (contains a high ratio of adjectives) style or not

Task Description – Style Parameters

Descriptive - Whether the review is in descriptive (contains a high ratio of adjectives) style or not

True

"Such a **hilarious** and **funny romantic** comedy."

Task Description – Style Parameters

Descriptive - Whether the review is in descriptive (contains a high ratio of adjectives) style or not

True	False
"Such a **hilarious** and **funny romantic** comedy."	"A **definite** must see for fans of anime fans, pop culture references and animation with a **good** laugh too."

And we would like to control for all these aspects **simultaniously**

Type	Parameter	Value
Style	Professional	False
Style	Personal	True
Style	Length	≤ 10
Style	Descriptive	False
Content	Theme	Other
Content	Sentiment	Positive

“I don't understand why it is rated so poorly.”

Model

Model

a conditioned language model:

$$P(w_1 \dots w_n | c) = \prod_{t=1}^{n} P(w_t | w_1, \dots, w_{t-1}, c)$$

Model

a conditioned language model:

$$P(w_1 \dots w_n | c) = \prod_{t=1}^{n} P(w_t | w_1, \dots, w_{t-1}, c)$$

Condition each word on the **history**, as well as on a context **c**.

Model

In our case, c is a concatenation of the parameters values embedding vectors

c:	Personal:False	Sentiment:Positive	Length:≤10	Descriptive:True	Proffesional:True	Theme:Plot

Model

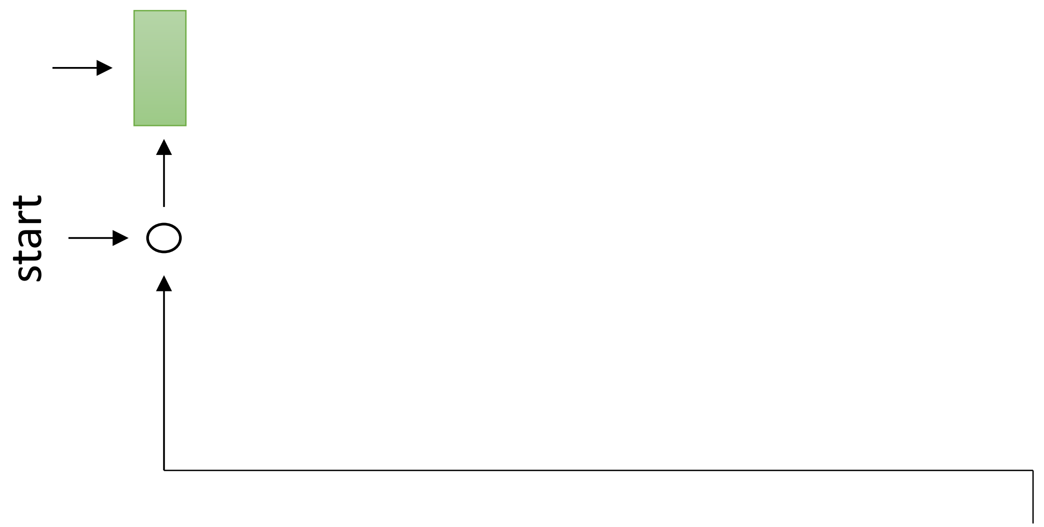
In our case, c is a concatenation of the parameters values embedding vectors

c:

Personal:False	Sentiment:Positive	Length:≤10	Descriptive:True	Proffesional:True	Theme:Plot

Model

In our case, c is a concatenation of the parameters values embedding vectors

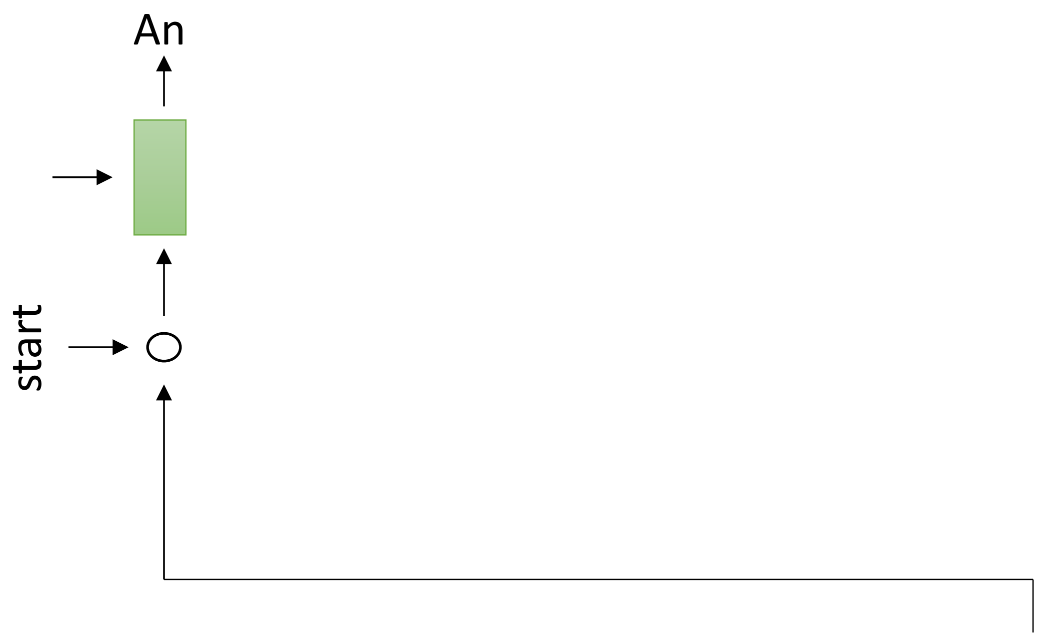


c:

Personal:False	Sentiment:Positive	Length:≤10	Descriptive:True	Proffesional:True	Theme:Plot

Model

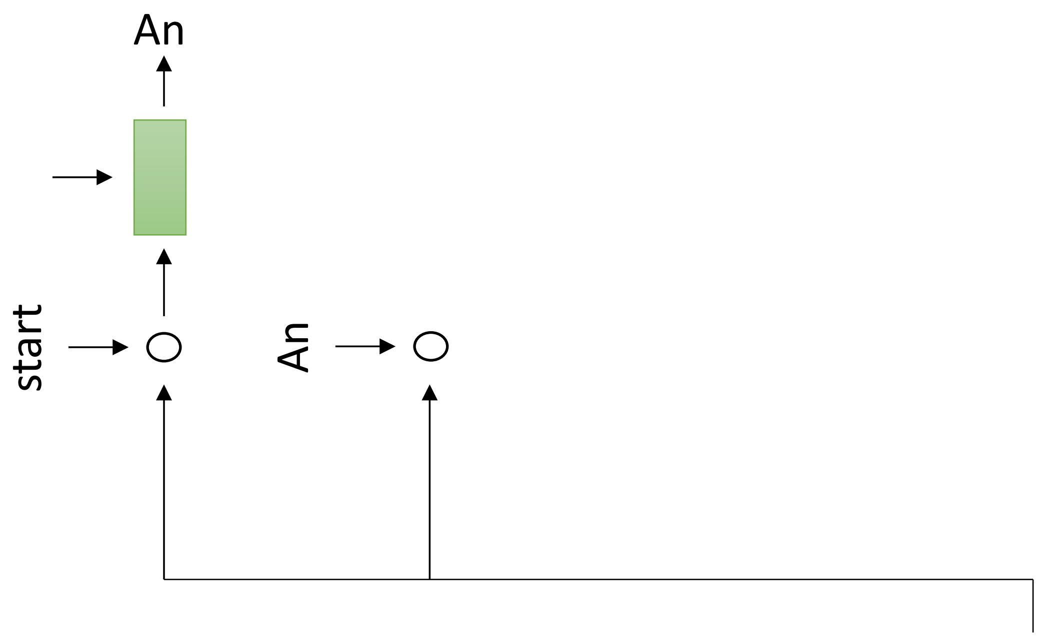
In our case, c is a concatenation of the parameters values embedding vectors



c:	Personal:False	Sentiment:Positive	Length:≤10	Descriptive:True	Proffesional:True	Theme:Plot

Model

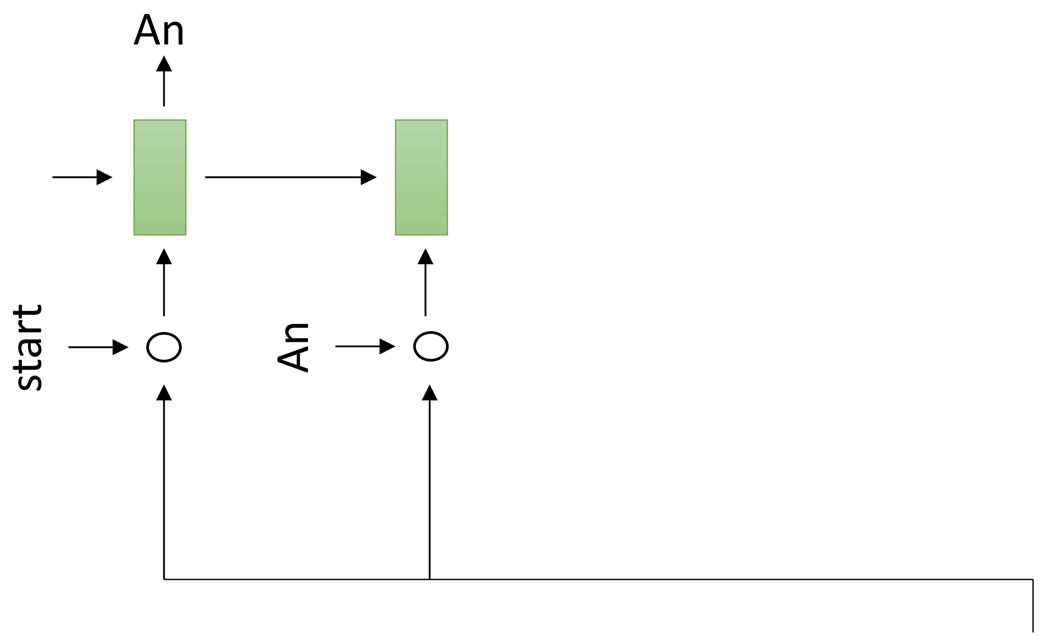
In our case, c is a concatenation of the parameters values embedding vectors



c:					
Personal:False	Sentiment:Positive	Length:≤10	Descriptive:True	Proffesional:True	Theme:Plot

Model

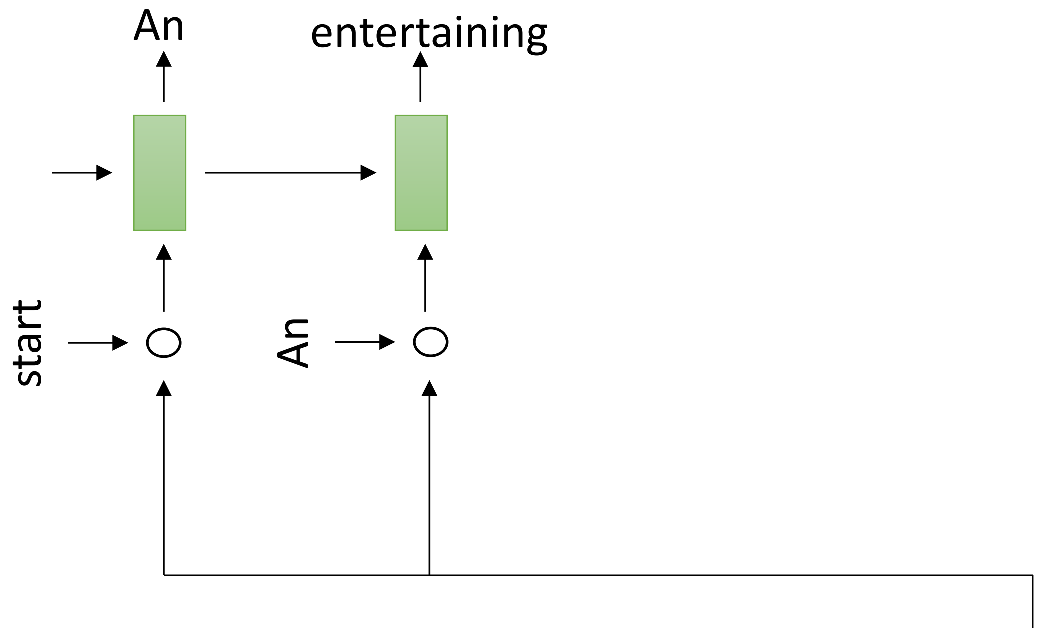
In our case, c is a concatenation of the parameters values embedding vectors



c:	Personal:False	Sentiment:Positive	Length:≤10	Descriptive:True	Proffesional:True	Theme:Plot

Model

In our case, c is a concatenation of the parameters values embedding vectors

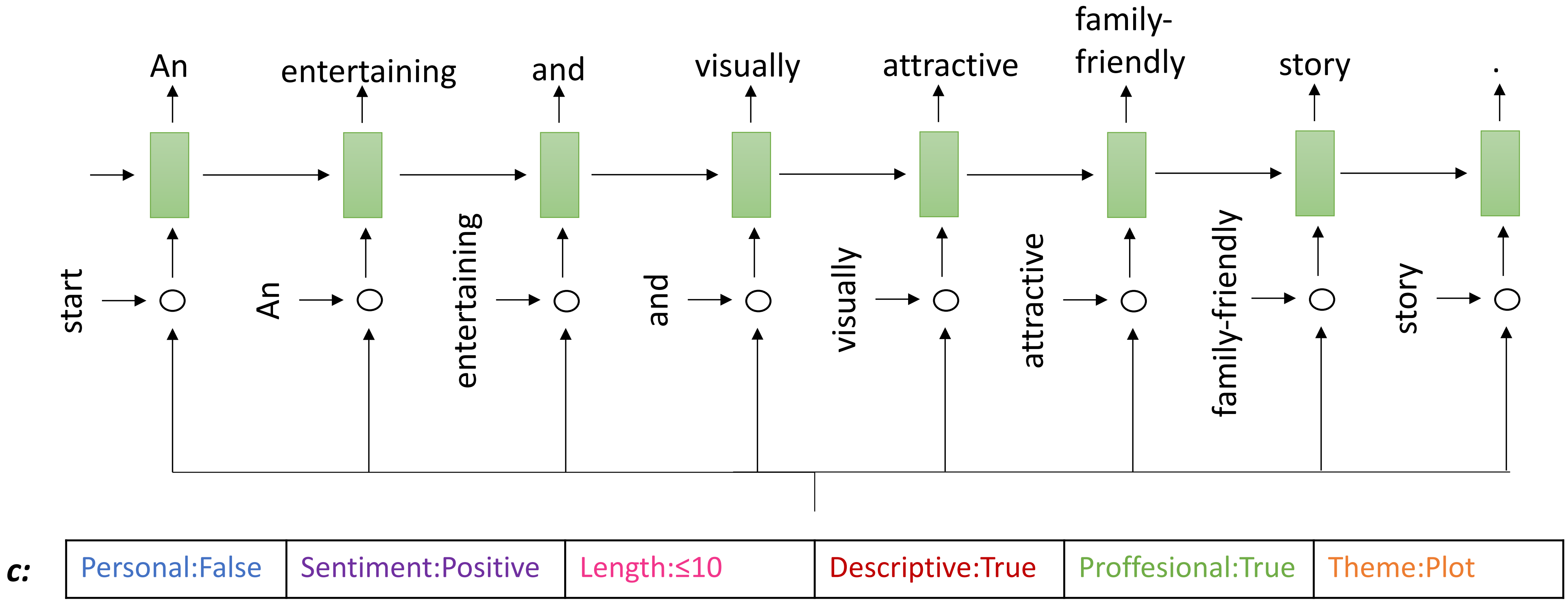


c:

Personal:False	Sentiment:Positive	Length:≤10	Descriptive:True	Proffesional:True	Theme:Plot

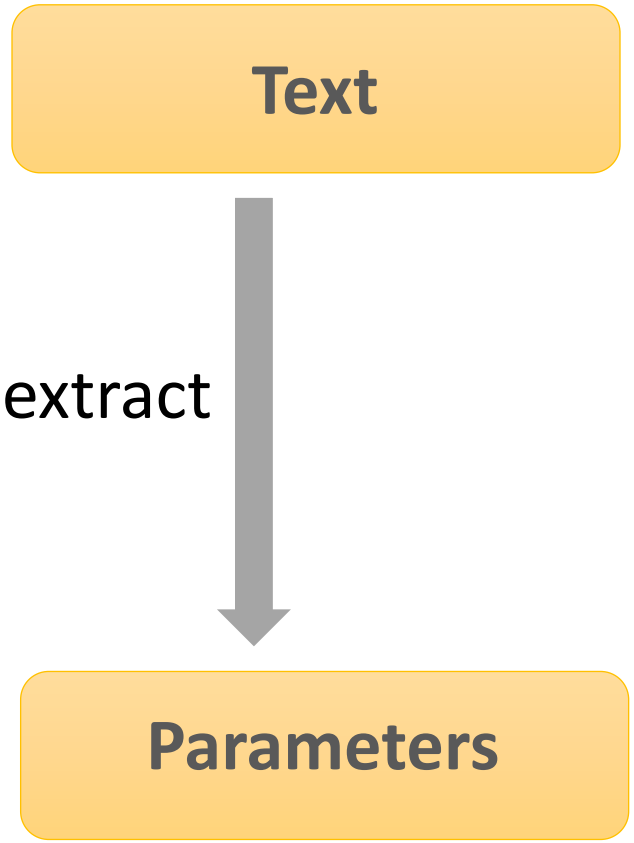
Model

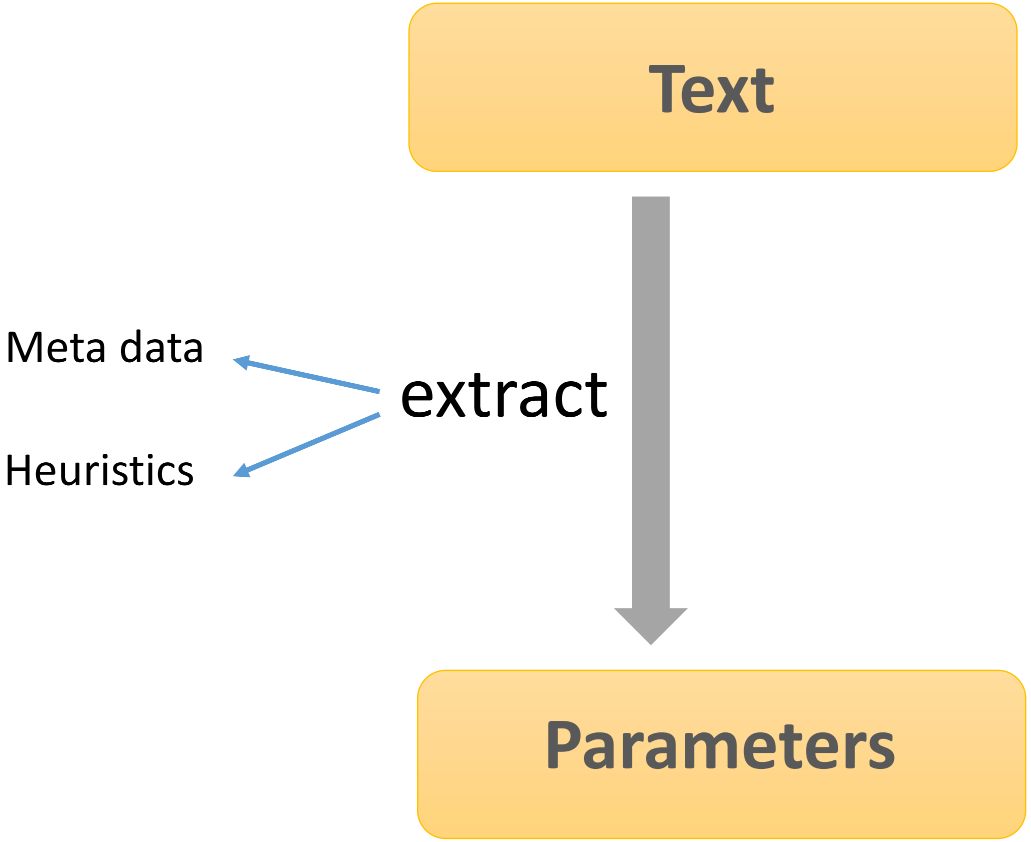
In our case, c is a concatenation of the parameters values embedding vectors

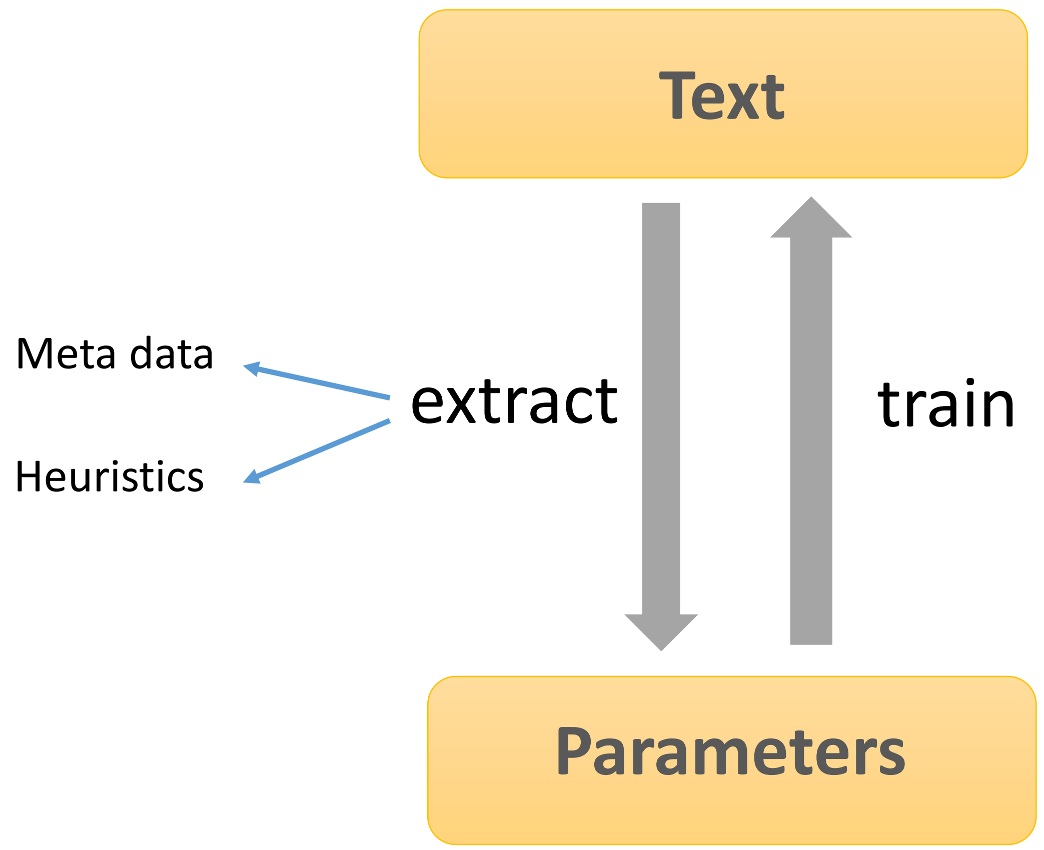


The model is simple, but…

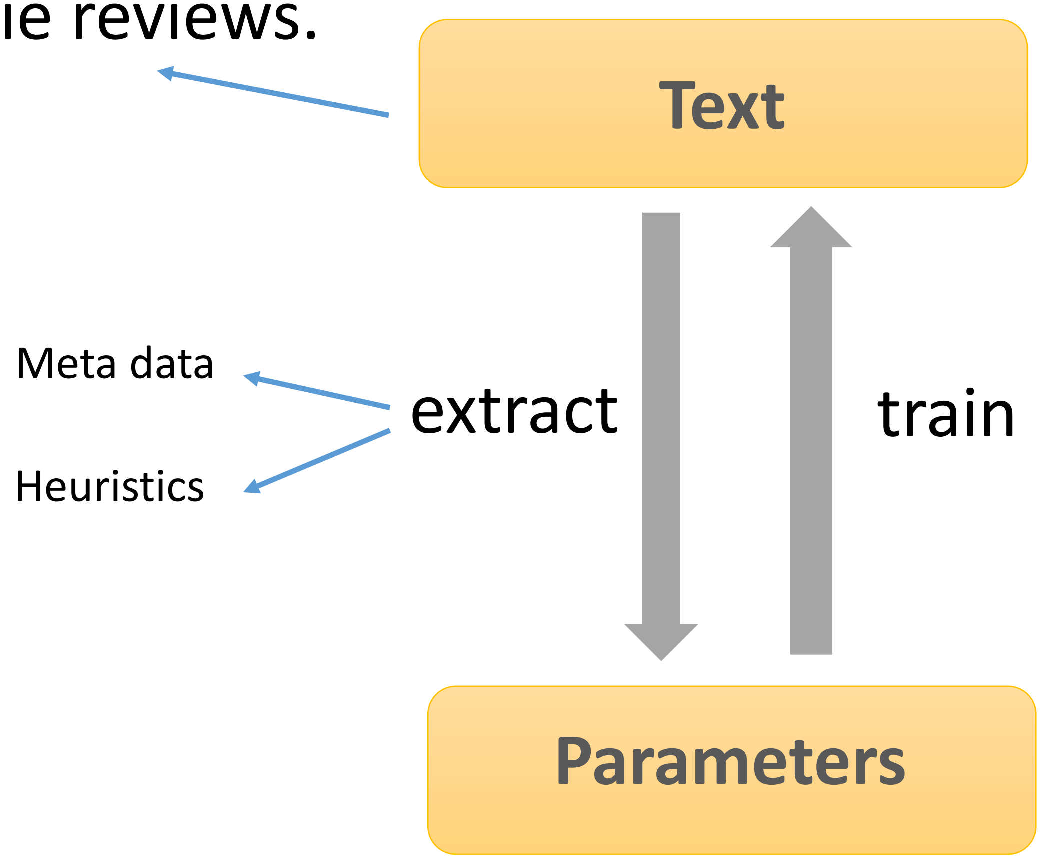
we need **training data** annotated with the appropriate values.


Text
extract
Parameters

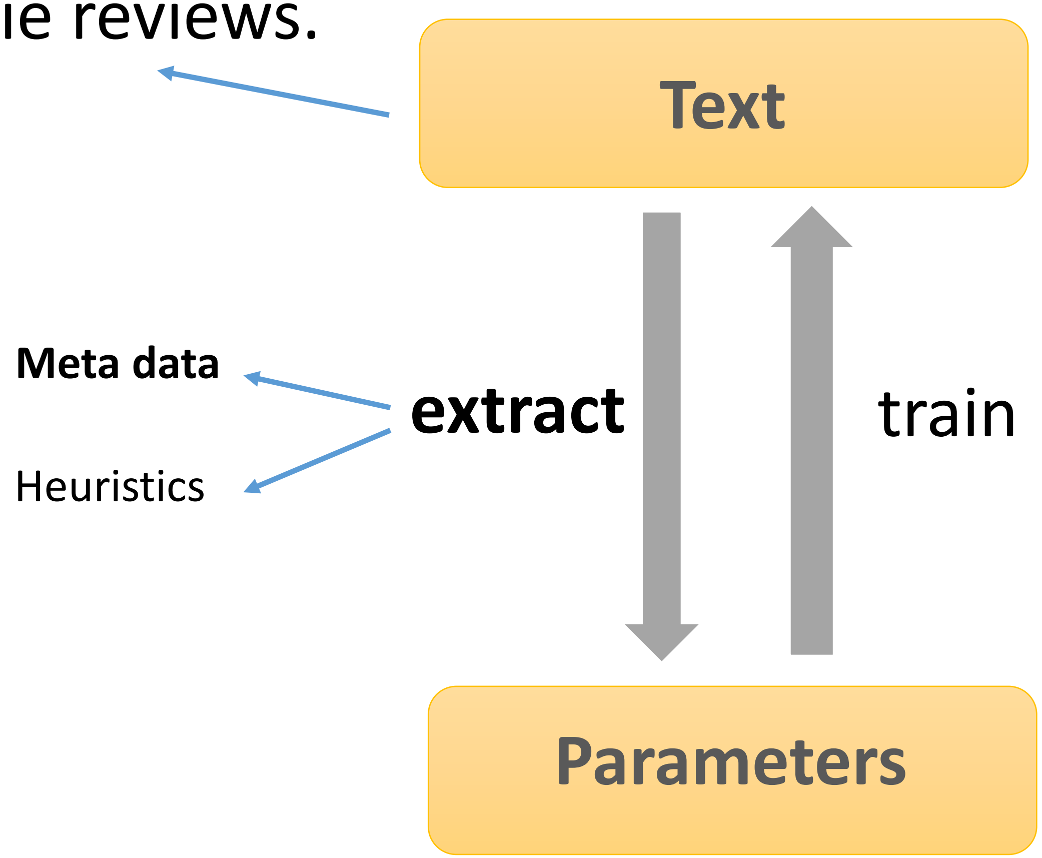

Text
Meta data
extract
Heuristics
Parameters


Text
Meta data
extract
train
Heuristics
Parameters

Rotten-Tomatoes website.
7,500 movies.
1,002,625 movie reviews.



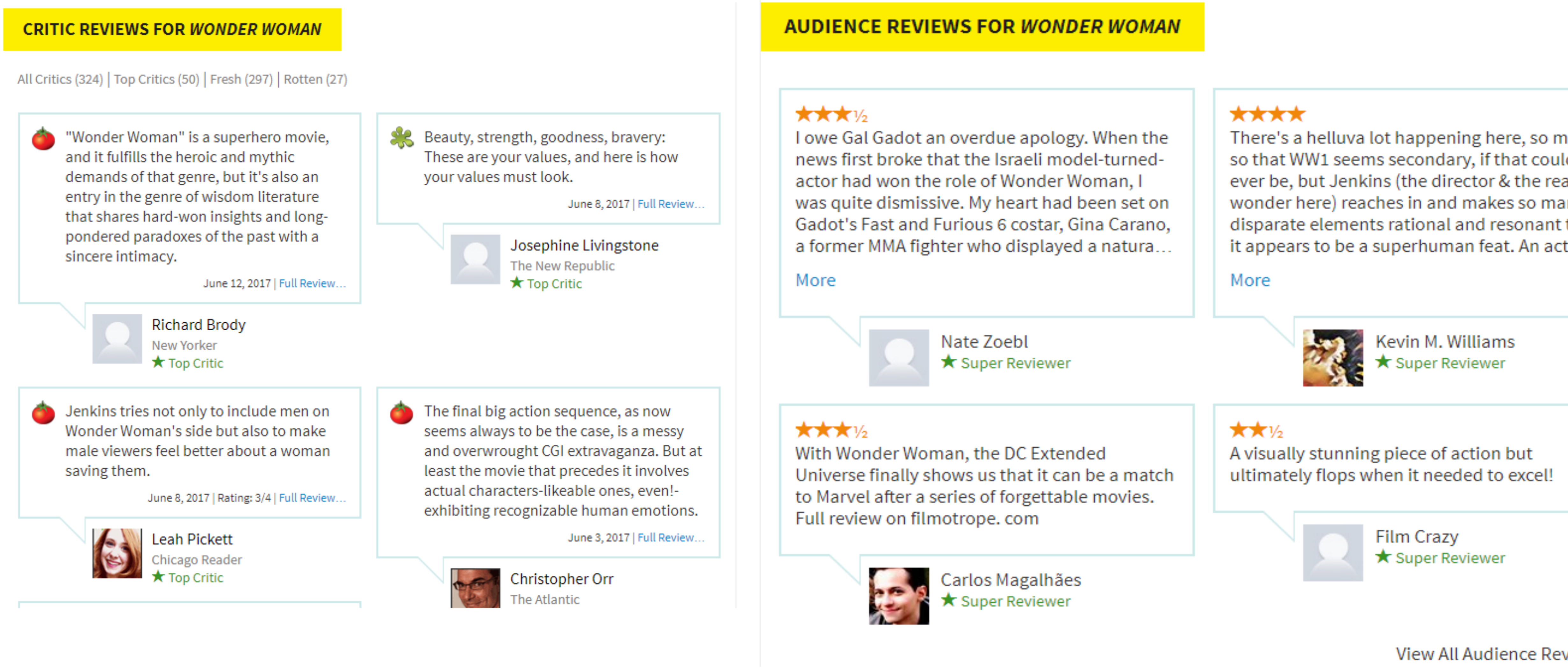
Rotten-Tomatoes website.
7,500 movies.
1,002,625 movie reviews.



Professional

Professional

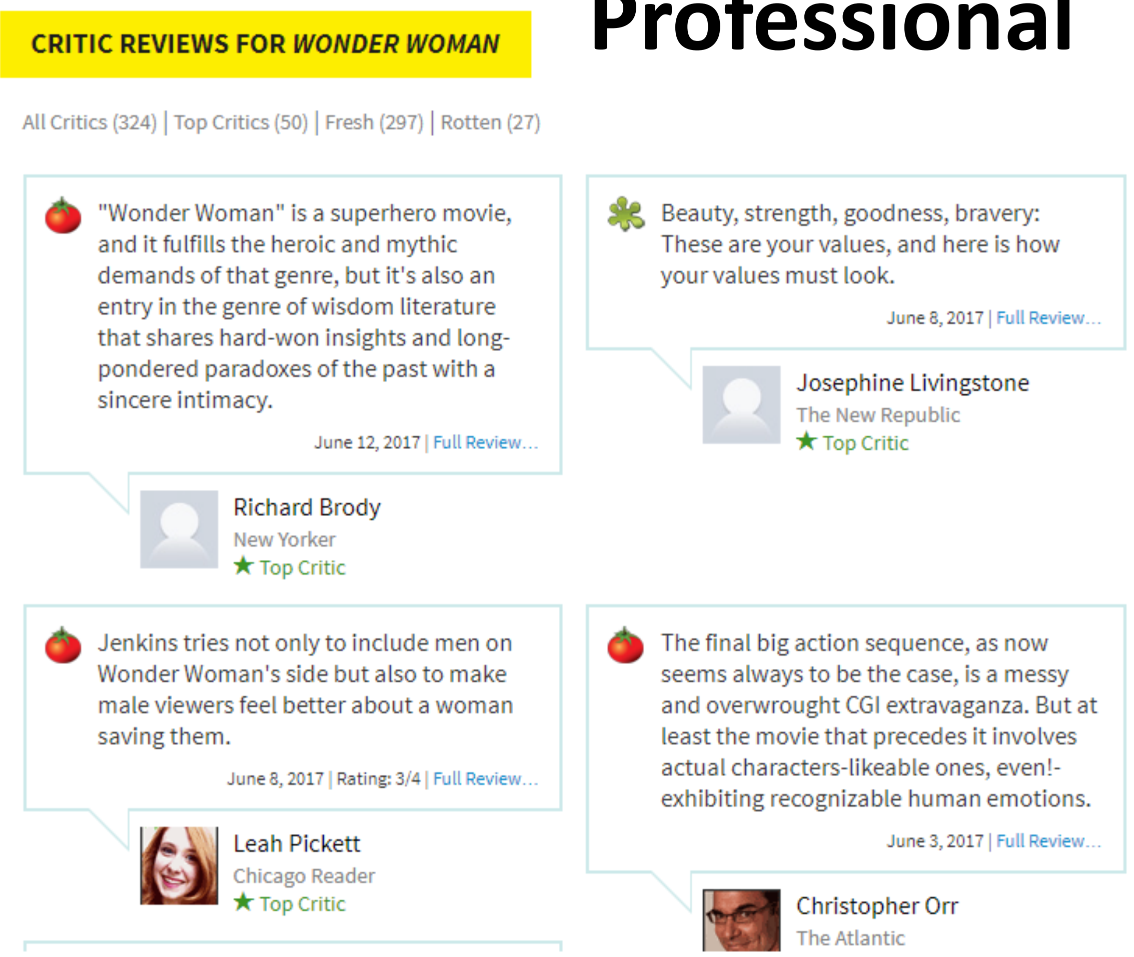
In rottentomatoes the critic reviews are separated from the audience review

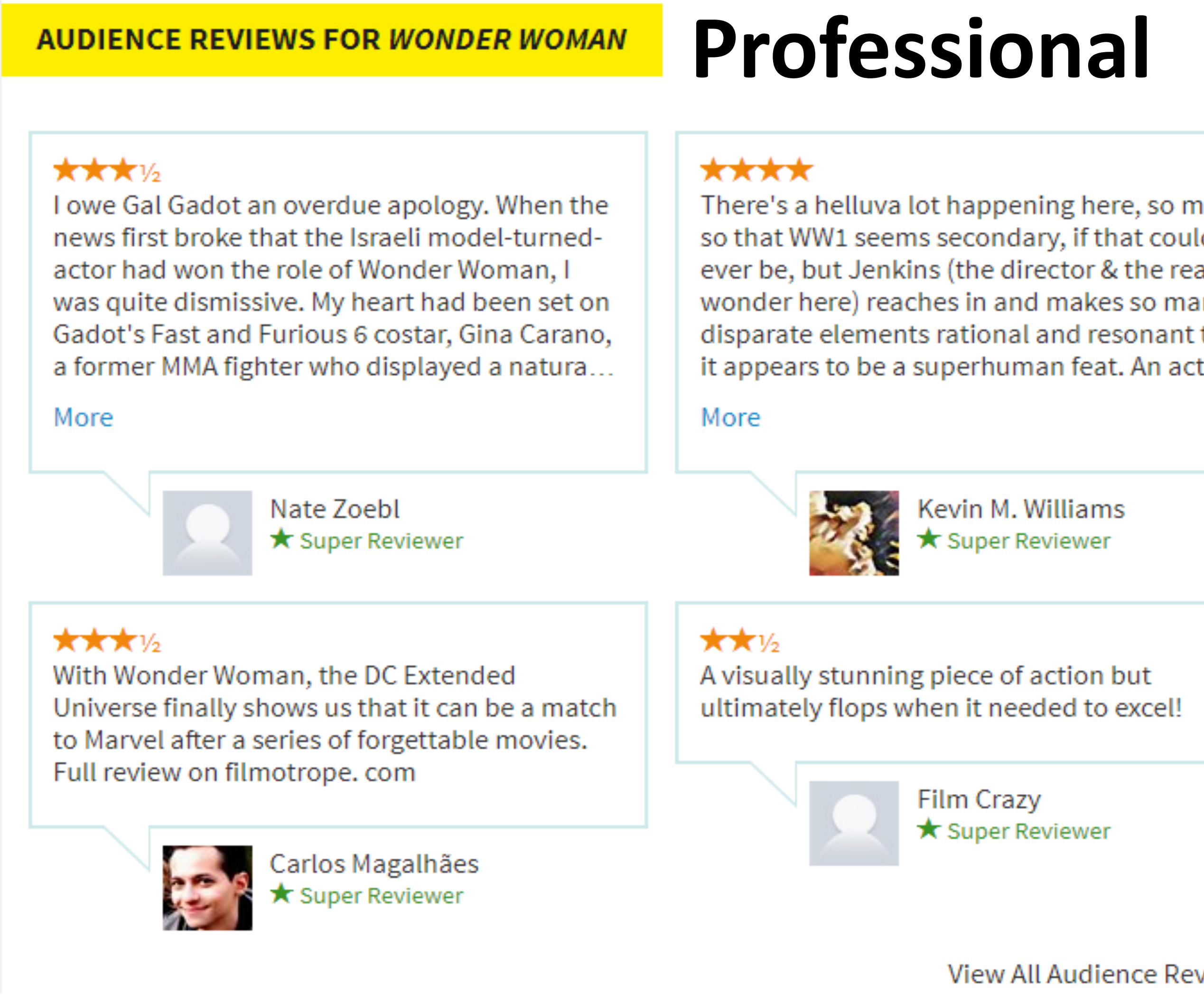


Professional

In rottentomatoes the critic reviews are separated from the audience **Non Professional**

Professional





WONDER WOMAN REVIEWS

All Critics | Top Critics | My Critics | Audience

Page 3 of 415

Robert W ★★★★ July 28, 2017

Decent film with likable characters.

Sherry M ★½ July 28, 2017

A three sleep movie for me.

Mina N ★★★★½ July 28, 2017

I recommend this movie to all of u go out anytime and see this movie do this as a service for yourself,thats a real good DCEU Movie,gal gadot is a great choice as wonder woman although went she was announced to play it when the batman vs superman cast was announced i said she is a terrible choice cause she is not a good actors but this role will make her learn from her mistakes and chris pine gave us his 2nd best performance after hell or high water and the sarcasm was made in its time the action sequence was perfect the directress patty Jenkins made it in a perfect way not like my expectations i bet that the film will get 2 oscar nods for costume design and best VFX I Can't wait to see it again and again
My score:9.5/10

Antonius B ★ Super Reviewer

I know this is a little different from the popular view, but I thought 'Wonder Woman' was just ok, though a decent way to spend summer afternoon. I liked a woman superhero, the diversity in the movie, and the character of the God of War, and how he debated the nature of humanity with Wonder Woman towards the end. On the other hand, it

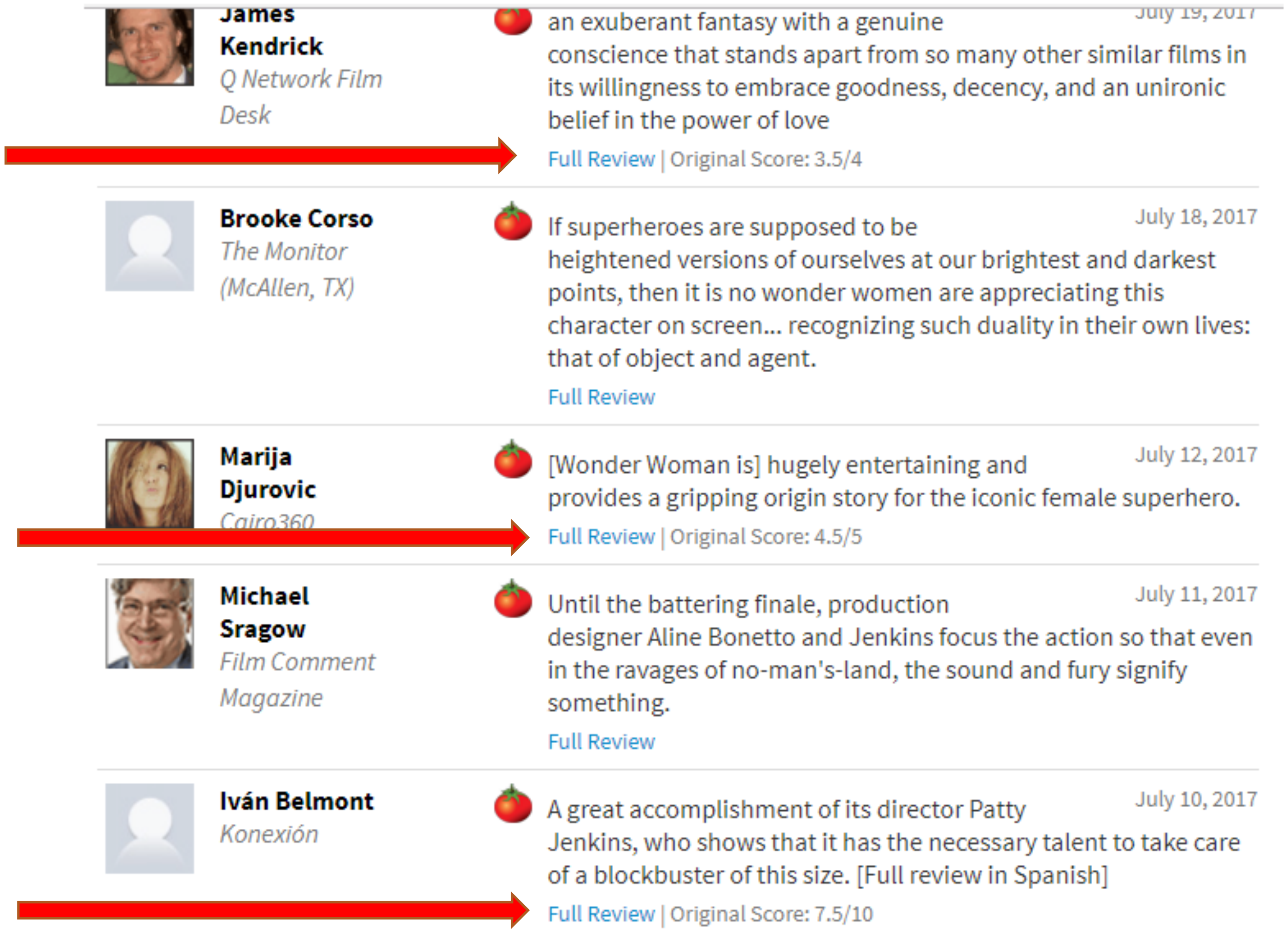
Some of the non-professional reviewers are considered as "super reviewers"

Also professional

Sentiment

Sentiment

Sentiment scores



James Kendrick
Q Network Film Desk
July 19, 2017
an exuberant fantasy with a genuine conscience that stands apart from so many other similar films in its willingness to embrace goodness, decency, and an unironic belief in the power of love
Full Review | Original Score: 3.5/4

Brooke Corso
The Monitor (McAllen, TX)
July 18, 2017
If superheroes are supposed to be heightened versions of ourselves at our brightest and darkest points, then it is no wonder women are appreciating this character on screen... recognizing such duality in their own lives: that of object and agent.
Full Review

Marija Djurovic
Cairo360
July 12, 2017
[Wonder Woman is] hugely entertaining and provides a gripping origin story for the iconic female superhero.
Full Review | Original Score: 4.5/5

Michael Sragow
Film Comment Magazine
July 11, 2017
Until the battering finale, production designer Aline Bonetto and Jenkins focus the action so that even in the ravages of no-man's-land, the sound and fury signify something.
Full Review

Iván Belmont
Konexión
July 10, 2017
A great accomplishment of its director Patty Jenkins, who shows that it has the necessary talent to take care of a blockbuster of this size. [Full review in Spanish]
Full Review | Original Score: 7.5/10

Sentiment

Sentiment

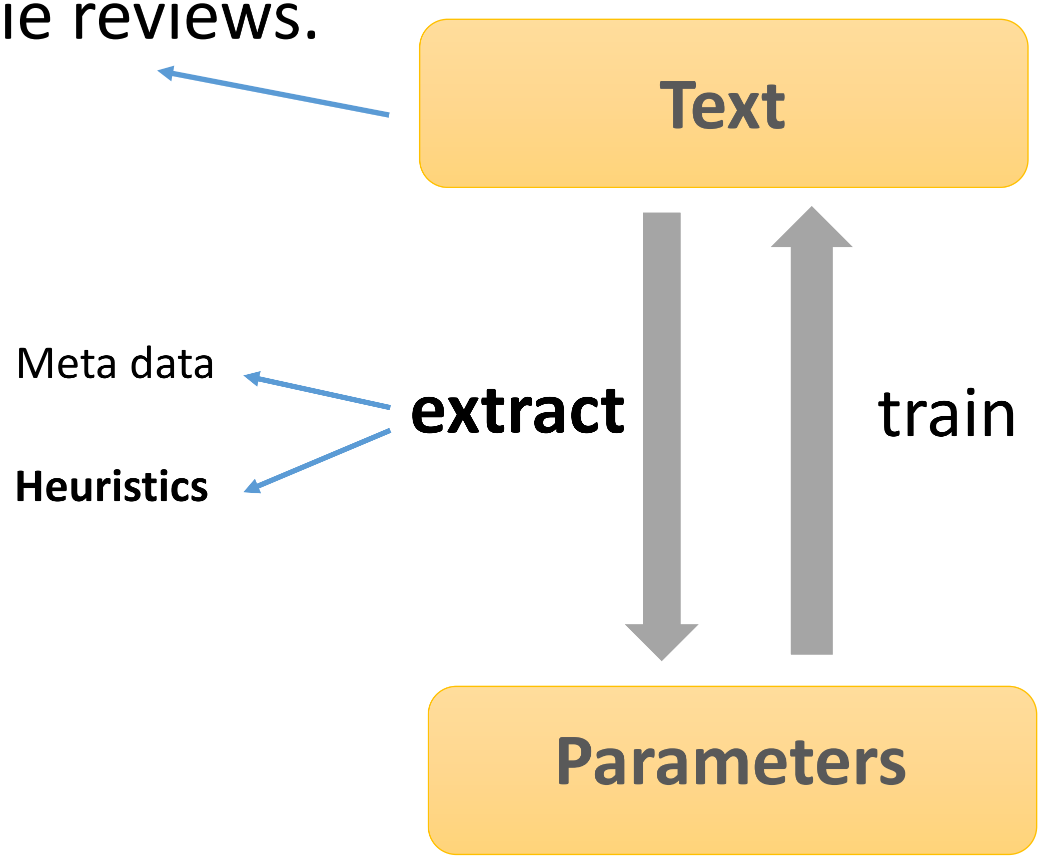
We normalized the critics scores to be on 0-5 scale

Negative	Neutral	Positive
0-2	3	4-5

Rotten-Tomatoes website.

7,500 movies.

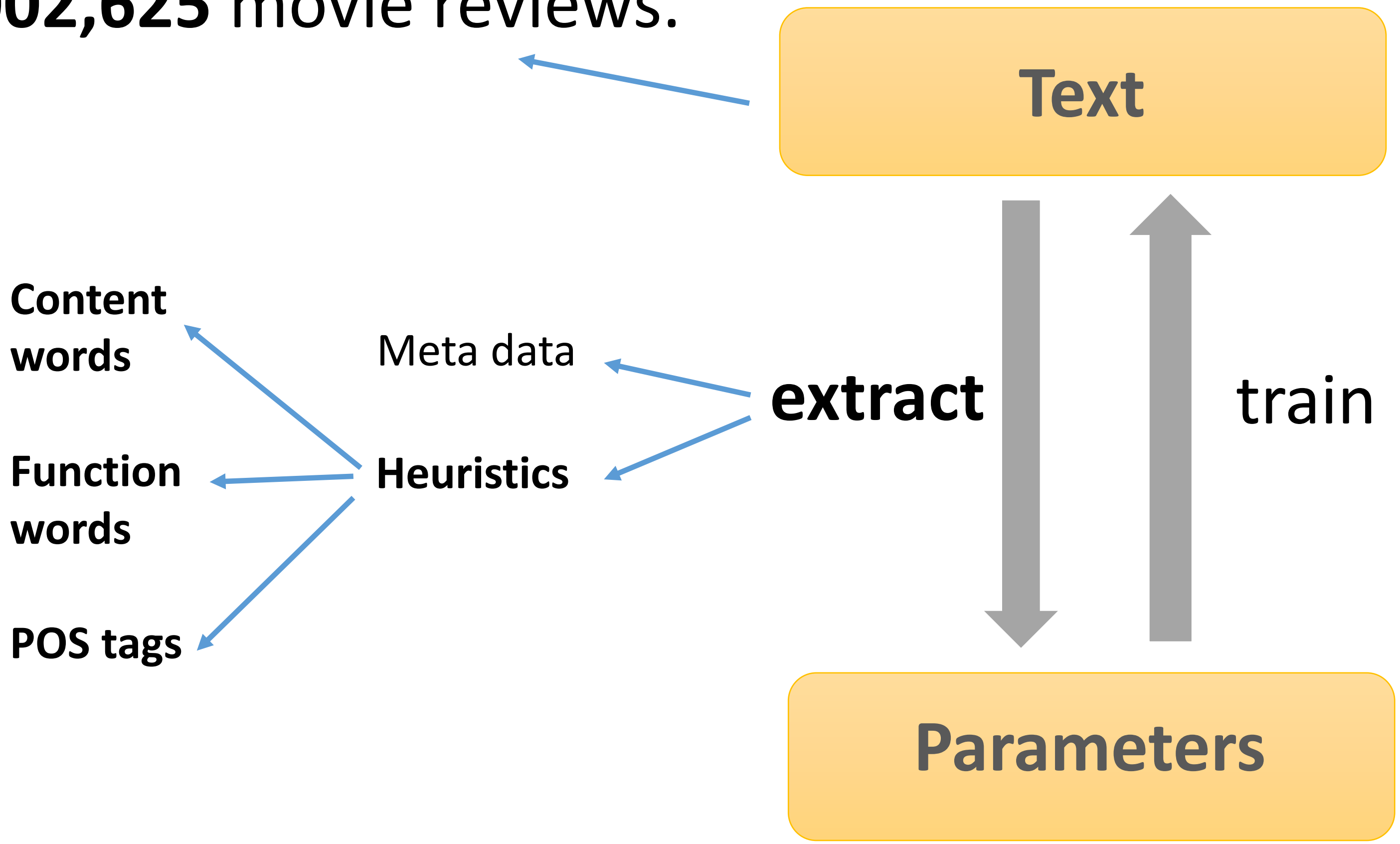
1,002,625 movie reviews.



Rotten-Tomatoes website.

7,500 movies.

1,002,625 movie reviews.



Theme

Content words

To determine the value for the theme parameter we searched for words that are related to the 4 topics and are common in our data set

Theme

Plot	Acting	Production	Effects
Story	Acting	Director	Effects
Storytelling	Cast	Directed	Song
Plot	Performance	Production	Music
Script	Play	co-production	Voice
Manuscript	Role		Visual
Tale	Miscasting		Soundtrack
Scene	Actor		Shot

Theme

Content words

To determine the value for the theme parameter we searched for words that are related to the 4 topics and are common in our data set

Theme

Plot	Acting	Production	Effects
Story	Acting	Director	Effects
Storytelling	Cast	Directed	Song
Plot	Performance	Production	Music
Script	Play	co-production	Voice
Manuscript	Role		Visual
Tale	Miscasting		Soundtrack
Scene	Actor		Shot

Each sentence was labeled with the category that has the most words in the sentence. Sentences that do not include any words from our lists are labeled as other

Personal Voice

Personal Pronouns

To determine weather a review is written in personal voice we search for words that express subjectivity

Personal

True	False
I My	Other cases

Descriptiveness

Distribution of part-of-speech tags

We assume that descriptive texts make heavy use of adjectives

Descriptive

True	False
% JJ $\geq$35	Other cases

Length

Length

≤ 10 words	11-20 words	21-40 words	> 40 words

Dataset Statistics

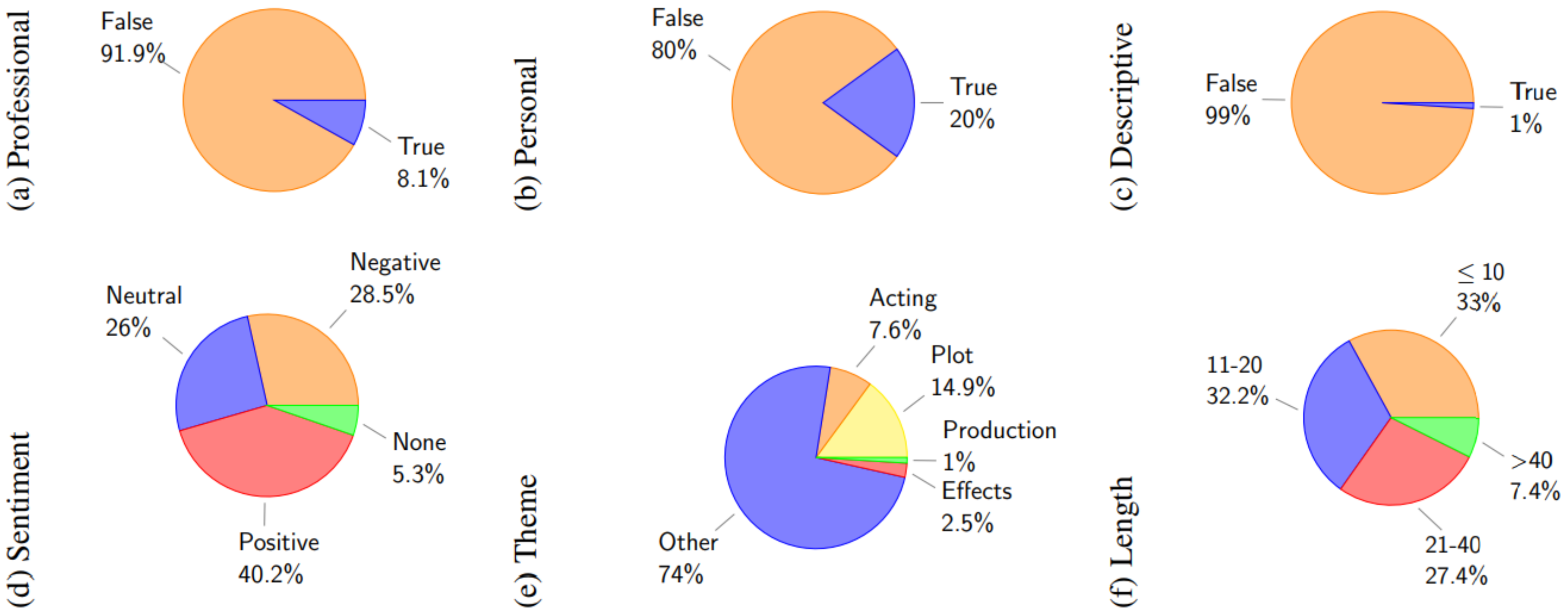
Our final data-set includes 2,773,435 sentences

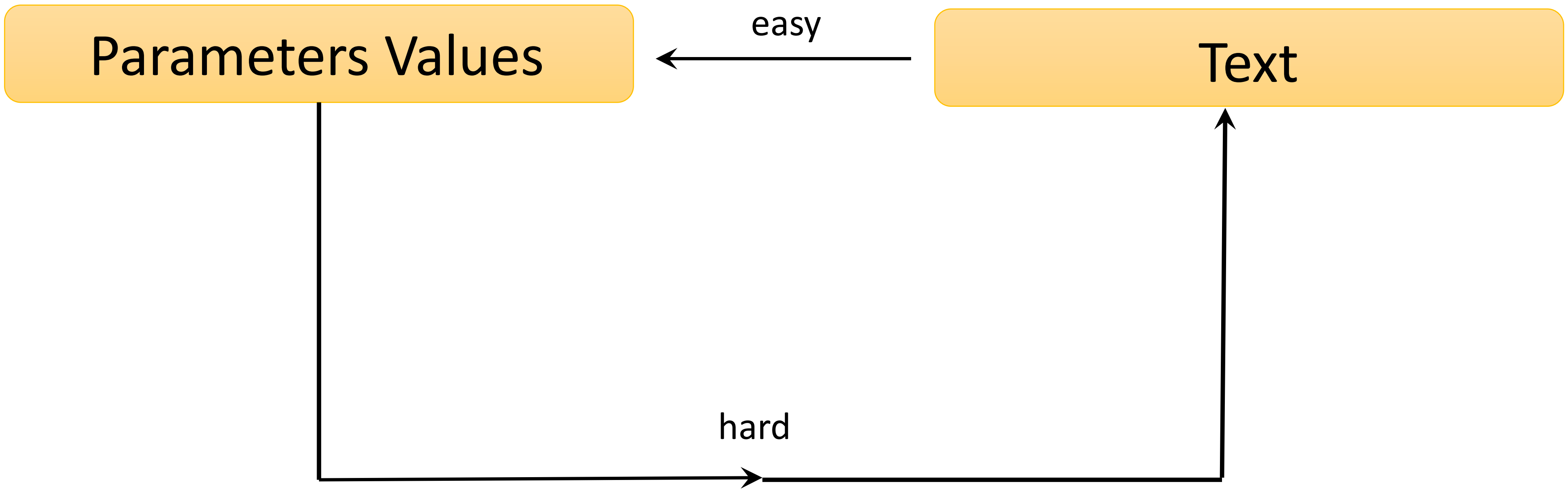
We divided the data set to training (~2.7M), development (~2K) and test (~2K) sets

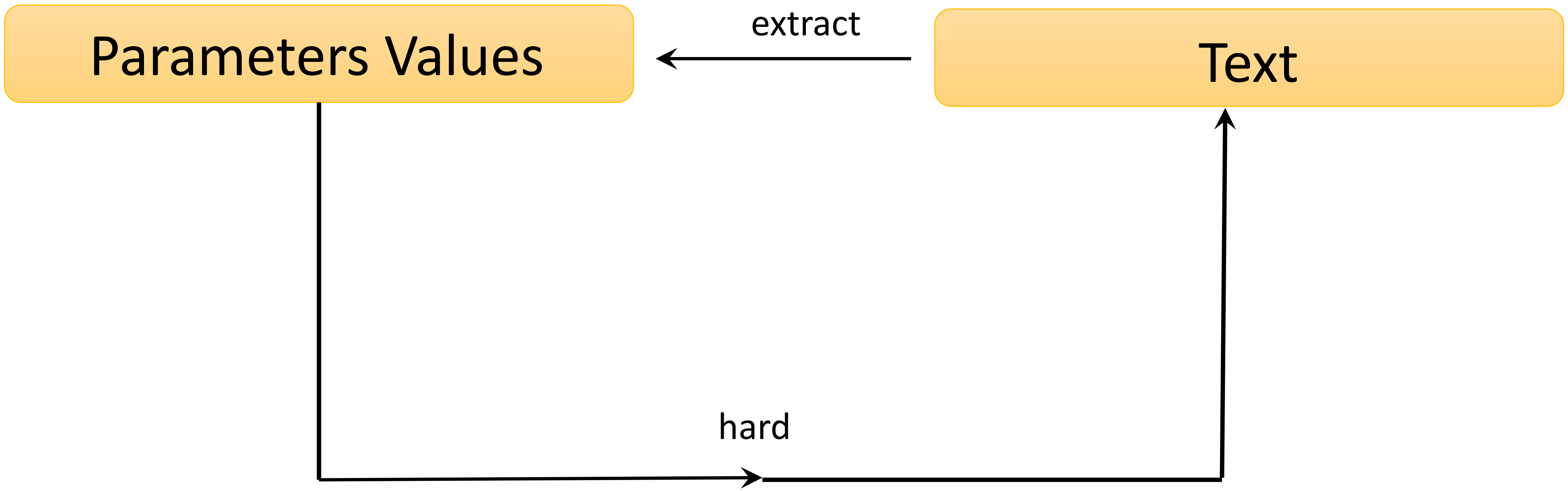
Each sentence is labeled with the 6 parameters

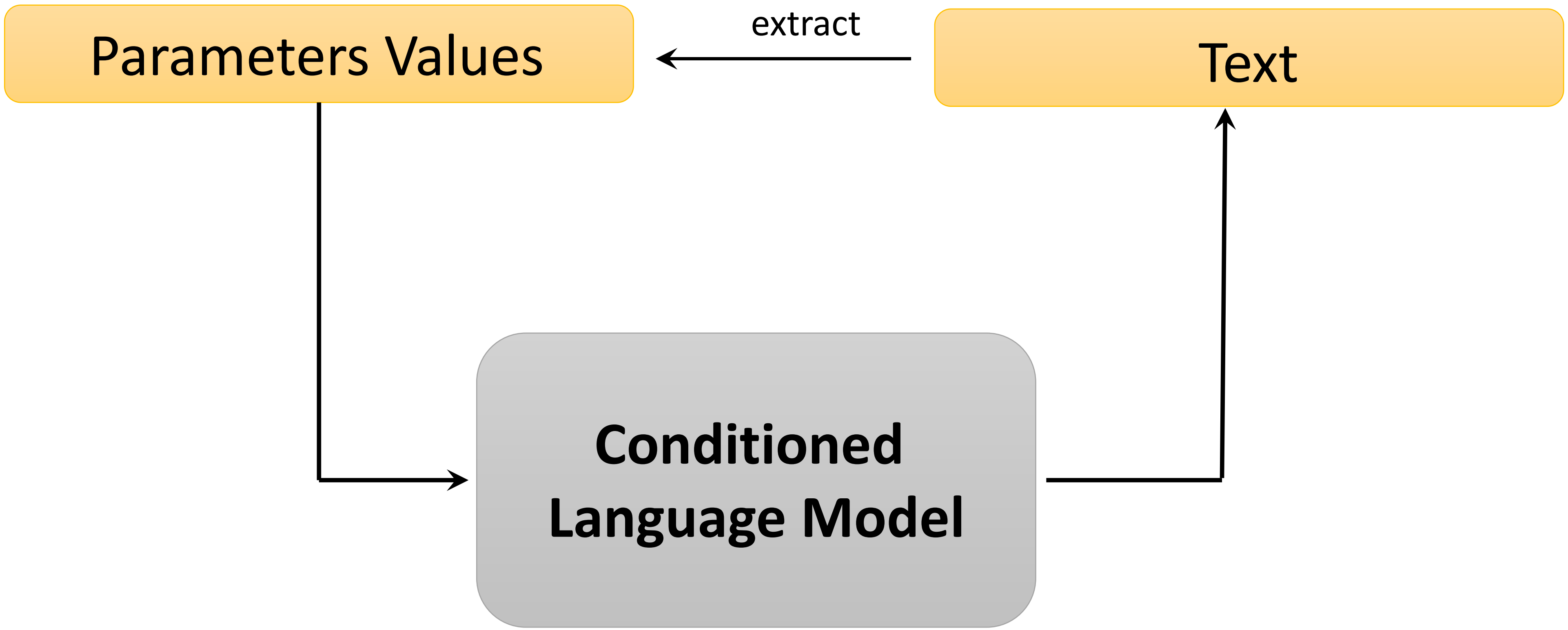
Parameters Values $\xleftarrow{\text{easy}}$ Text

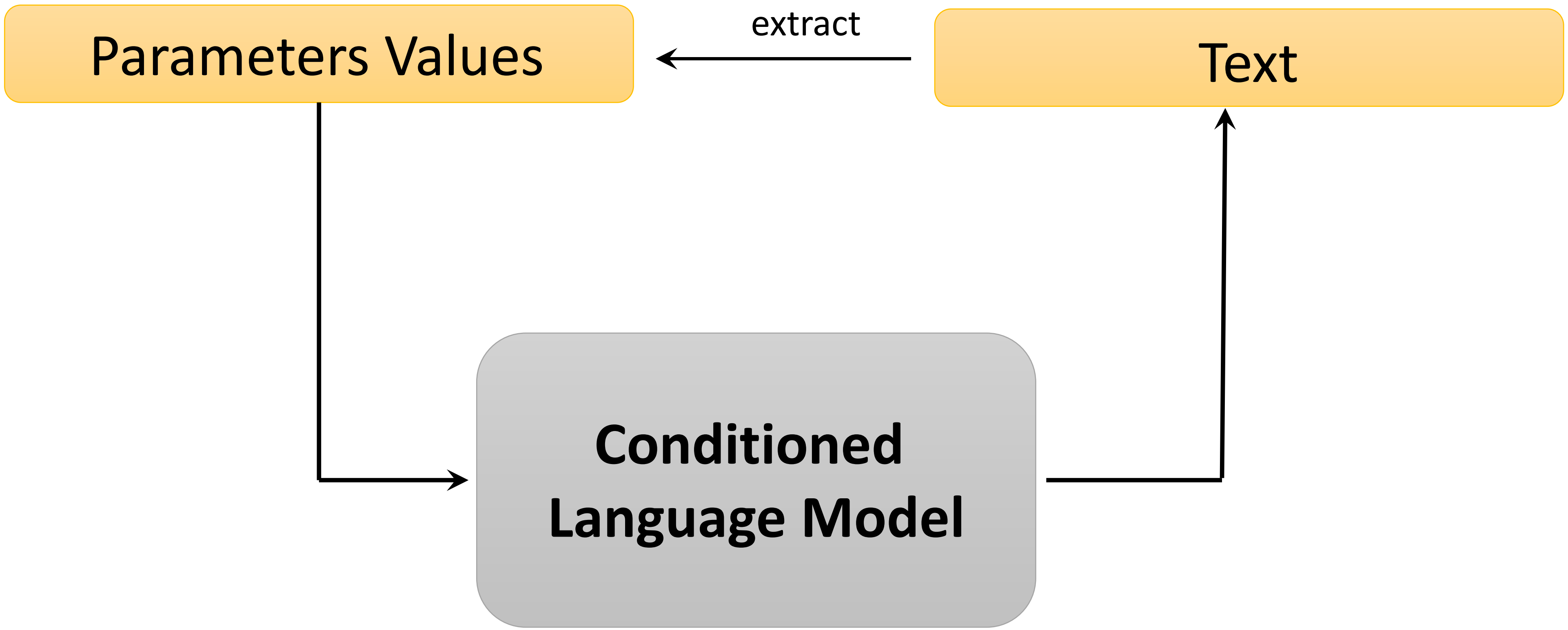
Parameters Values ← easy — Text




Parameters Values
easy
Text
hard


Parameters Values
extract
Text
hard


Parameters Values
extract
Text
Conditioned
Language Model



Does this work?

Examples of Generated Sentences

Parameter	Value
Professional	False
Personal	True
Length	11-20
Descriptive	True
Theme	Other
Sentiment	Negative

Examples of Generated Sentences

"Ultimately, I can honestly say that this movie is full of stupid stupid and stupid stupid stupid stupid stupid."

Parameter	**Value**
Professional	False
Personal	True
Length	11-20
Descriptive	True
Theme	Other
Sentiment	Negative

Examples of Generated Sentences

"Ultimately, I can honestly say that this movie is full of stupid stupid and stupid stupid stupid stupid stupid."

Parameter	Value
Professional	False
Personal	True
Length	11-20
Descriptive	True
Theme	Other
Sentiment	Negative

Examples of Generated Sentences

"Ultimately, I can honestly say that this movie is full of stupid stupid and stupid stupid stupid stupid stupid."

Parameter	Value
Professional	False ✔
Personal	True ✔
Length	11-20
Descriptive	True
Theme	Other
Sentiment	Negative

Examples of Generated Sentences

"Ultimately, I can honestly say that this movie is full of stupid stupid and stupid stupid stupid stupid stupid."

Parameter	**Value**	
Professional	False	✔
Personal	True	✔
Length	11-20	✔
Descriptive	True	
Theme	Other	
Sentiment	Negative	

Examples of Generated Sentences

"Ultimately, I can honestly say that this movie is full of stupid stupid and stupid stupid stupid stupid stupid."

Parameter	Value
Professional	False
Personal	True
Length	11-20
Descriptive	True
Theme	Other
Sentiment	Negative

Examples of Generated Sentences

"Ultimately, I can honestly say that this movie is full of stupid stupid and stupid stupid stupid stupid stupid."

Parameter	**Value**	
Professional	False	✔
Personal	True	✔
Length	11-20	✔
Descriptive	True	✔
Theme	Other	✔
Sentiment	Negative	

Examples of Generated Sentences

"Ultimately, I can honestly say that this movie is full of stupid stupid and stupid stupid stupid stupid stupid."

Parameter	**Value**
Professional	False
Personal	True
Length	11-20
Descriptive	True
Theme	Other
Sentiment	Negative

Examples of Generated Sentences

"Ultimately, I can honestly say that this movie is full of stupid stupid and stupid stupid stupid stupid stupid."

Parameter	Value	
Professional	False	✔
Personal	True	✔
Length	11-20	✔
Descriptive	True	✔
Theme	Other	✔
Sentiment	Negative	✔

"The film's simple, and a refreshing take on the complex family drama of the regions of human intelligence."

Parameter	Value
Professional	True
Personal	False
Length	11-20
Descriptive	False
Theme	Other
Sentiment	Positive

Examples of Generated Sentences

“Ultimately, I can honestly say that this movie is full of stupid stupid and stupid stupid stupid stupid stupid.”

Parameter	Value	
Professional	False	✔
Personal	True	✔
Length	11-20	✔
Descriptive	True	✔
Theme	Other	✔
Sentiment	Negative	✔

“The film’s simple, and a refreshing take on the complex family drama of the regions of human intelligence.”

Parameter	Value
Professional	True
Personal	False
Length	11-20
Descriptive	False
Theme	Other
Sentiment	Positive

We would like to quantitatively measure our model capabilities.

Common model failures

(via our split-and-rephrase example)

Split and Rephrase: Better Evaluation and a Stronger Baseline

Roee Aharoni & Yoav Goldberg
Computer Science Department
Bar-Ilan University
Ramat-Gan, Israel
{roee.aharoni,yoav.goldberg}@gmail.com

The Split and Rephrase Task

- Narayan, Gardent, Cohen & Shimorina, EMNLP 2017
- Dataset, evaluation method, baseline models
- Task definition: complex sentence -> several simple sentences with the same meaning
- Requires (a) **identifying** independent semantic units (b) **rephrasing** those units to single sentences

Alan Bean joined NASA in 1963 where he became a member of the Apollo 12 mission along with Alfred Worden as back up pilot and David Scott as commander .

↓

Alan Bean served as a crew member of Apollo 12 .
Alfred Worden was the backup pilot of Apollo 12 .
Apollo 12 was commanded by David Scott .
Alan Bean was selected by Nasa in 1963 .

WebSplit Dataset Construction
(Narayan et al. 2017)

Simple RDF Triples (facts from DBpedia)

Simple Sentences

<Alan_Bean | nationality | United_States> → Alan Bean is a US national.

<Alan_Bean | mission | Apollo_12> → Alan Bean was on the crew of Apollo 12.

<Alan_Bean | NASA selection | 1963> → Alan Bean was hired by NASA in 1963.

Matching via RDFs ↕ ~1M examples

Sets of RDF triples

Complex Sentences

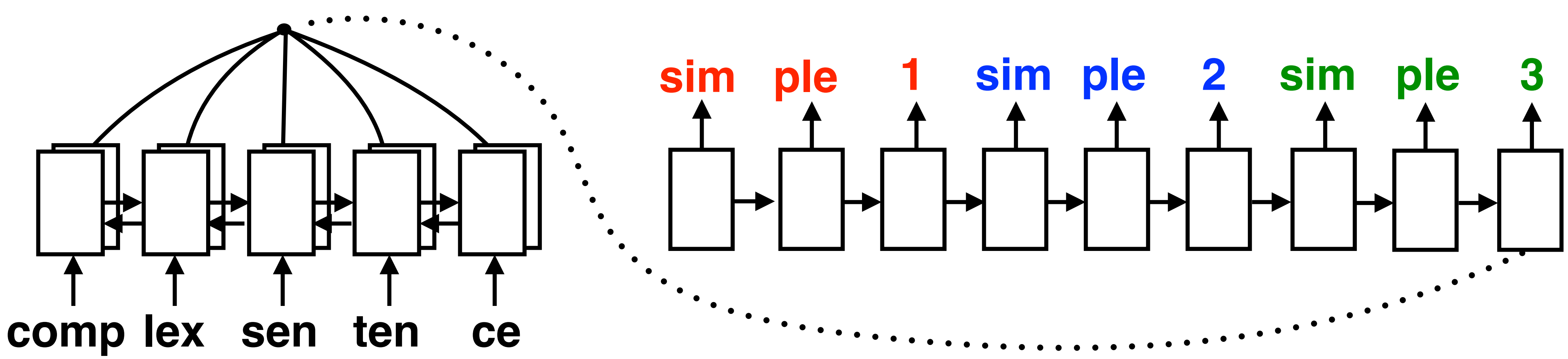
<Alan_Bean | nationality | United_States,
Alan_Bean | mission | Apollo_12,
Alan_Bean | NASA selection | 1963>

→

Alan Bean, born in the United States, was selected by NASA in 1963 and served as a crew member of Apollo 12.

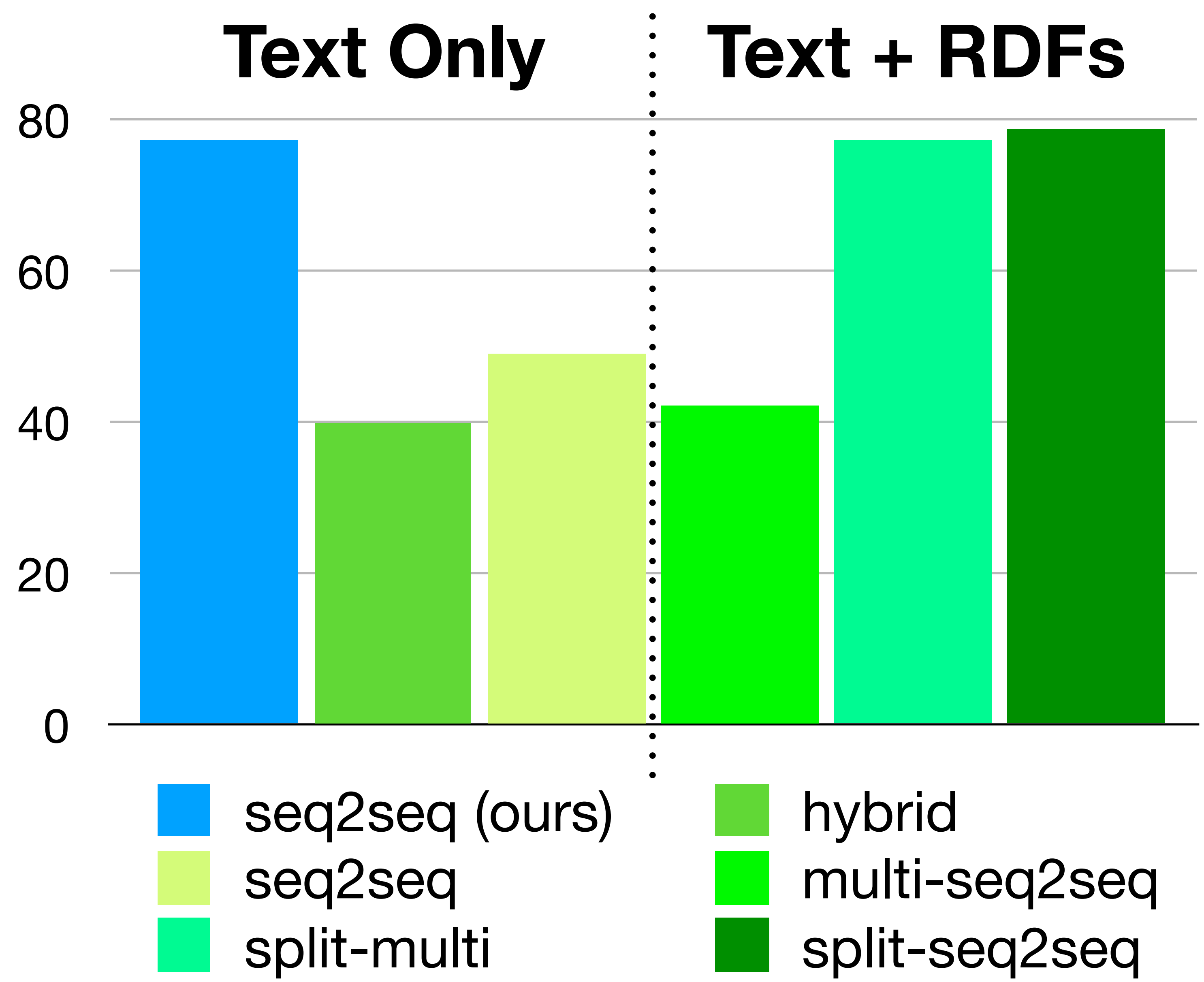
Preliminary Experiments

- ~1M training examples
- "Vanilla" LSTM seq2seq with attention
- Shared vocabulary between the encoder and the decoder
- Simple sentences predicted as a single sequence
- Evaluated using single-sentence, multi-reference BLEU as in Narayan et al. 2017



Preliminary Results

- Our simple seq2seq baseline outperform **all but one** of the baselines from Narayan et al. 2017
- Their best baselines were using the RDF structures as additional information
- **Do the simple seq2seq model really performs so well?**



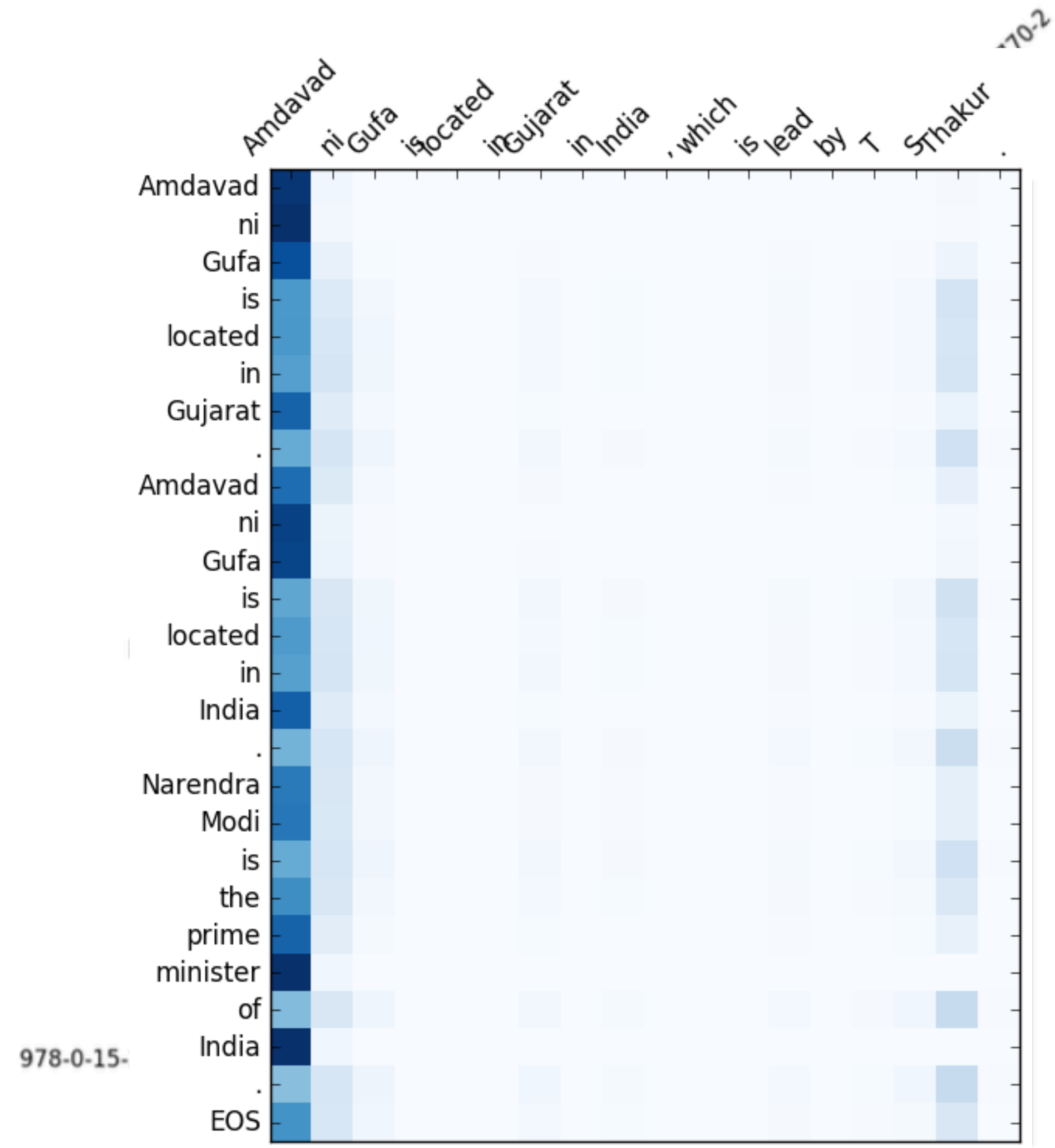
BLEU can be Misleading

- In spite of the high BLEU scores, our neural models suffer from:
  - Missing facts - appeared in the input but not in the output
  - Unsupported facts - appeared in the output but not in the input
  - Repeated facts - appeared several times in the output

Input	Prediction
A Fortress of Grey Ice with ISBM 0-7653-0633-6 has 672 pages .	J.V. Jones authored A Fortress of Grey Ice . A Fortress of Grey Ice has 672 pages .
The address , 11 Diagonal Street is located in South Africa where the leader is Cyril Ramaphosa and some Asian South Africans live .	The address , 11 Diagonal Street is located in South Africa . The leader of South Africa is called Cyril Ramaphosa . The leader of South Africa is called Cyril Ramaphosa . The leader of South Africa is called Cyril Ramaphosa .

A Closer Look

- Visualizing the attention weights we find an unexpected pattern
- The network mainly **attends to a single token** instead of spreading the attention
- This token was usually a part of the **first mentioned entity**
- Consistent among different input examples



Testing for Over-Memorization

- In this stage we suspect that the network heavily **memorizes** entity-fact pairs
- We test this by introducing it with inputs consisting of repeated entities alone
- The network indeed generates facts it memorized about those specific entities

Input	Prediction
Alan Shepard Alan Shepard Alan Shepard	

Searching for the Cause: Dataset Artifacts

- The original dataset included overlap between the training/development/test sets
- When looking at the complex sentences side, there is no overlap
- On the other hand, **most of the simple sentences** did overlap (~90%)
- Makes memorization very effective - “leakage” from train on the target side

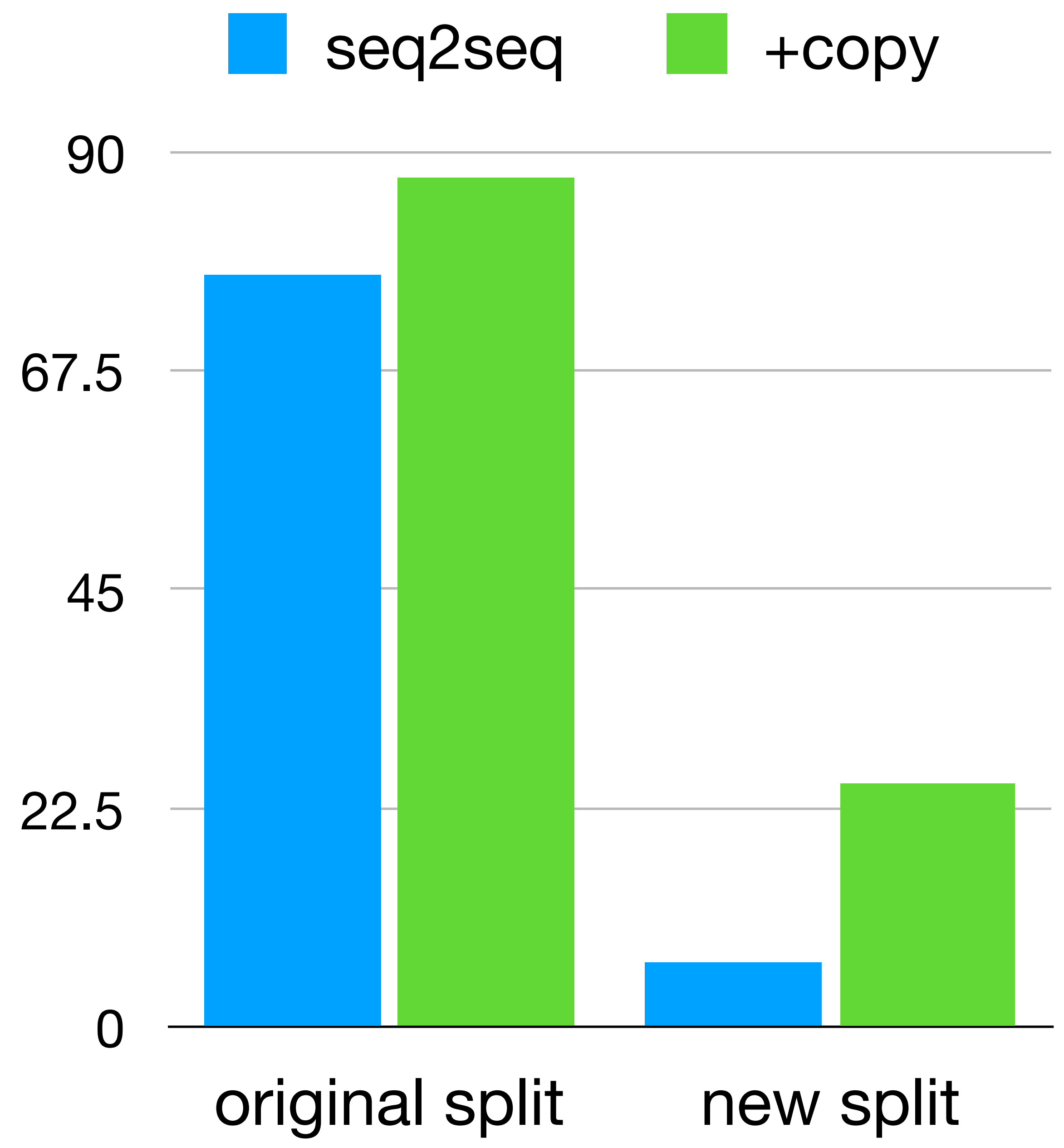
New Data Split

- To remedy this, we construct a new data split by using the RDF information:
  - Ensuring that all RDF **relation types** appear in the training set (enable generalization)
  - Ensuring that no RDF triple (fact) appears in two different sets (reduce memorization)
- The resulting dataset has no overlapping simple sentences
- Has more unknown symbols in dev/test - **need better models!**

	Original Split	**New Split**
unique dev simple sentences in train	90.9%	**0.09%**
unique test simple sentences in train	89.8%	**0%**

Results - New Split

- Baseline seq2seq models completely break (BLEU < 7) on the new split
- Copy mechanism helps to generalize
- Much lower than the original benchmark - memorization was crucial for the high BLEU



Takeaway

- **Creating datasets is hard!**
  - Think how models can “cheat"
  - Create a challenging evaluation environment to capture generalization
  - Look for leakage of train to dev/test
- **Numbers can be misleading!**
  - Look at the data
  - Look at the model
  - Error analysis

Evaluation

- Unsolved problem.
- Using BLEU, ROUGE (this is bad)
- Using human-eval (methodology varies)
- Using NLP classifiers on the generated output.

Two more techniques

Copy mechanism

Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond

Ramesh Nallapati
IBM Watson
nallapati@us.ibm.com

Bowen Zhou
IBM Watson
zhou@us.ibm.com

Cicero dos Santos
IBM Watson
cicerons@us.ibm.com

Çağlar Gůlçehre
Université de Montréal
gulcehrc@iro.umontreal.ca

Bing Xiang
IBM Watson
bingxia@us.ibm.com

Get To The Point: Summarization with Pointer-Generator Networks

Abigail See
Stanford University
abisee@stanford.edu

Peter J. Liu
Google Brain
peterjliu@google.com

Christopher D. Manning
Stanford University
manning@stanford.edu

- Allow the model to copy words from the source instead of generating them.

- Improves summarization / generation tasks.

Checklist model

Globally Coherent Text Generation with Neural Checklist Models

Chloé Kiddon   Luke Zettlemoyer   Yejin Choi
Computer Science & Engineering
University of Washington
{chloe, lsz, yejin}@cs.washington.edu

- Keep a soft-track of what's already covered in the source.
- Don't repeat yourself.

Recap

Neural NLG

- We have systems that can map inputs to outputs.

- Inputs can be:
  - nothing
  - set
  - table (key=value pairs)
  - graph
  - sentence
  - tree
  - image
  - sequence of images
  - combinations of the above

- Outputs can be:
  - sentences
  - trees
  - paragraphs...

We cannot fully understand the mapping. Data driven.

Neural NLG

- Performs well on generating **fluent sentences**.
- My not be so great at understanding what's going on.

Classic NLG

- You know what you want to say.
- Focus on the **best way** of saying it in order to achieve a communication goal.

Today's Neural NLG

- Generate me some text given this input.
- Yay it looks readable!!

Content selection?

Neural Text Generation from Structured Data with Application to the Biography Domain

Rémi Lebret*
EPFL, Switzerland

David Grangier
Facebook AI Research

Michael Auli
Facebook AI Research

Content selection?

Neural Text Generation from Structured Data with Application to the Biography Domain

Model	Generated Sentence
Reference	frederick parker-rhodes (21 march 1914 – 21 november 1987) was an english linguist, plant pathologist, computer scientist, mathematician, mystic, and mycologist.
Baseline (Template KN)	frederick parker-rhodes (born november 21 , 1914 – march 2 , 1987) was an english cricketer .
Table NLM +Local (field, start)	frederick parker-rhodes (21 november 1914 – 2 march 1987) was an australian rules footballer who played with carlton in the victorian football league (vfl) during the XXXXs and XXXXs .
+ Global (field)	frederick parker-rhodes (21 november 1914 – 2 march 1987) was an english mycology and plant pathology , mathematics at the university of uk .
+ Global (field, word)	frederick parker-rhodes (21 november 1914 – 2 march 1987) was a british computer scientist , best known for his contributions to computational linguistics .

Table 4: First sentence from the current Wikipedia article about Frederick Parker-Rhodes and the sentences generated from the three versions of our table-conditioned neural language model (Table NLM) using the Wikipedia infobox seen in Figure 1.

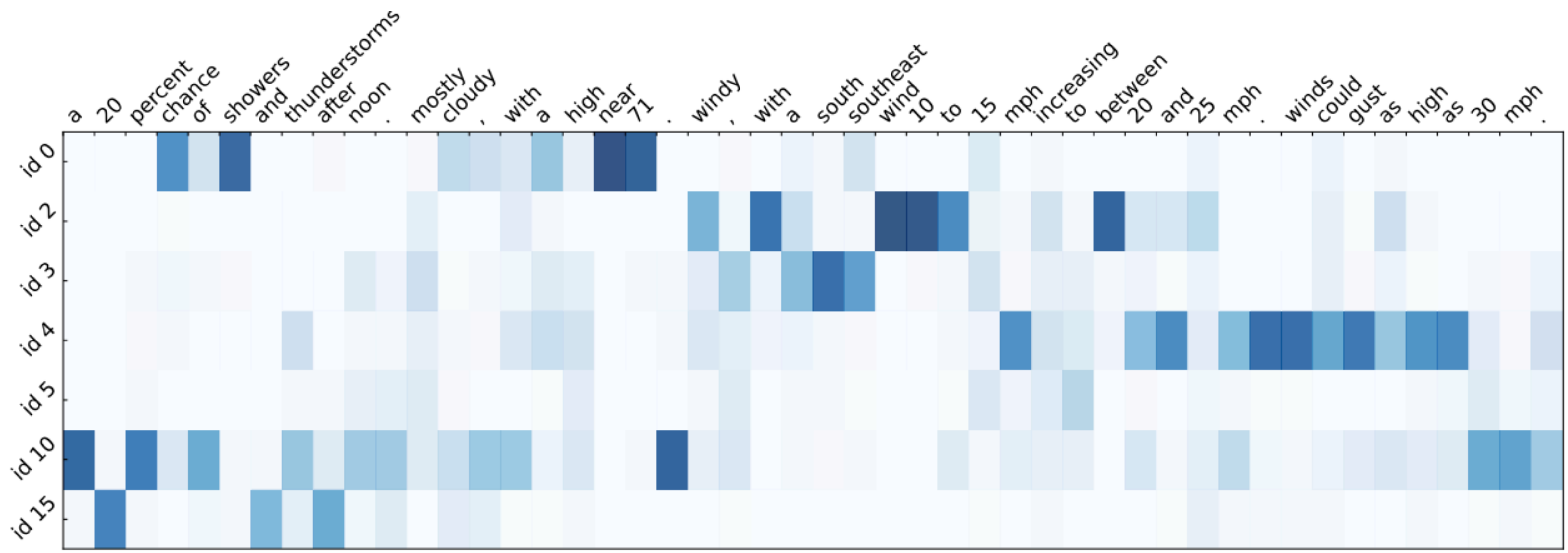
Content selection

What to talk about and how? Selective Generation using LSTMs with Coarse-to-Fine Alignment

Hongyuan Mei **Mohit Bansal** **Matthew R. Walter**
Toyota Technological Institute at Chicago
Chicago, IL 60637
{hongyuan,mbansal,mwalter}@ttic.edu

Content selection

What to talk about and how? Selective Generation using LSTMs with Coarse-to-Fine Alignment



Record details:
id-0: temperature(time=06-21, min=52, mean=63, max=71); id-2: windSpeed(time=06-21, min=8, mean=17, max=23);
id-3: windDir(time=06-21, mode=SSE); id-4: gust(time=06-21, min=0, mean=10, max=30);
id-5: skyCover(time=6-21, mode=50-75); id-10: precipChance(time=06-21, min=19, mean=32, max=73);
id-15: thunderChance(time=13-21, mode=SChc)

Figure 3: An example generation for a set of records from WEATHERGOV.

Challenges

- Very many challenges. Here are a few:
  - Generating longer text.
  - Tracking references to entities.
  - Staying cohesive.
- Finer grained control (content selection, ordering, ...)
- Evaluation!

Challenges in Data-to-Document Generation

Sam Wiseman and **Stuart M. Shieber** and **Alexander M. Rush**
School of Engineering and Applied Sciences
Harvard University
Cambridge, MA, USA
{swiseman,shieber,srush}@seas.harvard.edu

The Utah Jazz (38 - 26) defeated the Houston Rockets (38 - 26) 117 - 91 on Wednesday at Energy Solutions Arena in Salt Lake City . The Jazz got out to a quick start in this one , out - scoring the Rockets 31 - 15 in the first quarter alone . Along with the quick start , the Rockets were the superior shooters in this game , going 54 percent from the field and 43 percent from the three - point line , while the Jazz went 38 percent from the floor and a meager 19 percent from deep . The Rockets were able to out - rebound the Rockets 49 - 49 , giving them just enough of an advantage to secure the victory in front of their home crowd . The Jazz were led by the duo of Derrick Favors and James Harden . Favors went 2 - for - 6 from the field and 0 - for - 1 from the three - point line to score a game - high of 15 points , while also adding four rebounds and four assists ....

Figure 2: Example document generated by the Conditional Copy system with a beam of size 5. Text that accurately reflects a record in the associated box- or line-score is highlighted in blue, and erroneous text is highlighted in red.

Data-to-Text Generation with Content Selection and Planning

Ratish Puduppully and **Li Dong** and **Mirella Lapata**
Institute for Language, Cognition and Computation
School of Informatics, University of Edinburgh
10 Crichton Street, Edinburgh EH8 9AB
r.puduppully@sms.ed.ac.uk li.dong@ed.ac.uk mlap@inf.ed.ac.uk

The Golden State Warriors (10–2) defeated the Boston Celtics (6–6) 104–88. Klay Thompson scored 28 points (12–21 FG, 3–6 3PT, 1–1 FT) to go with 4 rebounds. Kevin Durant scored 23 points (10–13 FG, 1–2 3PT, 2–4 FT) to go with 10 rebounds. Isaiah Thomas scored 18 points (4–12 FG, 1–6 3PT, 9–9 FT) to go with 2 rebounds. Avery Bradley scored 17 points (7–15 FG, 2–4 3PT, 1–2 FT) to go with 10 rebounds. Stephen Curry scored 16 points (7–20 FG, 2–10 3PT, 0–0 FT) to go with 3 rebounds. Terry Rozier scored 11 points (3–5 FG, 2–3 3PT, 3–4 FT) to go with 7 rebounds. The Golden State Warriors' next game will be at home against the Dallas Mavericks, while the Boston Celtics will travel to play the Bulls.
The Golden State Warriors defeated the Boston Celtics 104–88 at TD Garden on Friday. The Warriors (10–2) came into this game winners of five of their last six games, but the Warriors (6–6) were able to pull away in the second half. Klay Thompson led the way for the Warriors with 28 points on 12–of–21 shooting, while Kevin Durant added 23 points, 10 rebounds, seven assists and two steals. Stephen Curry added 16 points and eight assists, while Draymond Green rounded out the box score with 11 points, eight rebounds and eight assists. For the Celtics, it was Isaiah Thomas who shot just 4–of–12 from the field and finished with 18 points. Avery Bradley added 17 points and 10 rebounds, while the rest of the Celtics combined to score just seven points. Boston will look to get back on track as they play host to the 76ers on Friday.

Table 6: Example output from TEMPL (top) and NPC+CC (bottom). Text that accurately reflects a record in the associated box or line score is in blue, erroneous text is in red.

Neural NLG Research

- A lot of neural NLG research today:
  - How to best encode your input?
  - How to model the attention?

- The more important things (in my opinion):
  - How to get finer-grained control on the generated text?
  - How to construct a good dataset?
  - How to evaluate?
  - What is a good input representation (!= good input encoding)
  - Neural nets are great at surface realization. What about the other parts?

Neural NLG today

Input →

~~Content selection~~
~~Macro-planning~~
~~Packaging~~
~~Micro-planning~~
~~Referring Expression Generation~~
~~Linearization~~

→ Output

Neural NLG tomorrow?

Input → Content selection / Macro-planning / Packaging / Micro-planning / Referring Expression Generation / Linearization → Output

Content selection
Macro-planning
Packaging
Micro-planning
Referring Expression Generation
Linearization

Discussion?