UNIVERSITY of WASHINGTON

UWNLP

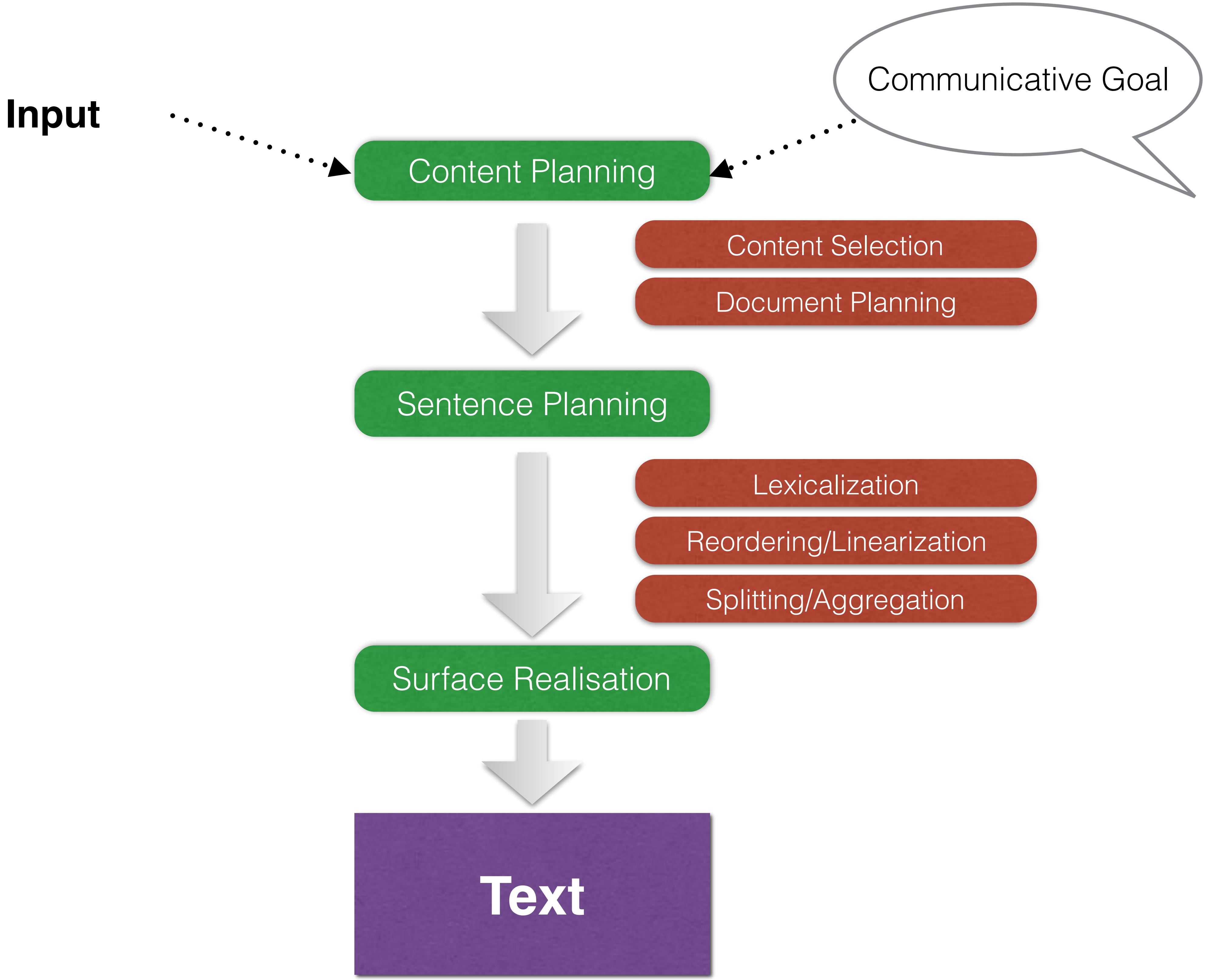
NNLG

Neural Natural Language Generation

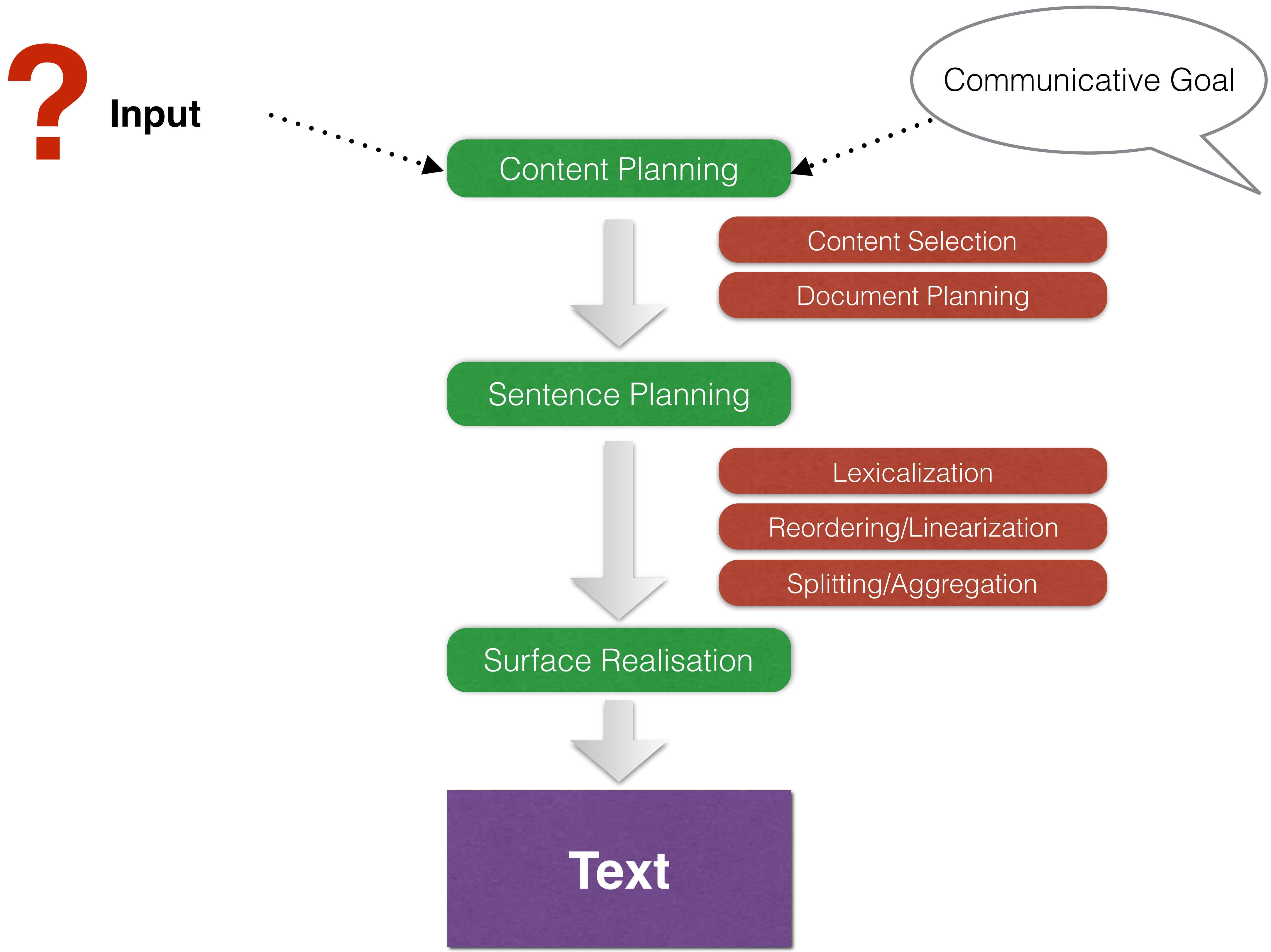
Yannis Konstas

Joint work with
Srinivasan Iyer, Mark Yatskar, Rik Koncel-Kedziorski, Li Zilles,
Luke Zettlemoyer, Yejin Choi, Hannaneh Hajishirzi

NLG Pipeline



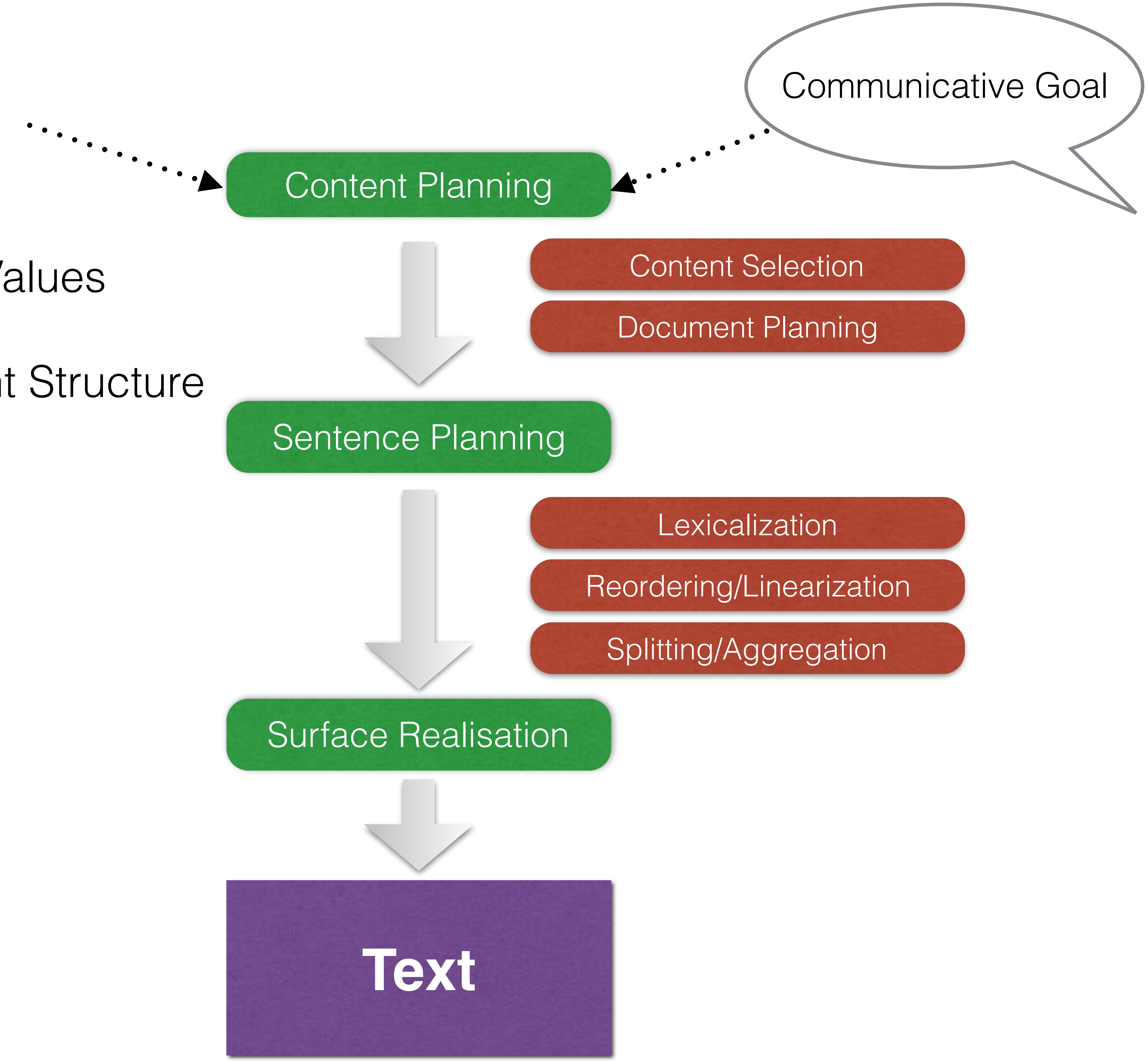
NLG Pipeline



NLG Pipeline

? **Input**

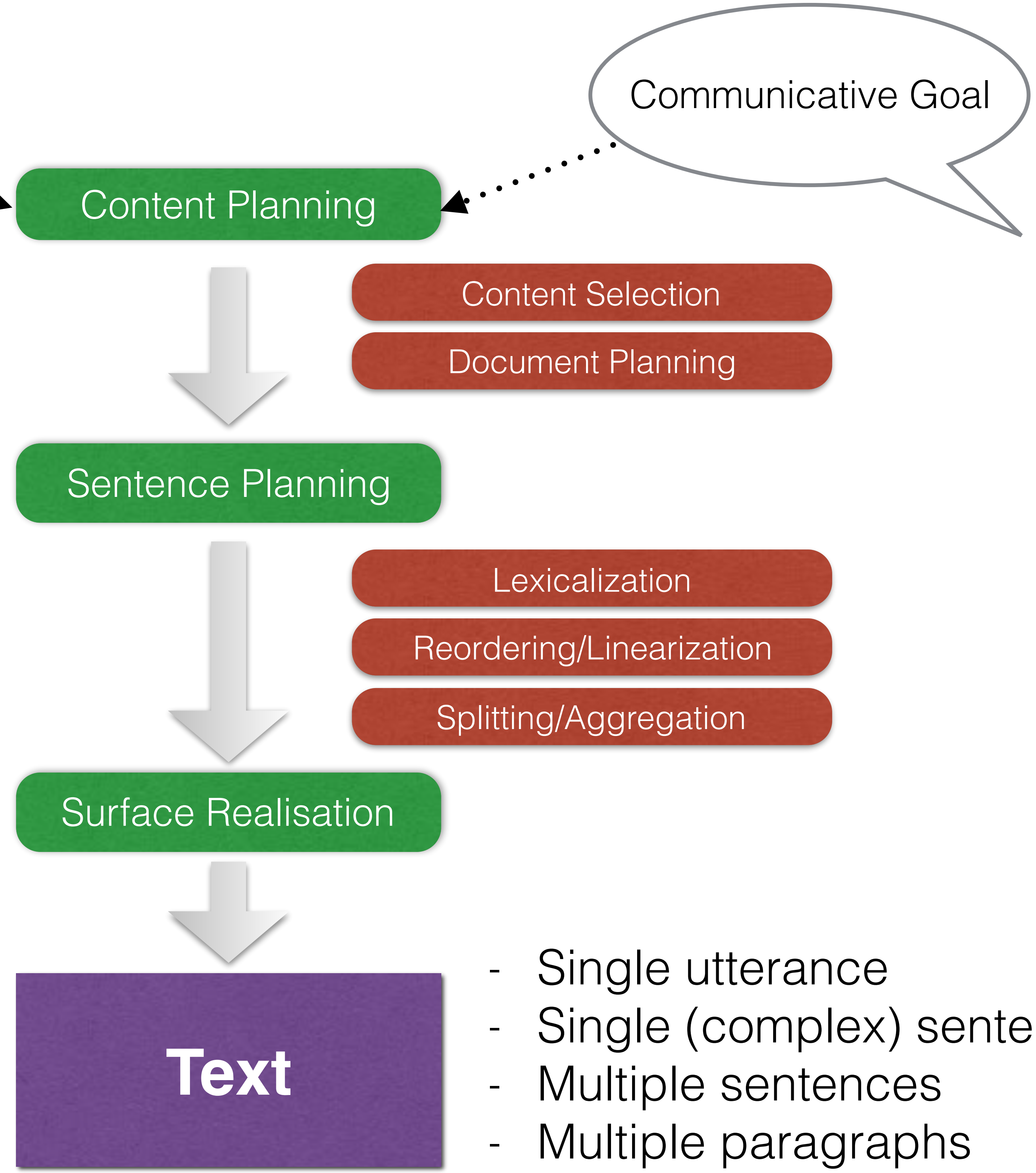
- Records / Fields / Values
- Source Code
- Predicate-Argument Structure
- Algebra equation
- Text / Script
- Multiple Sources



NLG Pipeline

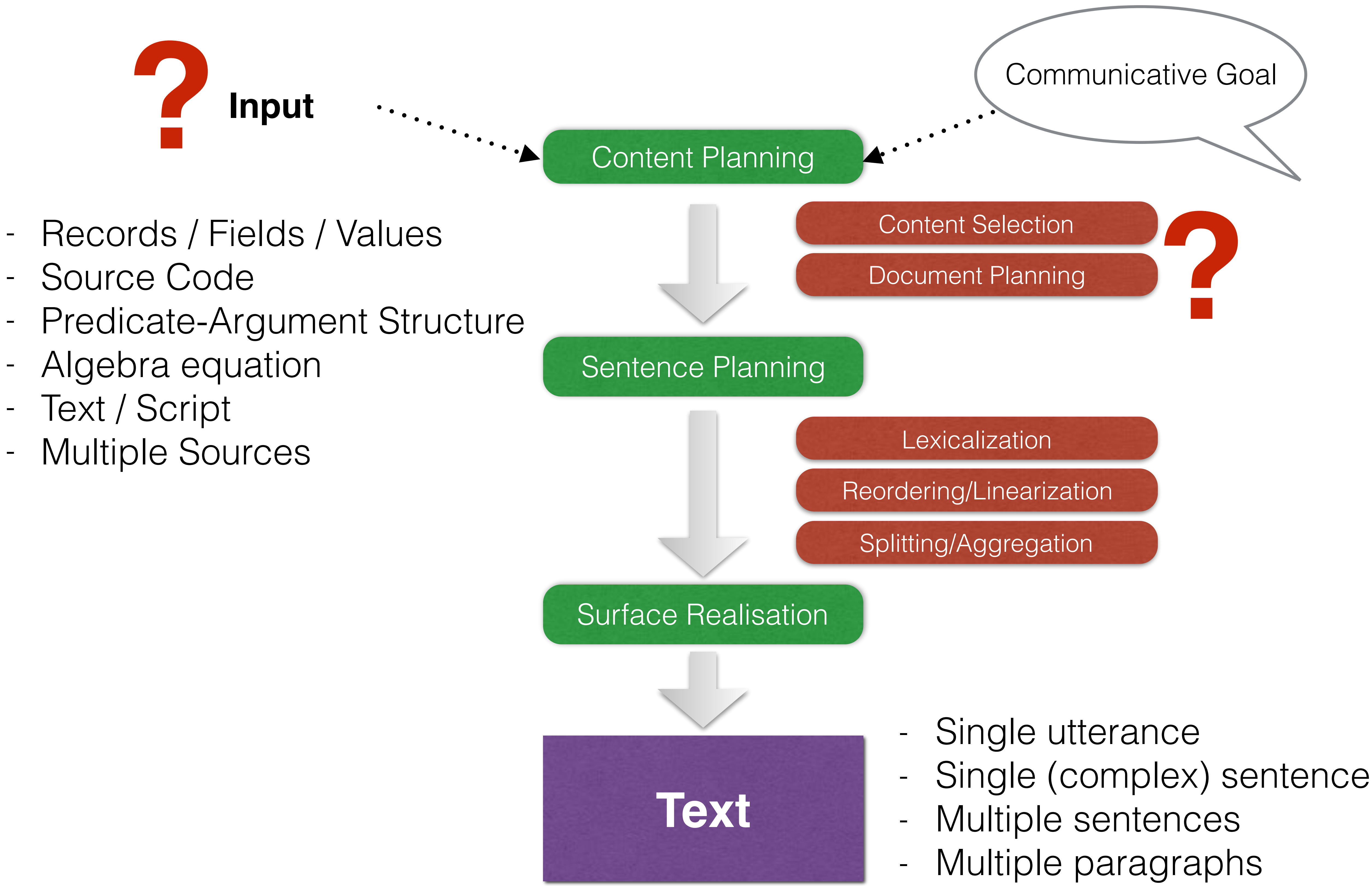
? **Input**

- Records / Fields / Values
- Source Code
- Predicate-Argument Structure
- Algebra equation
- Text / Script
- Multiple Sources



- Single utterance
- Single (complex) sentence
- Multiple sentences
- Multiple paragraphs

NLG Pipeline



NLG Pipeline

? **Input**

- Records / Fields / Values
- Source Code
- Predicate-Argument Structure
- Algebra equation
- Text / Script
- Multiple Sources

Communicative Goal

Content Planning

Content Selection

Document Planning

?

Sentence Planning

?

Lexicalization

Reordering/Linearization

Splitting/Aggregation

Surface Realisation

Text

- Single utterance
- Single (complex) sentence
- Multiple sentences
- Multiple paragraphs

NLG is everywhere

(A Global Model for Concept-to-Text Generation. Konstas and Lapata, JAIR 2013.)

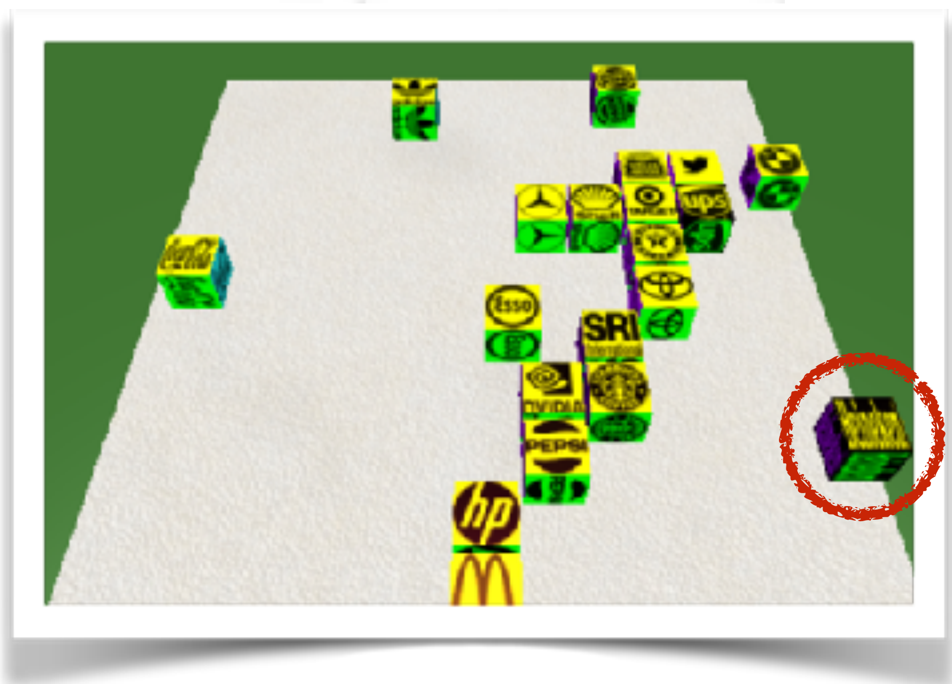
NLG is everywhere

Concept-to-Text Generation

Input: Machine-generated Representation

(A Global Model for Concept-to-Text Generation. Konstas and Lapata, JAIR 2013.)

NLG is everywhere



Concept-to-Text Generation

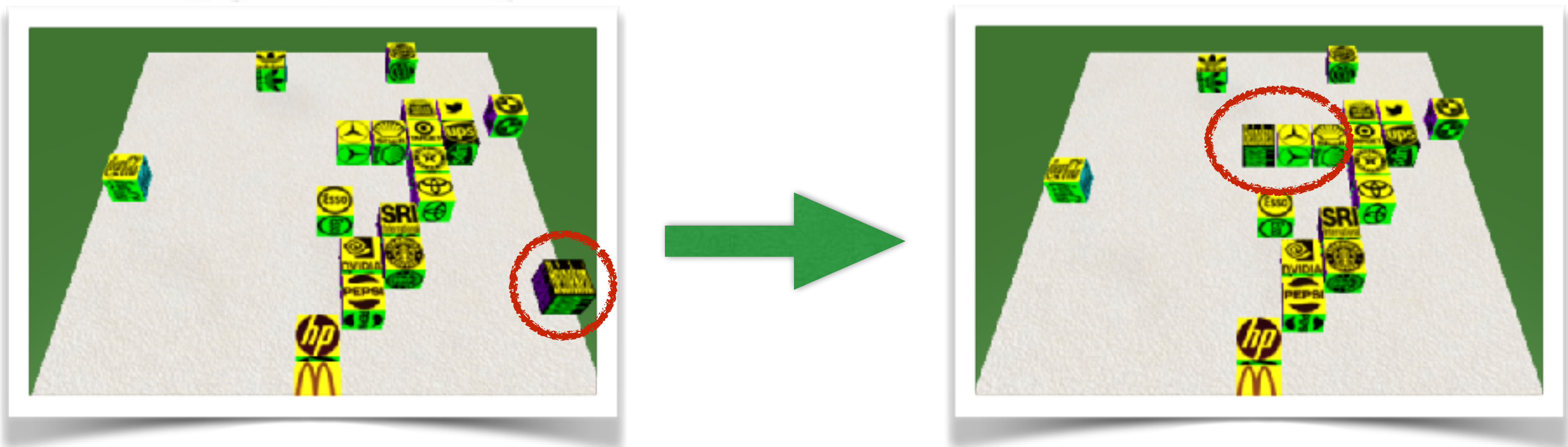
Input: Machine-generated Representation

(A Global Model for Concept-to-Text Generation. Konstas and Lapata, JAIR 2013.)

NLG is everywhere

Concept-to-Text Generation

Input: Machine-generated Representation

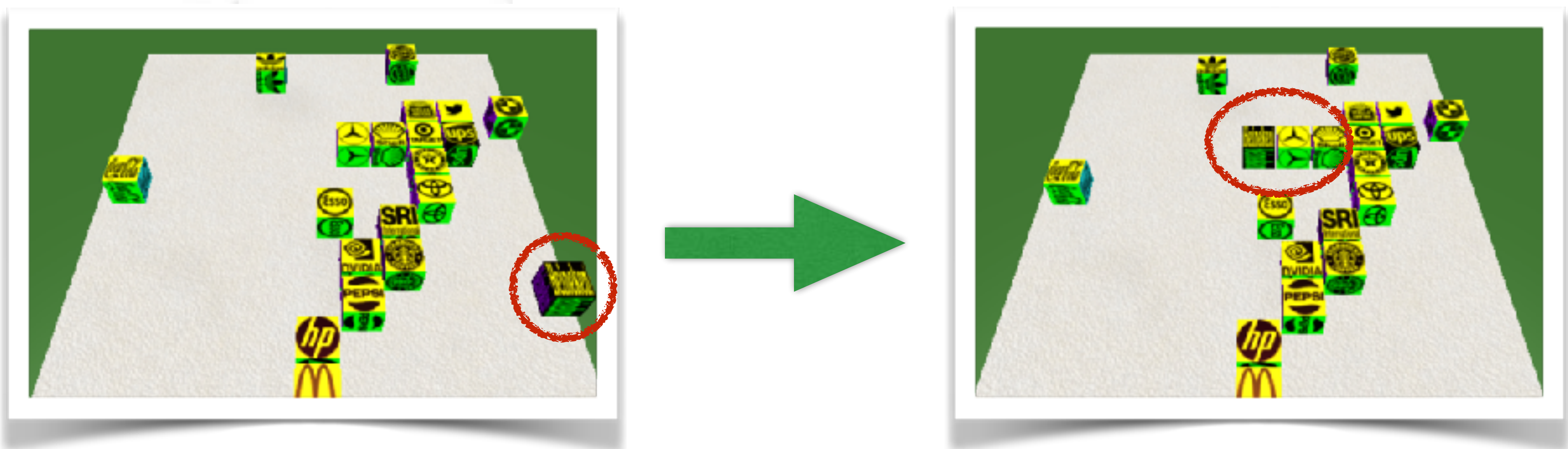


(A Global Model for Concept-to-Text Generation. Konstas and Lapata, JAIR 2013.)

NLG is everywhere

Concept-to-Text Generation

Input: Machine-generated Representation



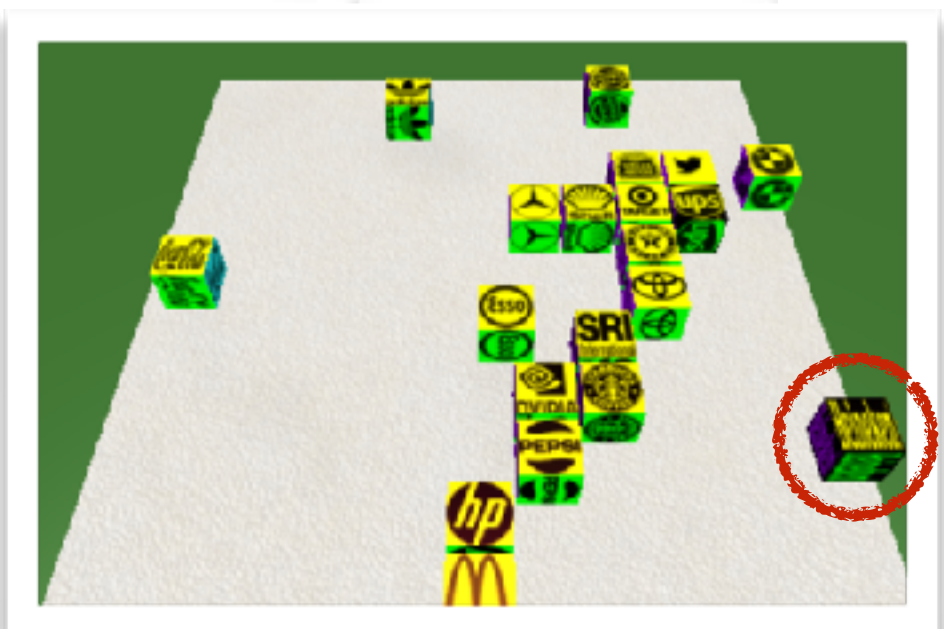
source	block:	hk		
target	block:	ms		
pos	RP:	W	scale:	small

(A Global Model for Concept-to-Text Generation. Konstas and Lapata, JAIR 2013.)

NLG is everywhere

Concept-to-Text Generation

Input: Machine-generated Representation



source	block:	hk		
target	block:	ms		
pos	RP:	W	scale:	small

Place the heineken block west of the mercedes block.

(A Global Model for Concept-to-Text Generation. Konstas and Lapata, JAIR 2013.)

NLG is everywhere

(Summarizing Source Code using a Neural Attention Model. Iyer, Konstas, Cheung, Zettlemoyer. ACL 2016.)

NLG is everywhere

Code-to-Text Generation

Input: Source Code

(Summarizing Source Code using a Neural Attention Model. Iyer, Konstas, Cheung, Zettlemoyer. ACL 2016.)

NLG is everywhere

Code-to-Text Generation

Input: Source Code

CODE-NN

(Summarizing Source Code using a Neural Attention Model. Iyer, Konstas, Cheung, Zettlemoyer. ACL 2016.)

NLG is everywhere

Code-to-Text Generation

Input: Source Code

CODE-NN

```
public int TextWidth (string text) {
  TextBlock t = new TextBlock();
  t.Text = text;
  return (int) Math.Ceiling(t.ActualWidth);
}
```

(Summarizing Source Code using a Neural Attention Model. Iyer, Konstas, Cheung, Zettlemoyer. ACL 2016.)

NLG is everywhere

Code-to-Text Generation

Input: Source Code

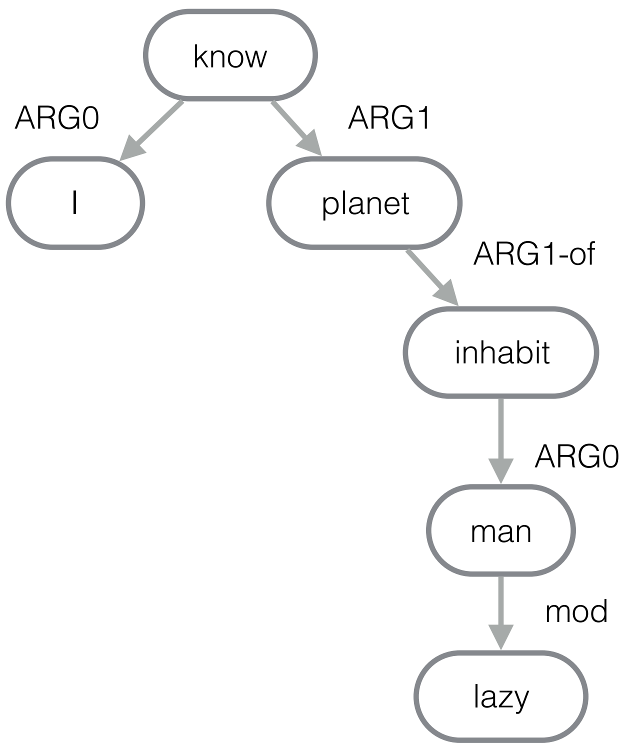
CODE-NN

```
public int TextWidth (string text) {
  TextBlock t = new TextBlock();
  t.Text = text;
  return (int) Math.Ceiling(t.ActualWidth);
}
```

Get rendered width of string rounded up to the nearest integer.

(Summarizing Source Code using a Neural Attention Model. Iyer, Konstas, Cheung, Zettlemoyer. ACL 2016.)

NLG is everywhere

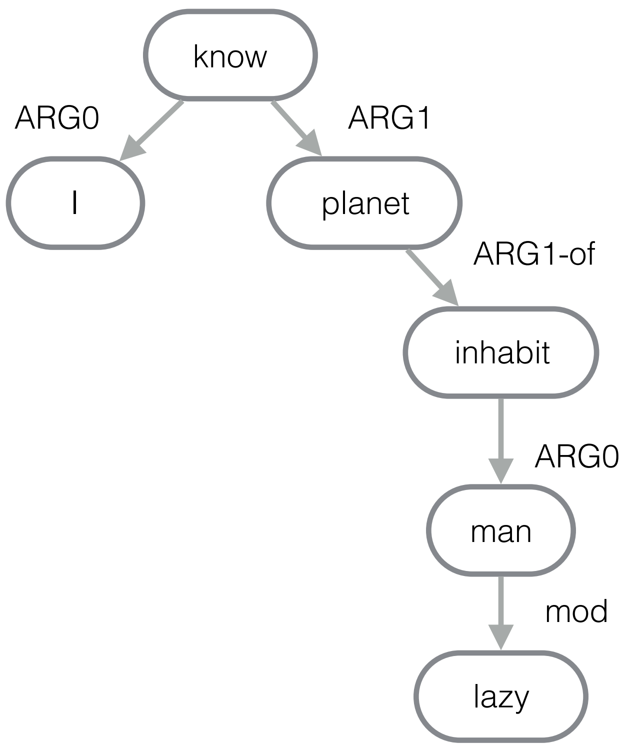


Meaning Representation Generation

Input: Predicate - Argument Structure

(Flanigan et al, NAACL 2016, Pourdamaghani and Knight, INLG 2016, Song et al, EMNLP 2016.)

NLG is everywhere



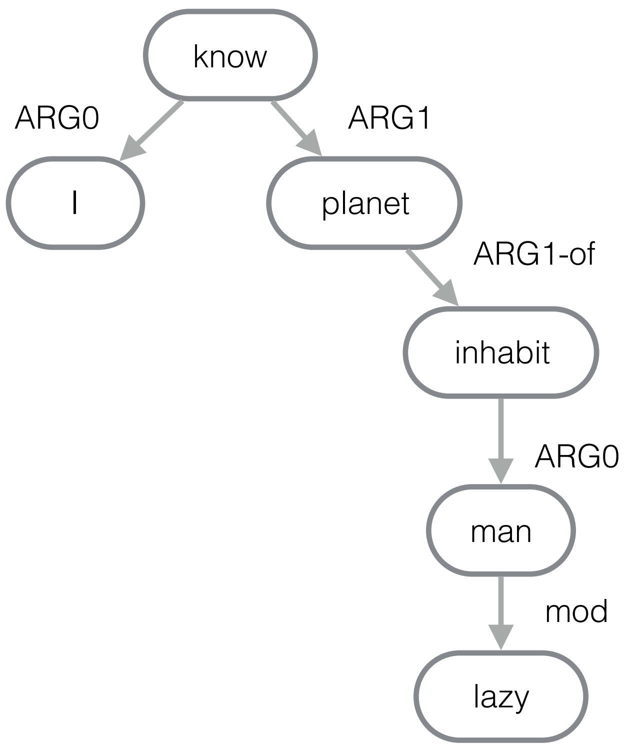
Meaning Representation Generation

Input: Predicate - Argument Structure

I knew a **planet** that was **inhabited** by a **lazy man**.

(Flanigan et al, NAACL 2016, Pourdamaghani and Knight, INLG 2016, Song et al, EMNLP 2016.)

NLG is everywhere



Meaning Representation Generation

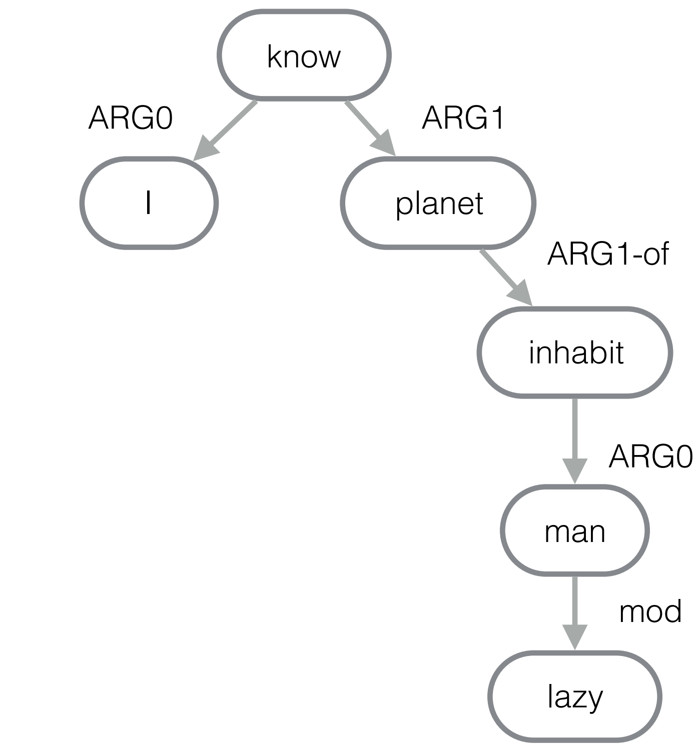
Input: Predicate - Argument Structure

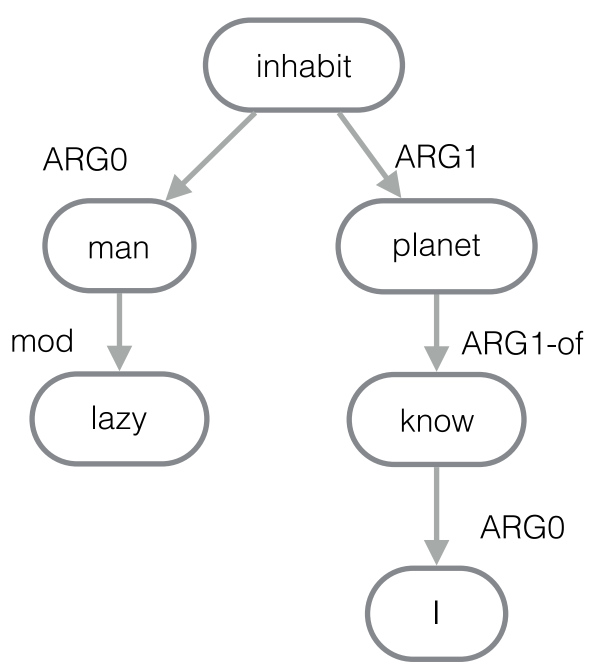
I knew a **planet** that was **inhabited** by a **lazy man**.

I have **known** a **planet** that was **inhabited** by a **lazy man**.

(Flanigan et al, NAACL 2016, Pourdamaghani and Knight, INLG 2016, Song et al, EMNLP 2016.)

NLG is everywhere





Meaning Representation Generation

Input: Predicate - Argument Structure

I knew a **planet** that was **inhabited** by a **lazy man**.

I have **known** a **planet** that was **inhabited** by a **lazy man**.

There is a **lazy man who inhabited** a **planet I know** about.

(Flanigan et al, NAACL 2016, Pourdamaghani and Knight, INLG 2016, Song et al, EMNLP 2016.)

NLG is everywhere

Instructional Text Generation

Input: Goal Cue - Bag of concepts

NLG is everywhere

Instructional Text Generation

Input: Goal Cue - Bag of concepts

Spanakopita
(Greek Spinach Pie)

Ingredients

3 tbsp olive oil
1 large onion, chopped
1 bunch green onions, chopped
2 cloves garlic, minced
2 pounds spinach
1/2 cup chopped fresh parsley
2 eggs
1/2 cup ricotta cheese
1 cup feta cheese
8 sheets filo dough
1/4 cup olive oil

(Globally Coherent Text Generation with Neural Checklist Models. Kiddon et al, EMNLP 2016.)

NLG is everywhere

Spanakopita (Greek Spinach Pie)

Ingredients

3 tbsp olive oil
1 large onion, chopped
1 bunch green onions, chopped
2 cloves garlic, minced
2 pounds spinach
1/2 cup chopped fresh parsley

2 eggs
1/2 cup ricotta cheese
1 cup feta cheese
8 sheets filo dough
1/4 cup olive oil

Instructional Text Generation

Input: Goal Cue - Bag of concepts

Preheat oven to 350 degrees F (175 degrees C). Lightly oil a 9x9 inch square baking pan.

Heat 3 tablespoons olive oil in a large skillet over medium heat. Saute onion, green onions and garlic, until soft and lightly browned. Stir in spinach and parsley, and continue to saute until spinach is limp, about 2 minutes. Remove from heat and set aside to cool.

In a medium bowl, mix together eggs, ricotta, and feta. Stir in spinach mixture. Lay 1 sheet of phyllo dough in prepared baking pan, and brush lightly with olive oil. Lay another sheet of phyllo dough on top, brush with olive oil, and repeat process with two more sheets of phyllo. The sheets will overlap the pan. Spread spinach and cheese mixture into pan and fold overhanging dough over filling. Brush with oil, then layer remaining 4 sheets of phyllo dough, brushing each with oil. Tuck overhanging dough into pan to seal filling.

Bake in preheated oven for 30 to 40 minutes, until golden brown. Cut into squares and serve while hot.

(Globally Coherent Text Generation with Neural Checklist Models. Kiddon et al, EMNLP 2016.)

NLG is everywhere

Storytelling Generation

Input: Script - Text - N/A

NLG is everywhere

Storytelling Generation

Input: Script - Text - N/A

Jim was obsessed with super heroes.
His sister told him if he tied a sheet on his back he could fly.
She convinced Jim to climb the ladder to the roof and jump off.
When he got up there he felt like he was superman.

NLG is everywhere

Storytelling Generation

Input: Script - Text - N/A

Jim was obsessed with super heroes.
His sister told him if he tied a sheet on his back he could fly.
She convinced Jim to climb the ladder to the roof and jump off.
When he got up there he felt like he was superman.

He ended up having a great time!

NLG is everywhere

Storytelling Generation

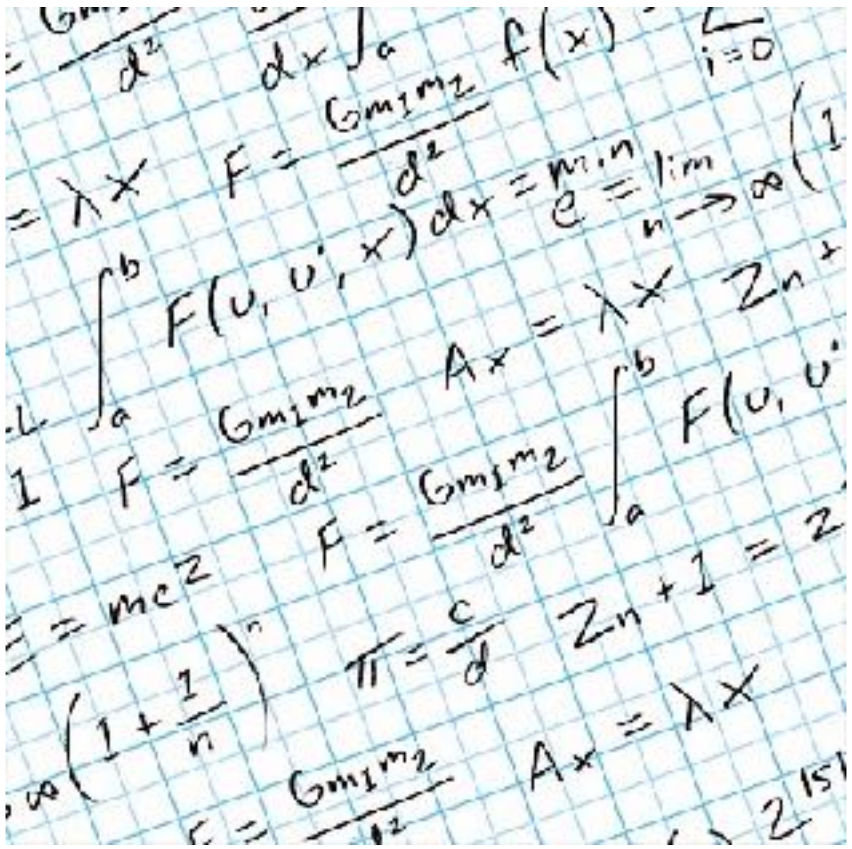
Input: Script - Text - N/A

Jim was obsessed with super heroes.
His sister told him if he tied a sheet on his back he could fly.
She convinced Jim to climb the ladder to the roof and jump off.
When he got up there he felt like he was superman.

He ended up having a great time!

Jim broke his arm and his sister was grounded for a year.

NLG is everywhere

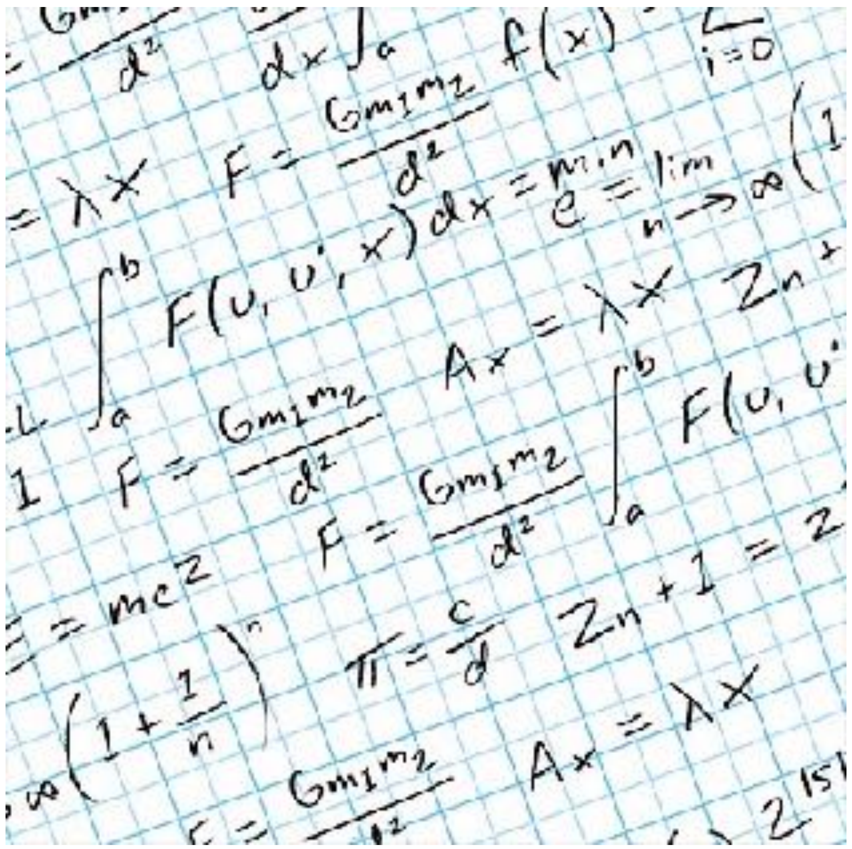


Storytelling Generation

Input: Equation + Theme

(Koncel-Kedziorski, Konstas, Zettlemoyer, Hajishirzi. A Theme-Rewriting Approach for Generating Algebra Word Problems. EMNLP 2016.)

NLG is everywhere



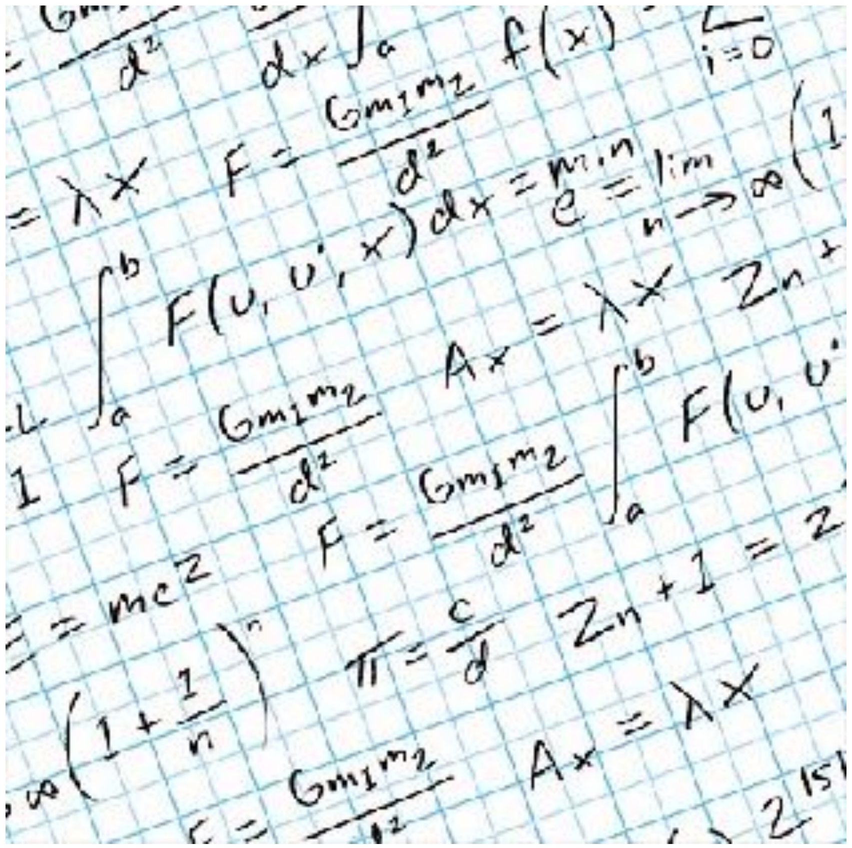
Storytelling Generation

Input: Equation + Theme

$504 + x = 639$ **+**

(Koncel-Kedziorski, Konstas, Zettlemoyer, Hajishirzi. A Theme-Rewriting Approach for Generating Algebra Word Problems. EMNLP 2016.)

NLG is everywhere



Storytelling Generation

Input: Equation + Theme

504 + x = 639 **+** STAR WARS

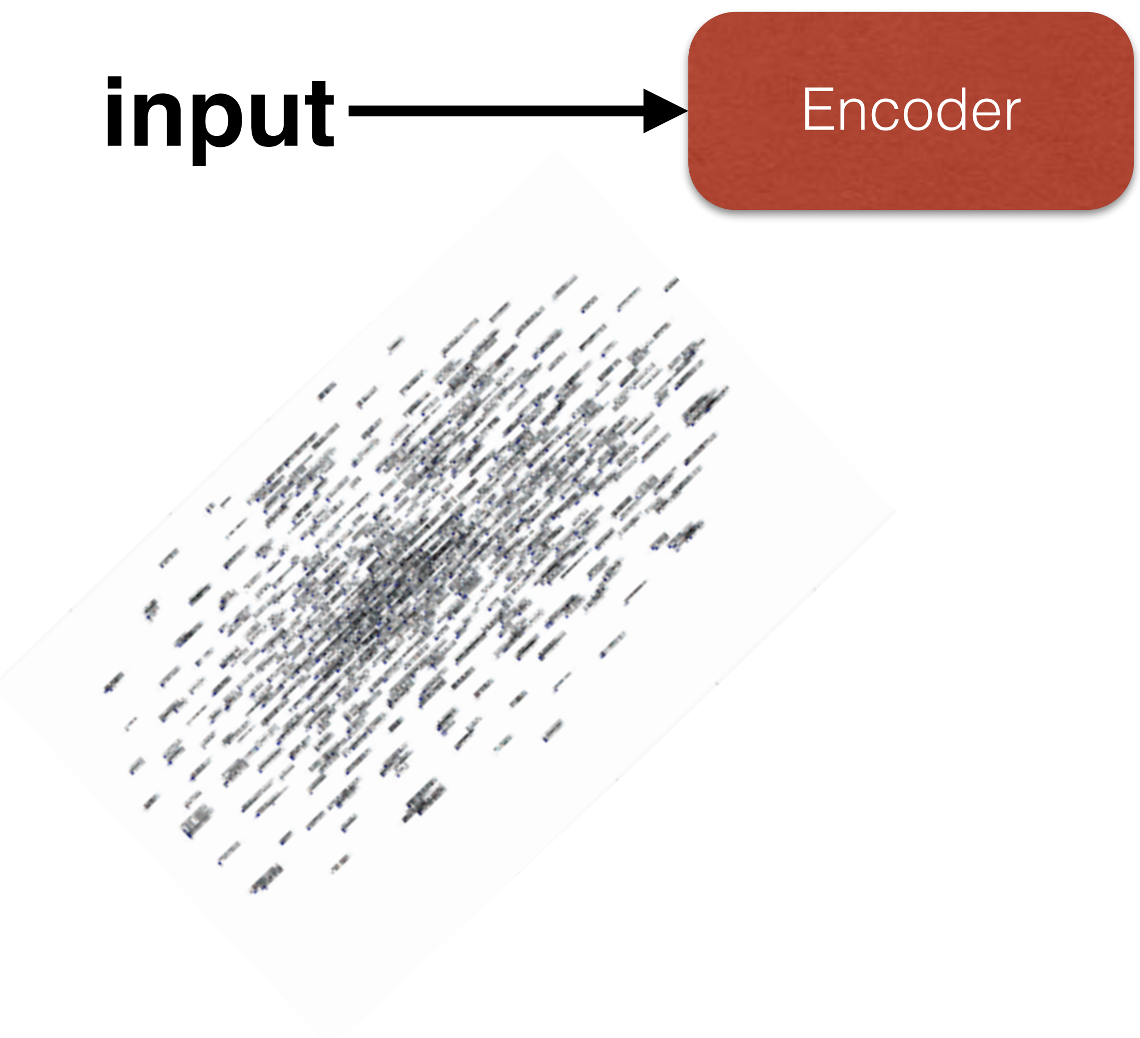
Luke Skywalker has 639 blasters. Leia has 504 blasters. How many more blasters does Luke Skywalker have than Leia?

(Koncel-Kedziorski, Konstas, Zettlemoyer, Hajishirzi. A Theme-Rewriting Approach for Generating Algebra Word Problems. EMNLP 2016.)

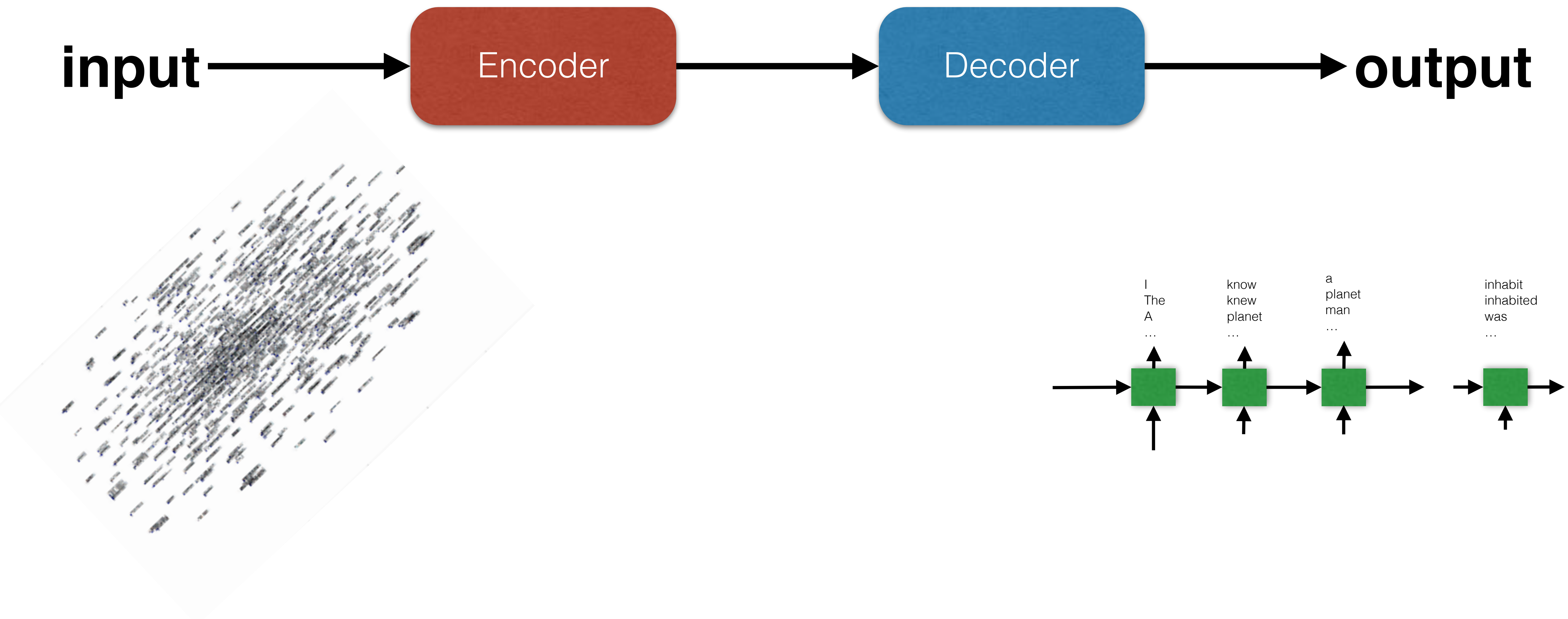
NNLG Framework

input

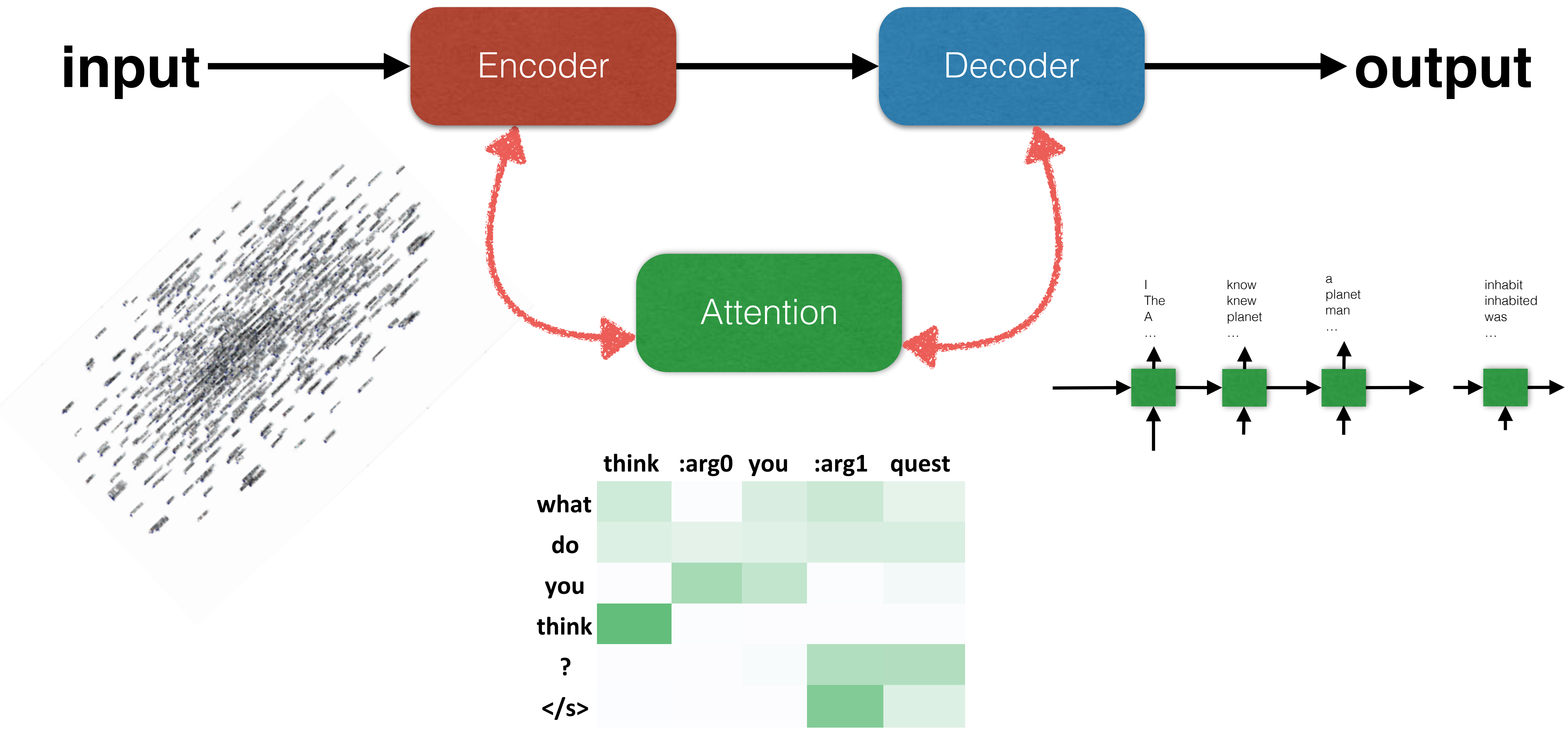
NNLG Framework



NNLG Framework



NNLG Framework



Encoding

Encoding

Bag of Words

CODE-NN

```
SELECT max(marks) FROM stud_records WHERE marks <
(SELECT max(marks) FROM stud_records);
```

(Summarizing Source Code using a Neural Attention Model. Iyer, Konstas, Cheung, Zettlemoyer. ACL 2016.)

Encoding

Bag of Words

CODE-NN

```
SELECT max(marks) FROM stud_records WHERE marks <
(SELECT max(marks) FROM stud_records);
```

anonymization

```
SELECT max(col0) FROM tab0 WHERE col0 <
(SELECT max(col1) FROM tab1);
```

(Summarizing Source Code using a Neural Attention Model. Iyer, Konstas, Cheung, Zettlemoyer. ACL 2016.)

Encoding

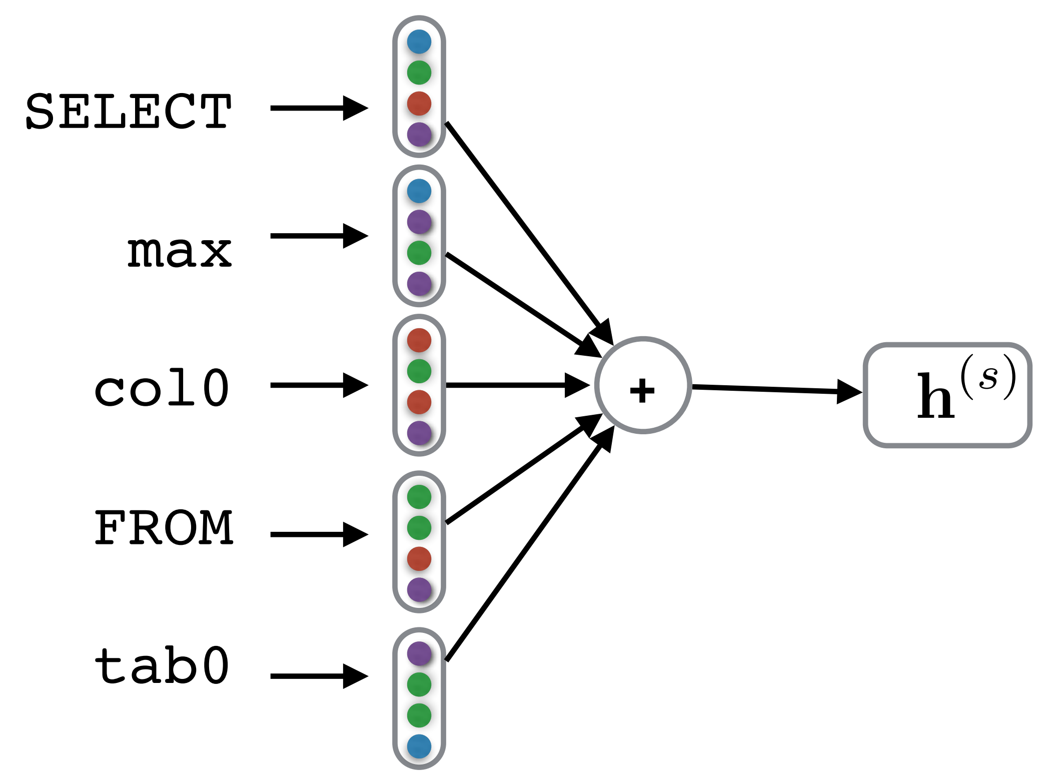
Bag of Words

CODE-NN

```
SELECT max(marks) FROM stud_records WHERE marks <
(SELECT max(marks) FROM stud_records);
```

anonymization

```
SELECT max(col0) FROM tab0 WHERE col0 <
(SELECT max(col1) FROM tab1);
```

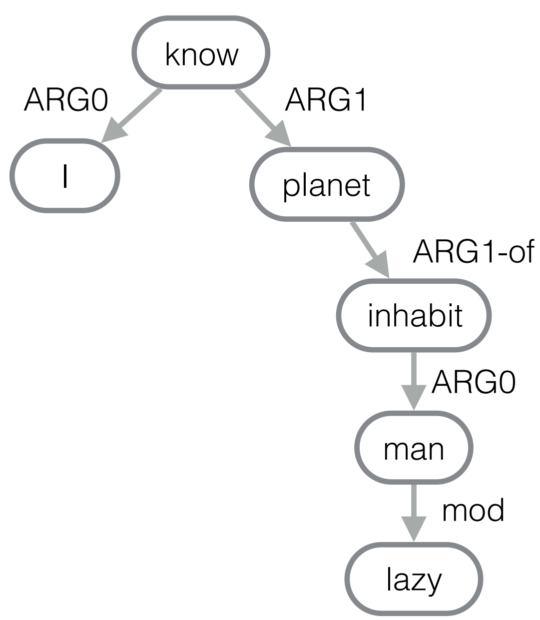


(Summarizing Source Code using a Neural Attention Model. Iyer, Konstas, Cheung, Zettlemoyer. ACL 2016.)

Encoding

Linearize —> RNN encoding

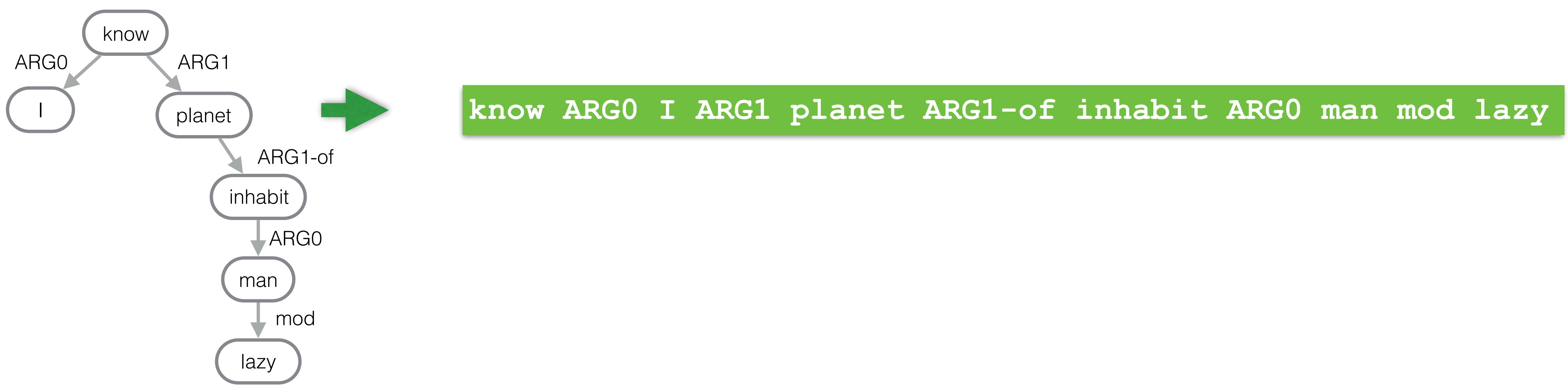
AMR Generation



Encoding

Linearize —> RNN encoding

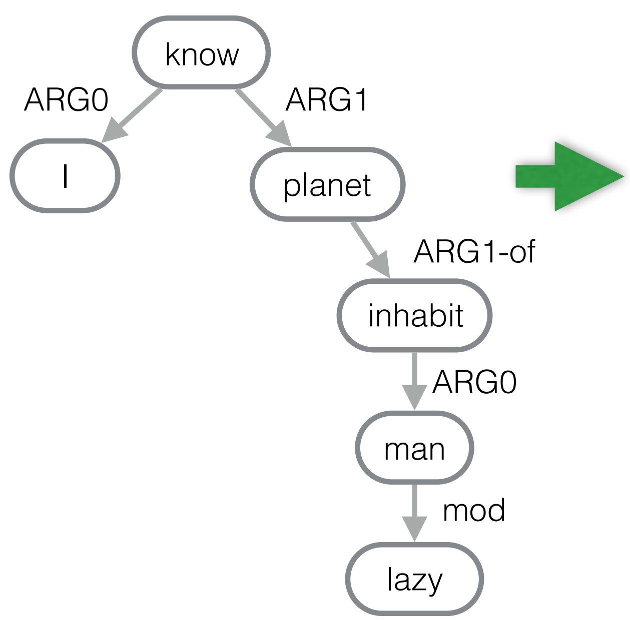
AMR Generation



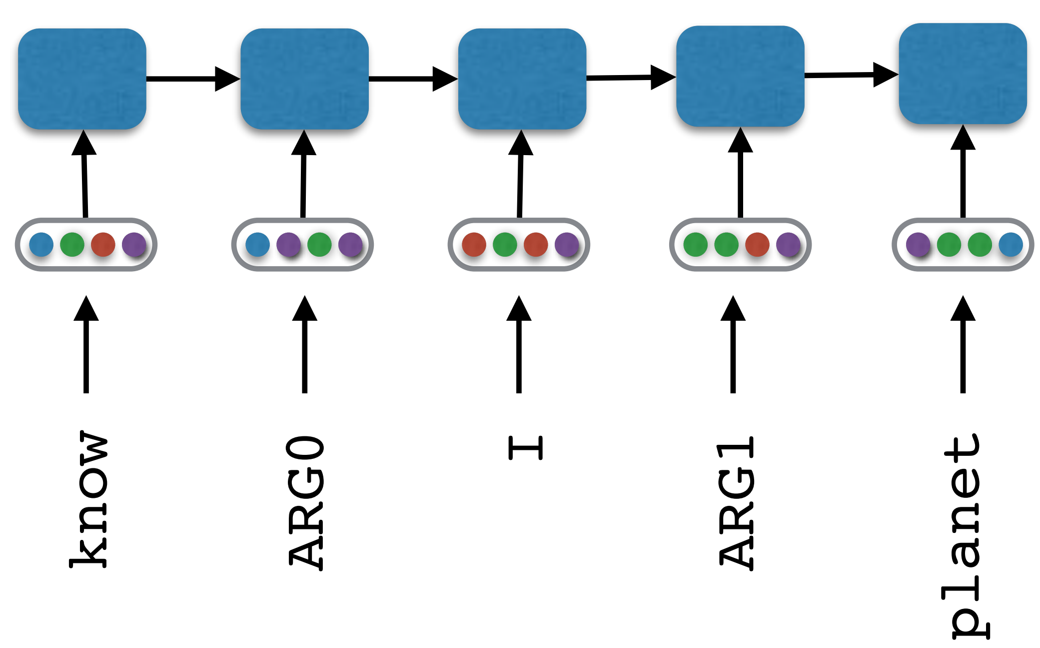
Encoding

Linearize —> RNN encoding

AMR Generation



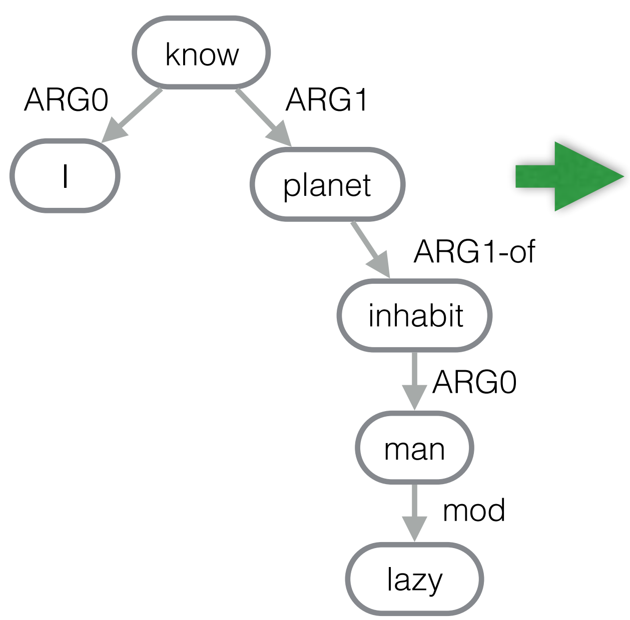
know ARG0 I ARG1 planet ARG1-of inhabit ARG0 man mod lazy



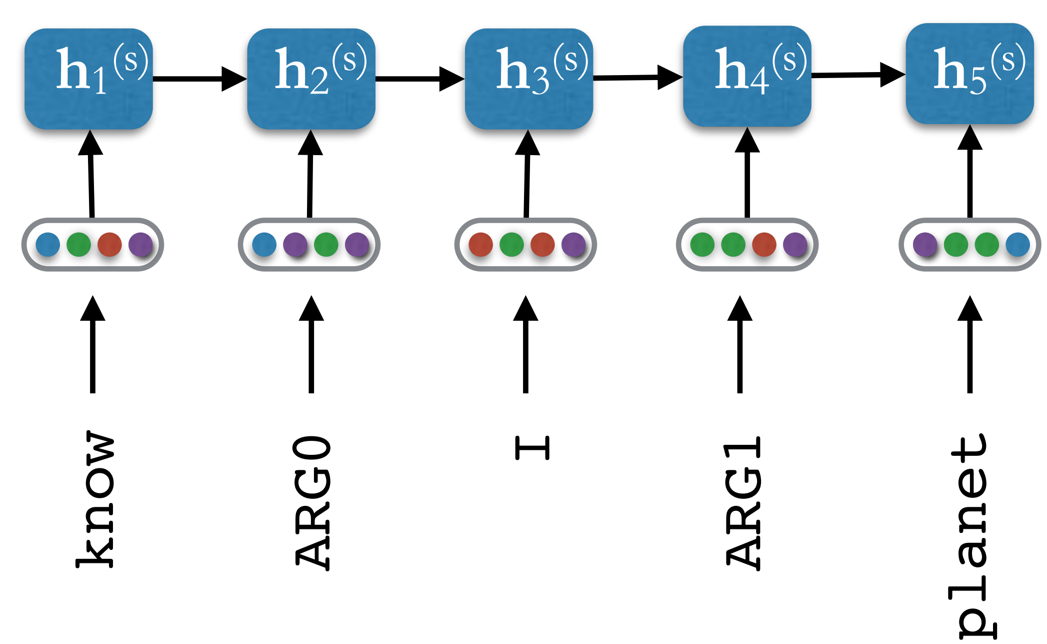
Encoding

Linearize —> RNN encoding

AMR Generation

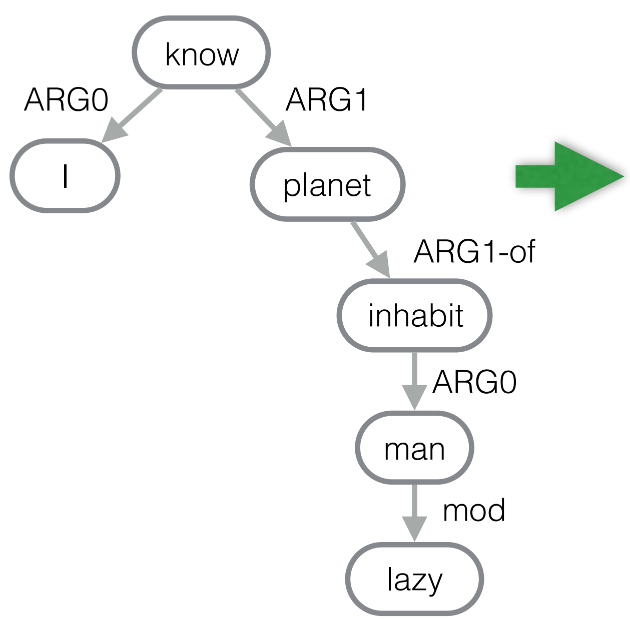


`know ARG0 I ARG1 planet ARG1-of inhabit ARG0 man mod lazy`

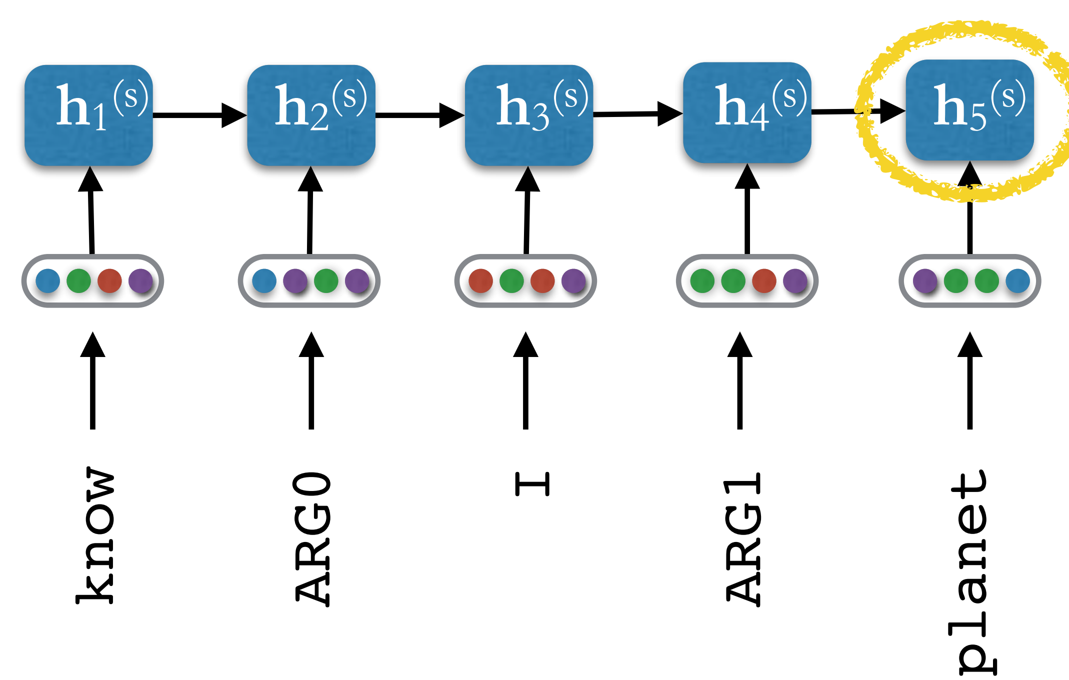


Encoding

Linearize —> RNN encoding **AMR Generation**



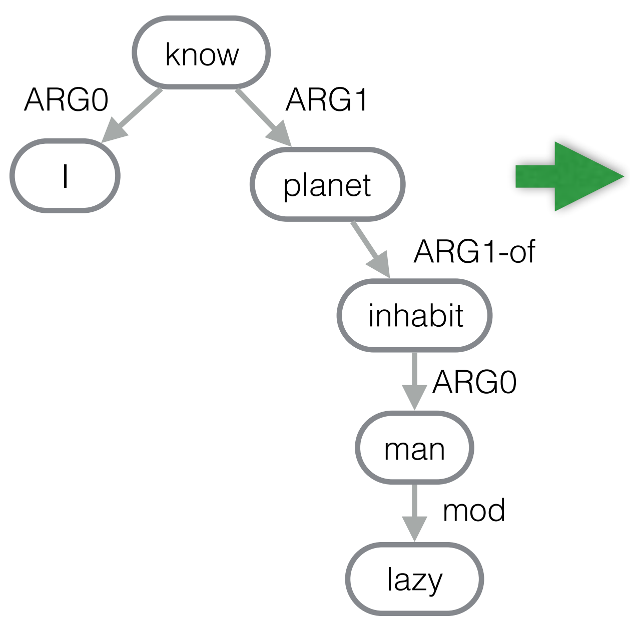
`know ARG0 I ARG1 planet ARG1-of inhabit ARG0 man mod lazy`



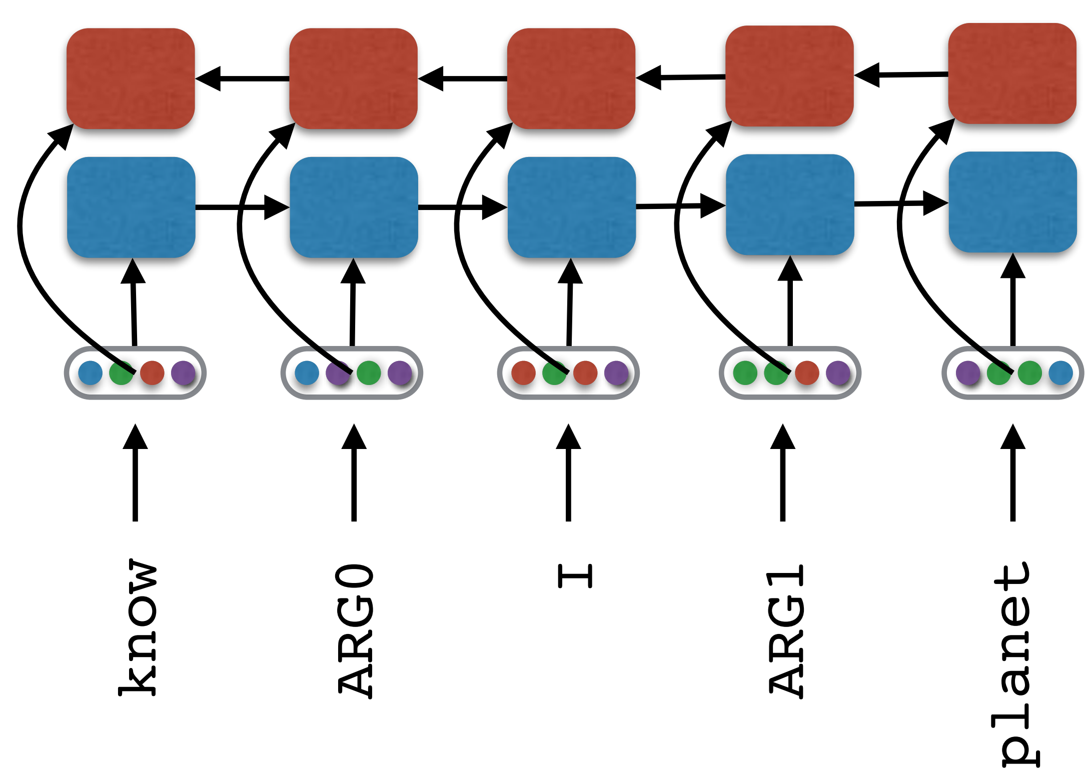
Encoding

Linearize —> RNN encoding

AMR Generation



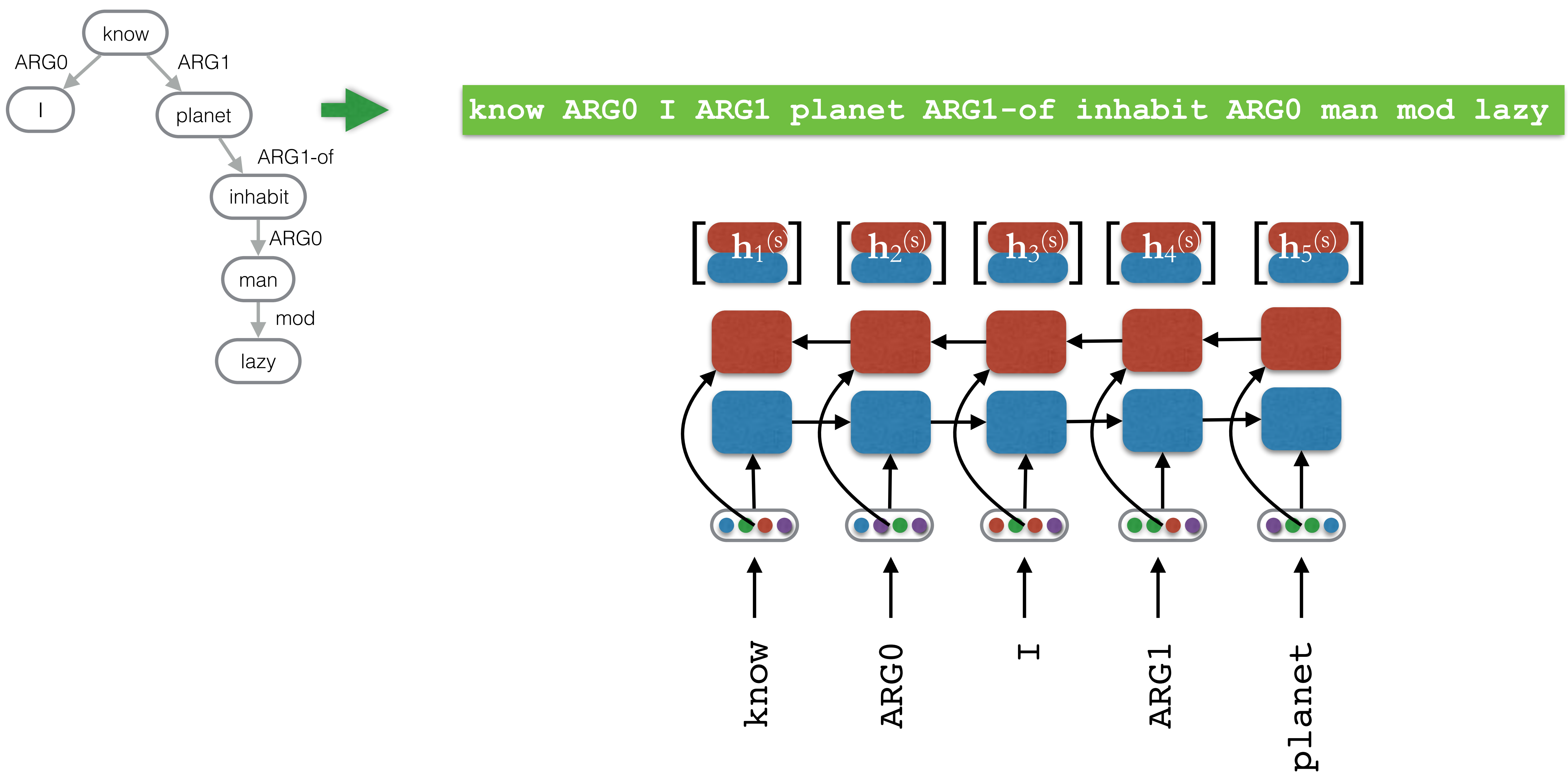
know ARG0 I ARG1 planet ARG1-of inhabit ARG0 man mod lazy



Encoding

Linearize —> RNN encoding

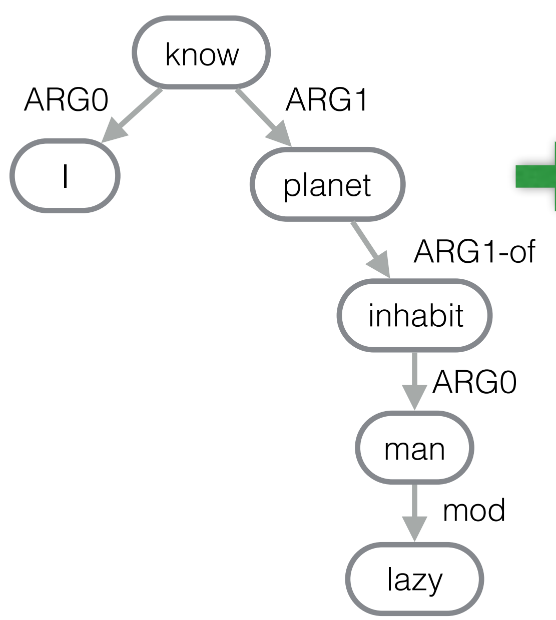
AMR Generation



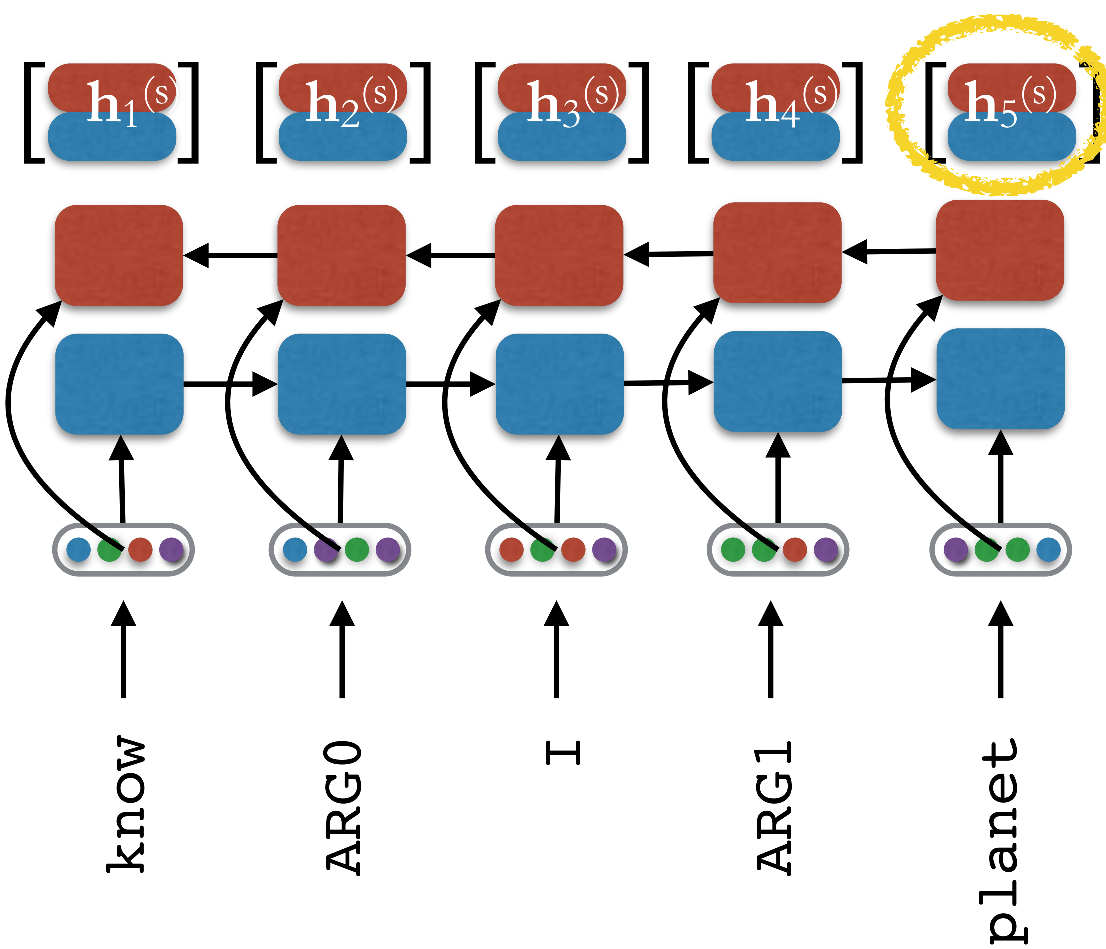
Encoding

Linearize —> RNN encoding

AMR Generation



`know ARG0 I ARG1 planet ARG1-of inhabit ARG0 man mod lazy`



Encoding

Hierarchical RNN encoding **Storytelling Generation**

Encoding

Hierarchical RNN encoding

Storytelling Generation

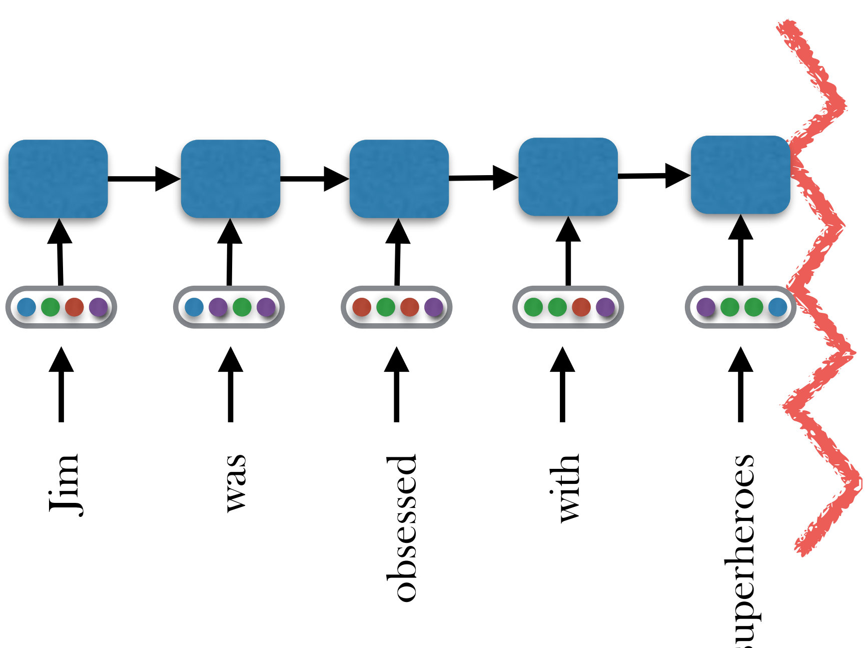
Jim was obsessed with superheroes.
His sister told him if he tied a sheet on his back he could fly.
She convinced Jim to climb the ladder to the roof and jump off.
When he got up there he felt like he was superman.

Encoding

Hierarchical RNN encoding

Storytelling Generation

Jim was obsessed with superheroes.
His sister told him if he tied a sheet on his back he could fly.
She convinced Jim to climb the ladder to the roof and jump off.
When he got up there he felt like he was superman.

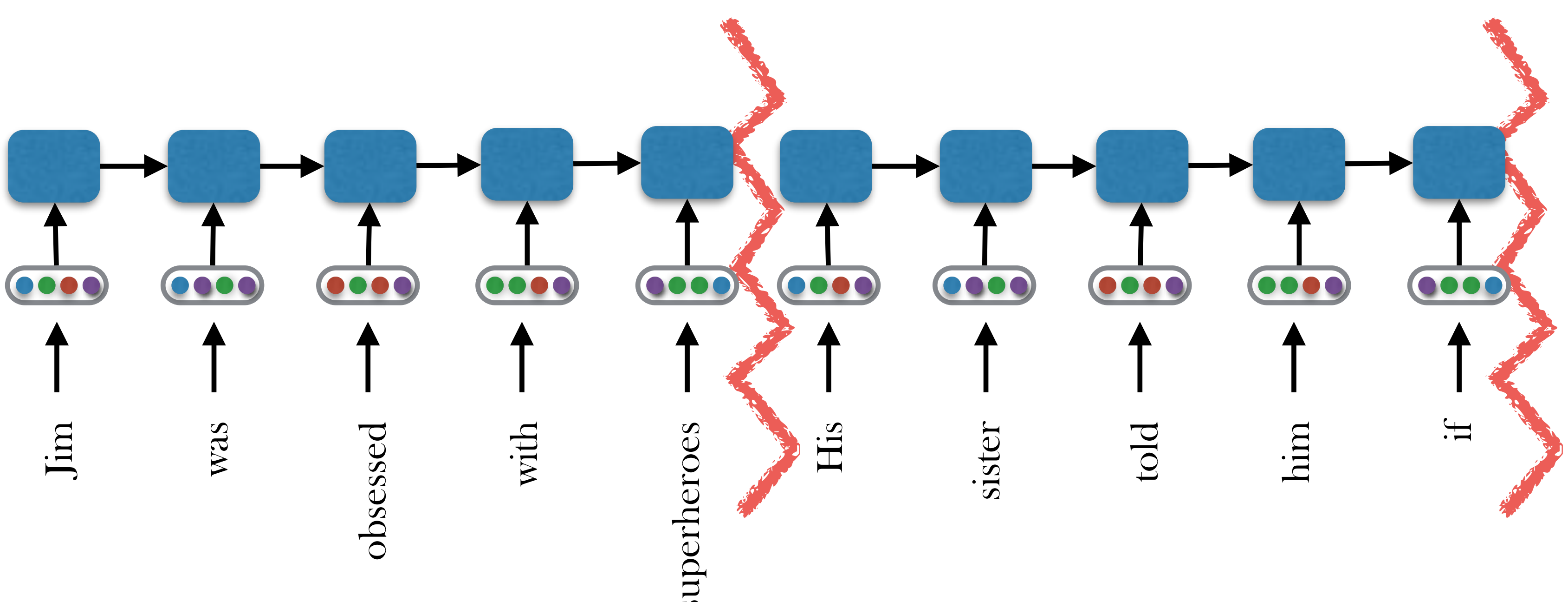


Encoding

Hierarchical RNN encoding

Storytelling Generation

Jim was obsessed with superheroes.
His sister told him if he tied a sheet on his back he could fly.
She convinced Jim to climb the ladder to the roof and jump off.
When he got up there he felt like he was superman.

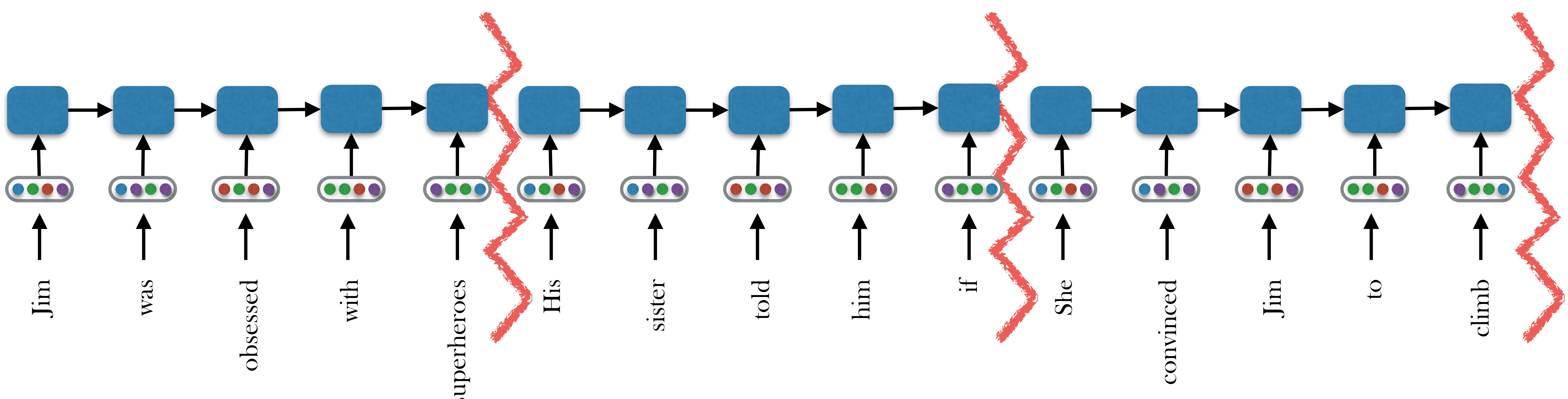


Encoding

Hierarchical RNN encoding

Storytelling Generation

Jim was obsessed with superheroes.
His sister told him if he tied a sheet on his back he could fly.
She convinced Jim to climb the ladder to the roof and jump off.
When he got up there he felt like he was superman.

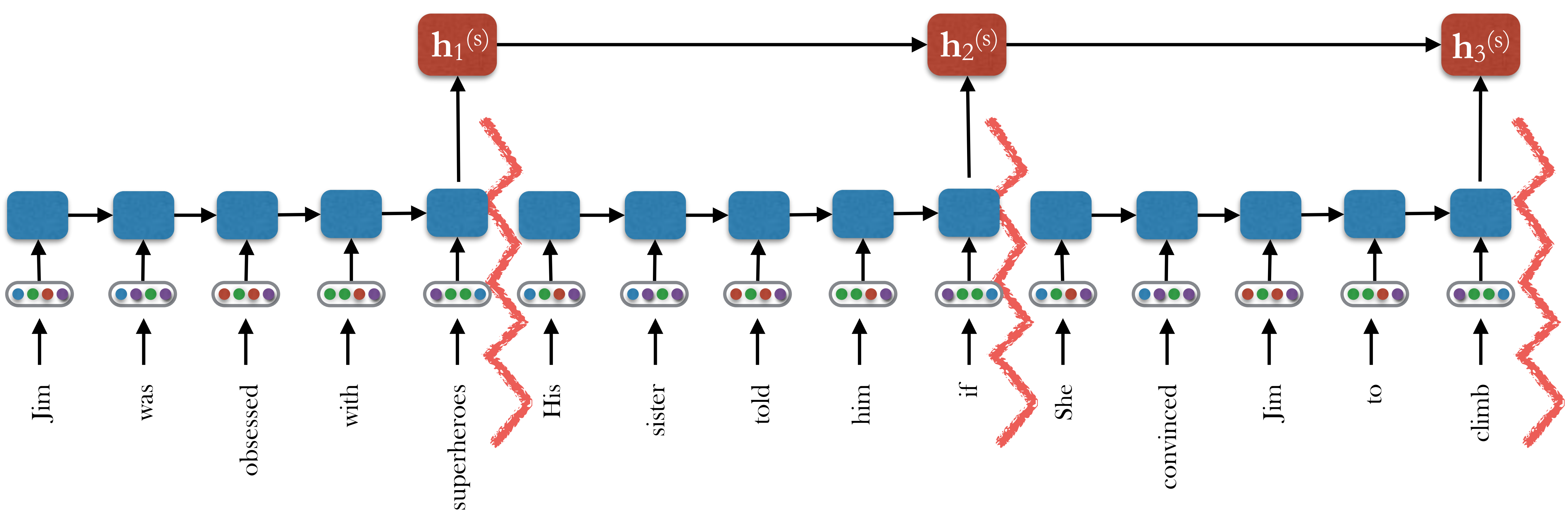


Encoding

Hierarchical RNN encoding

Storytelling Generation

Jim was obsessed with superheroes.
His sister told him if he tied a sheet on his back he could fly.
She convinced Jim to climb the ladder to the roof and jump off.
When he got up there he felt like he was superman.

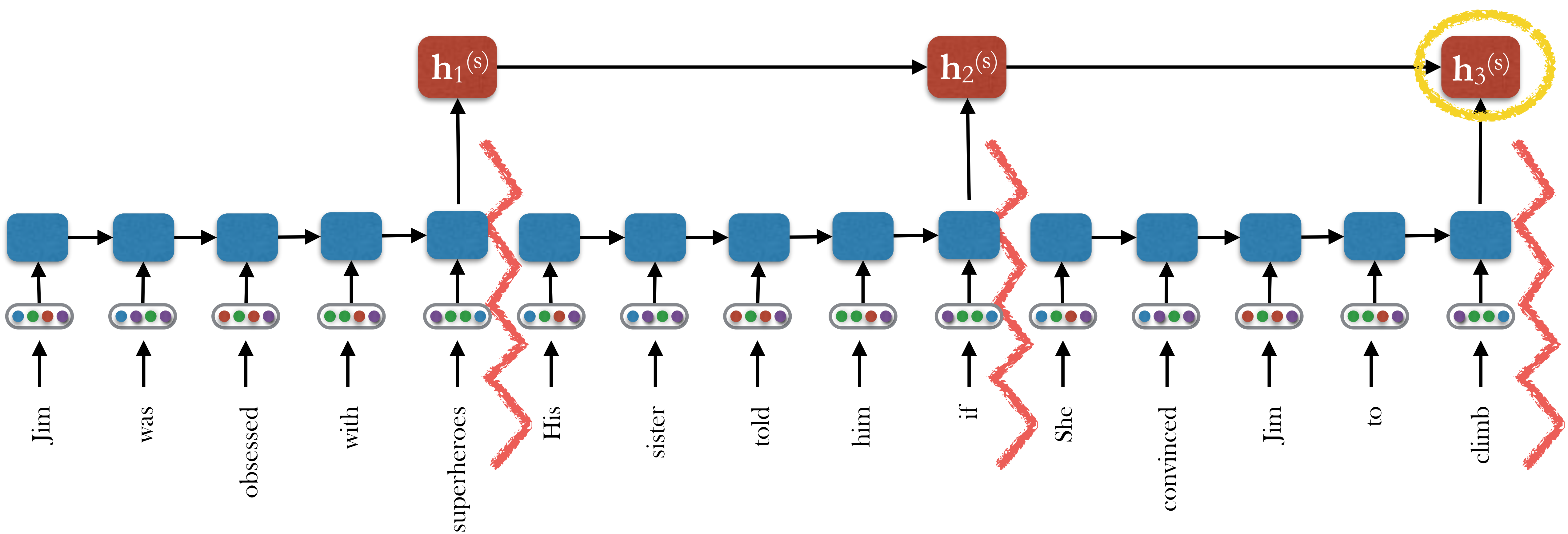


Encoding

Hierarchical RNN encoding

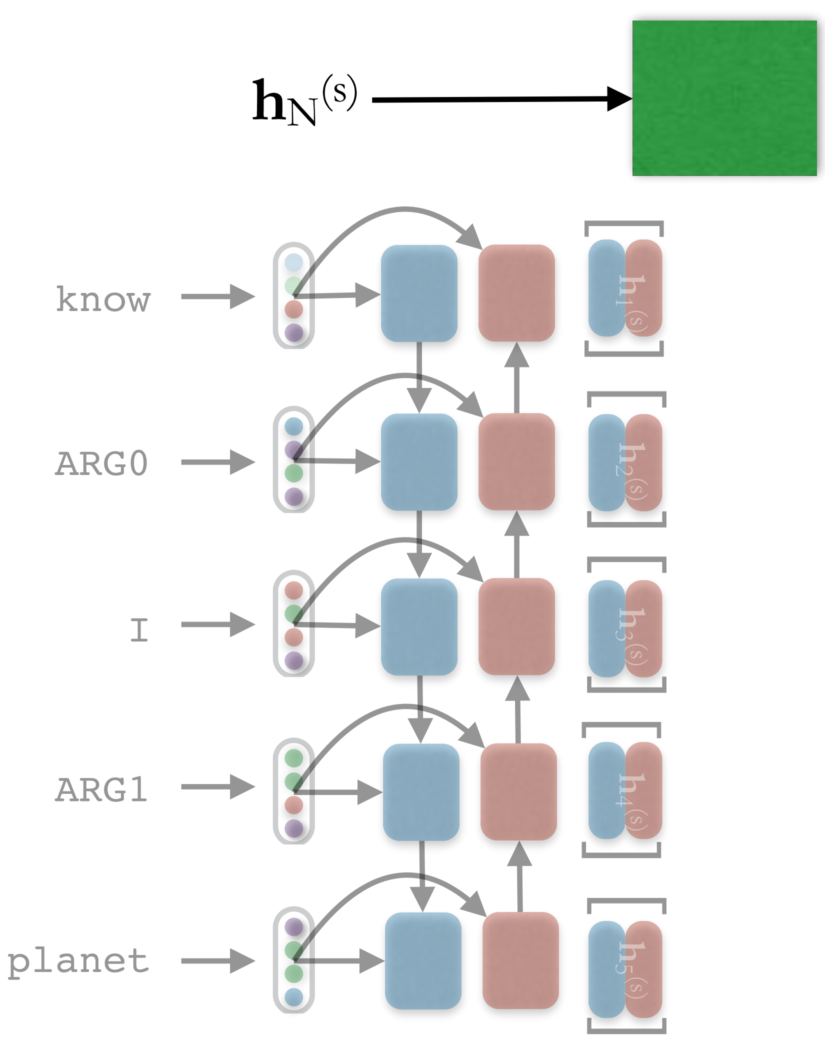
Storytelling Generation

Jim was obsessed with superheroes.
His sister told him if he tied a sheet on his back he could fly.
She convinced Jim to climb the ladder to the roof and jump off.
When he got up there he felt like he was superman.



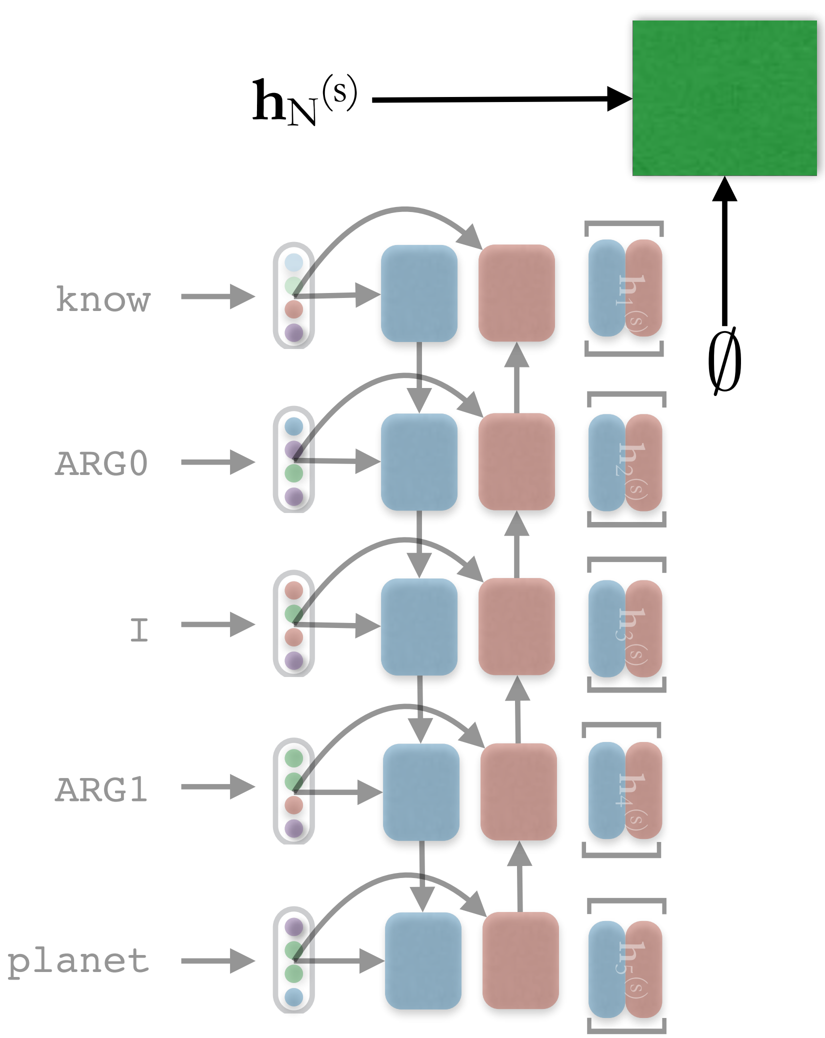
Decoding

Beam search (Left-to-Right)



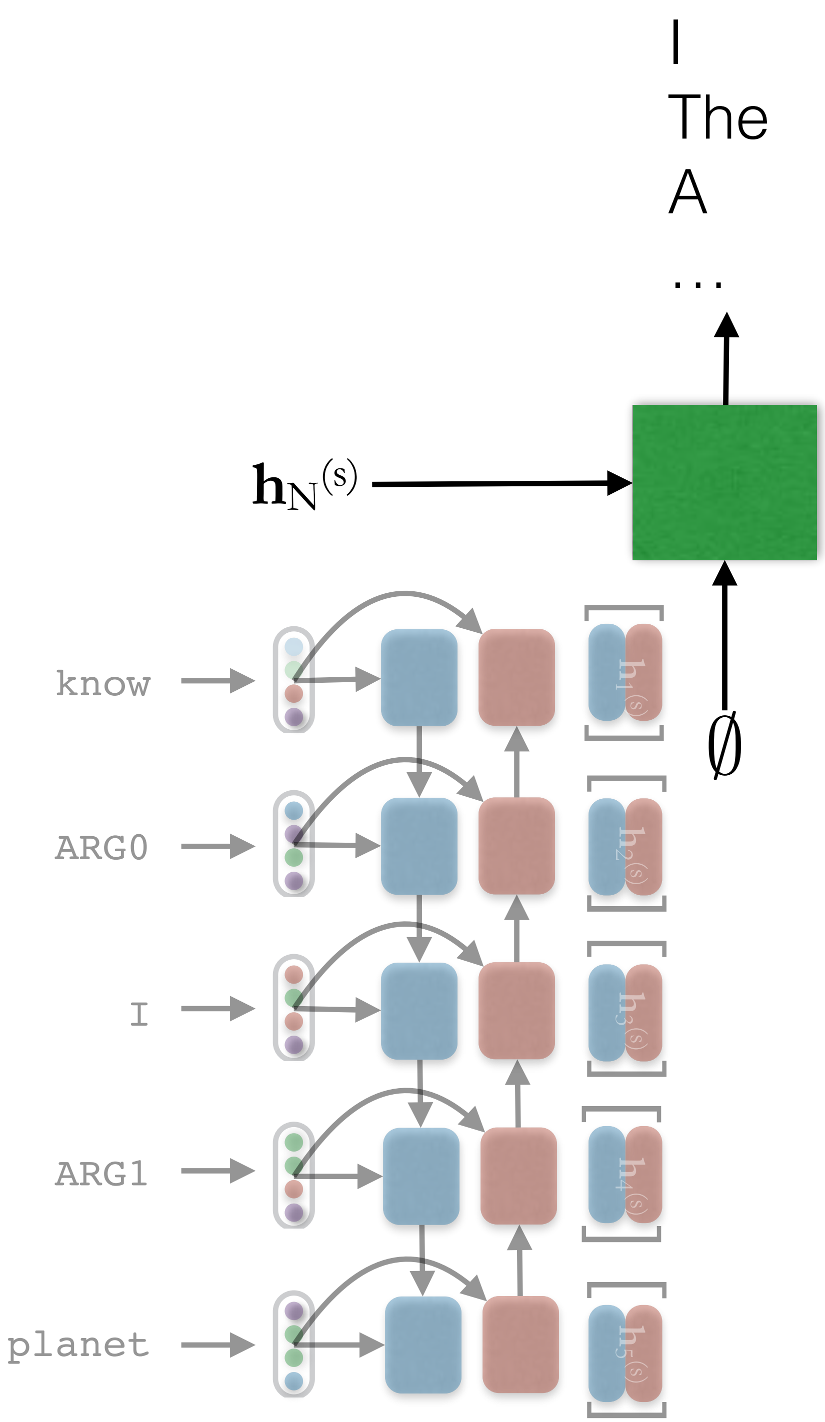
Decoding

Beam search (Left-to-Right)



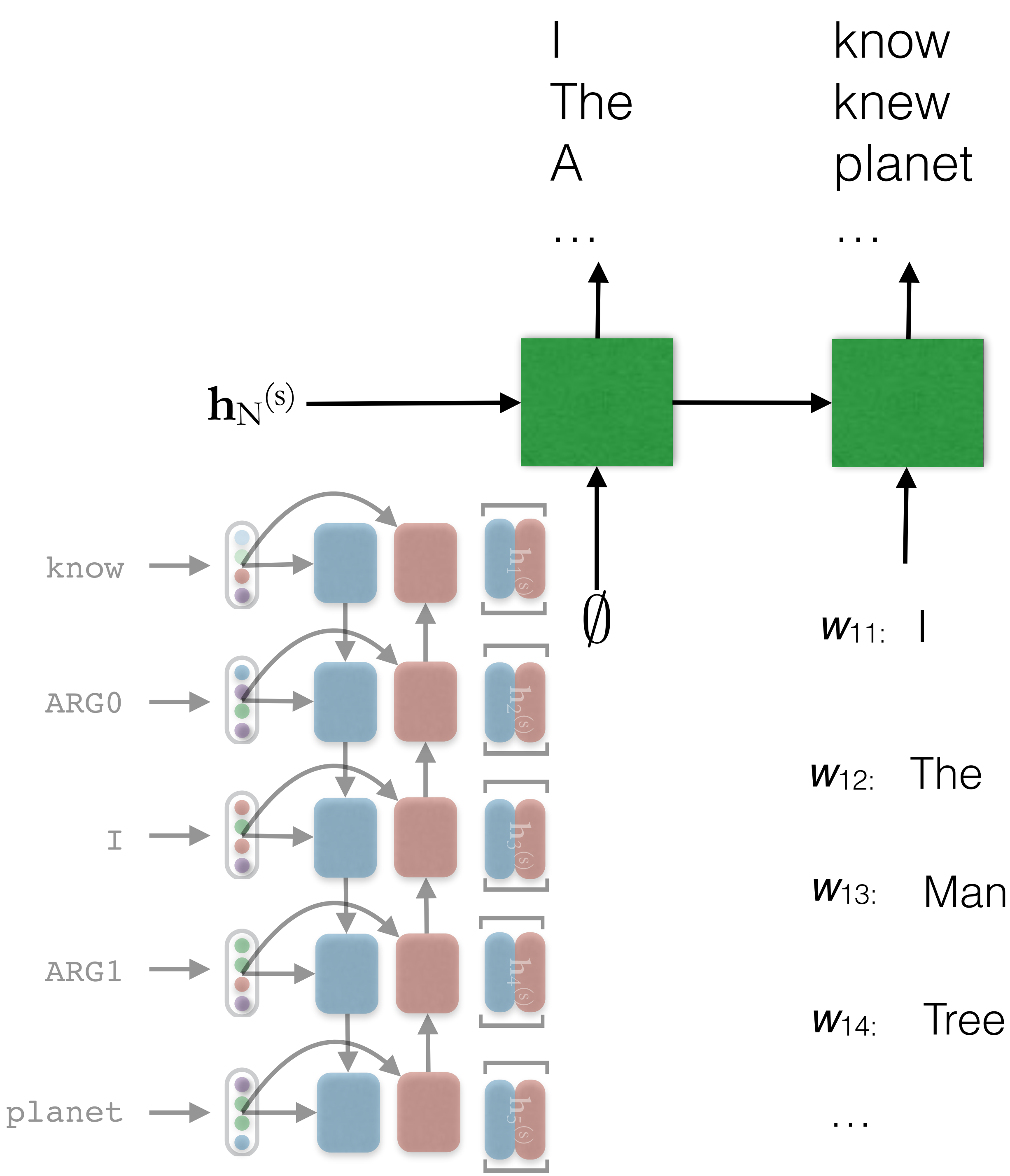
Decoding

Beam search (Left-to-Right)



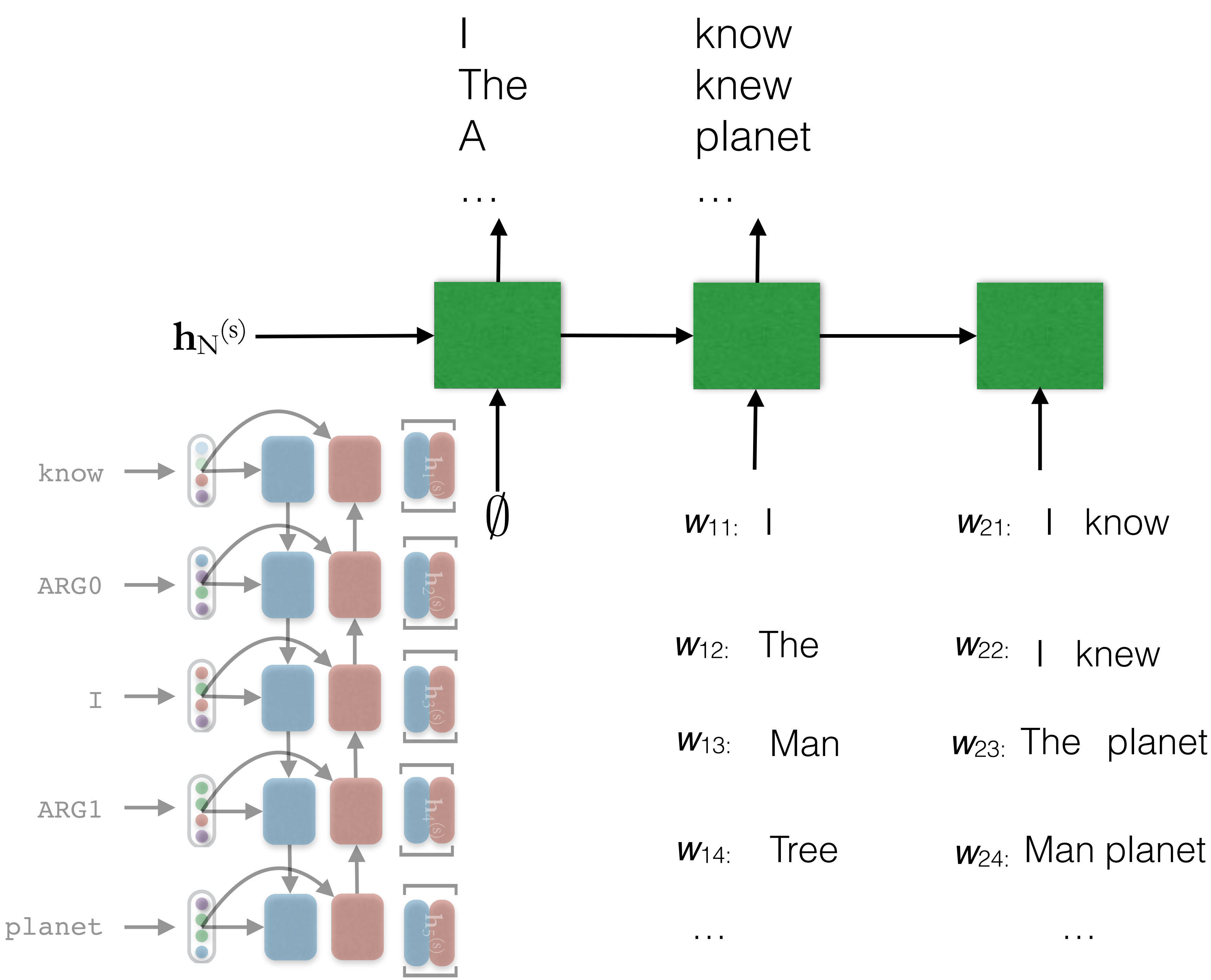
Decoding

Beam search (Left-to-Right)



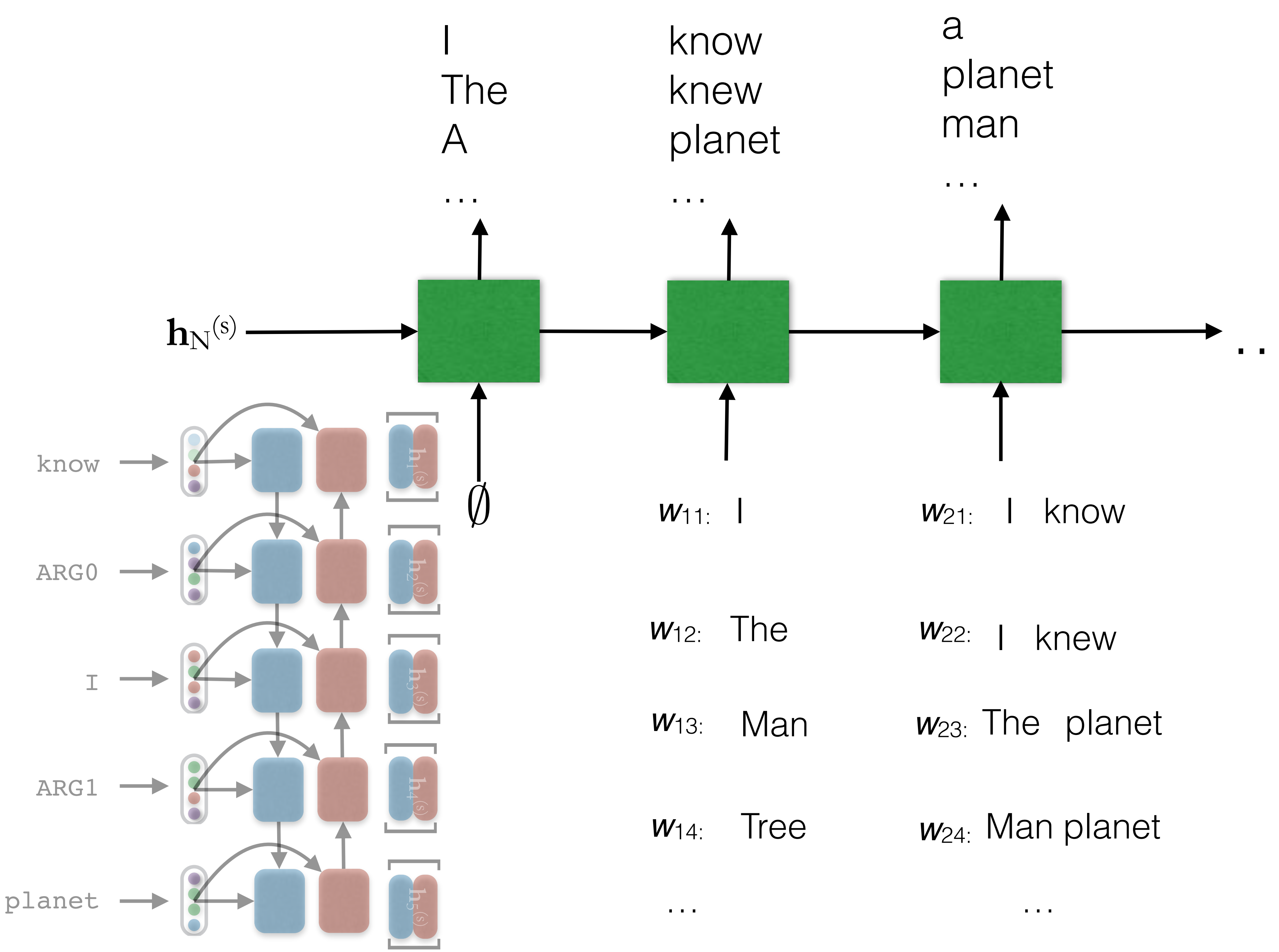
Decoding

Beam search (Left-to-Right)



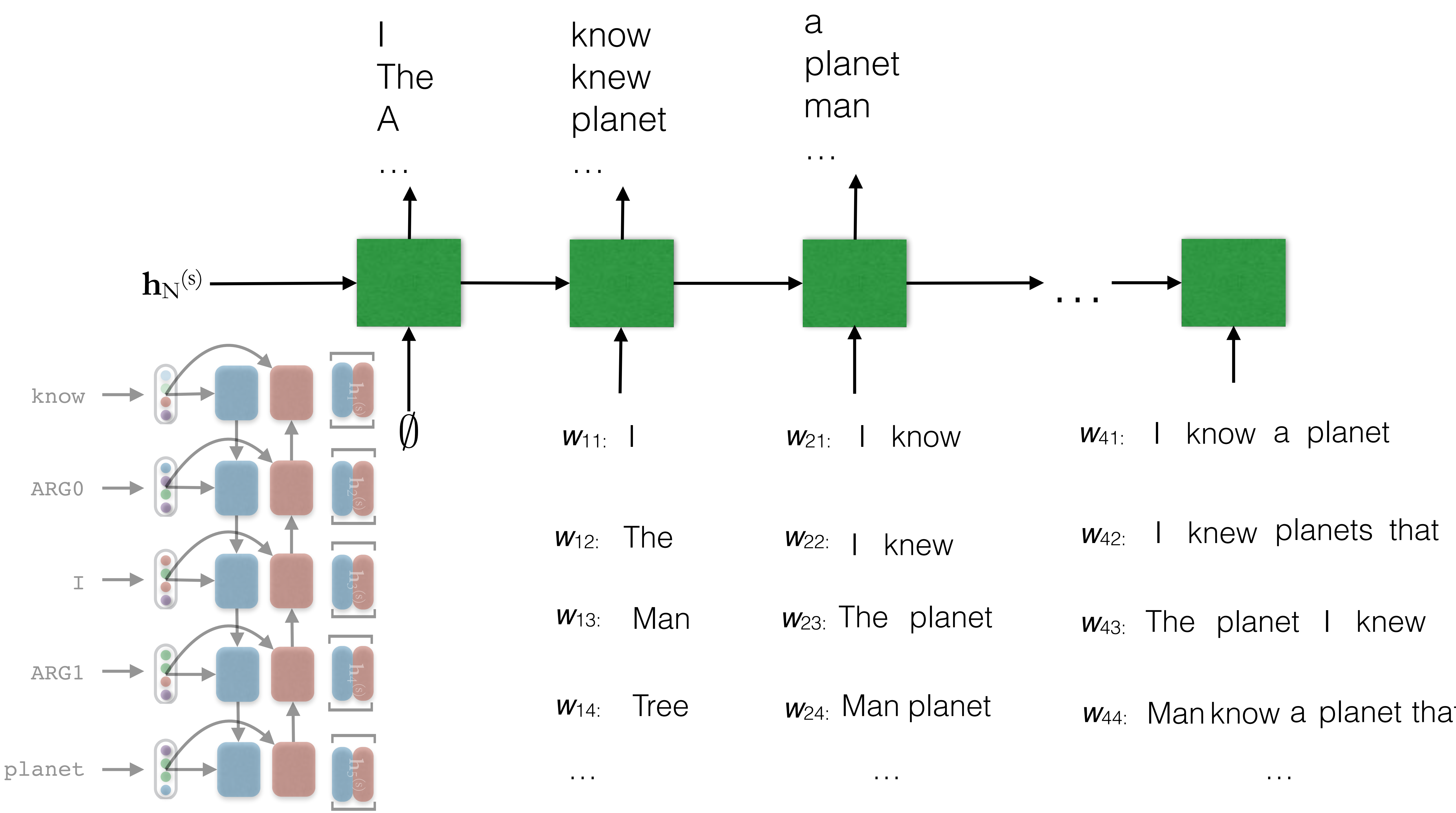
Decoding

Beam search (Left-to-Right)



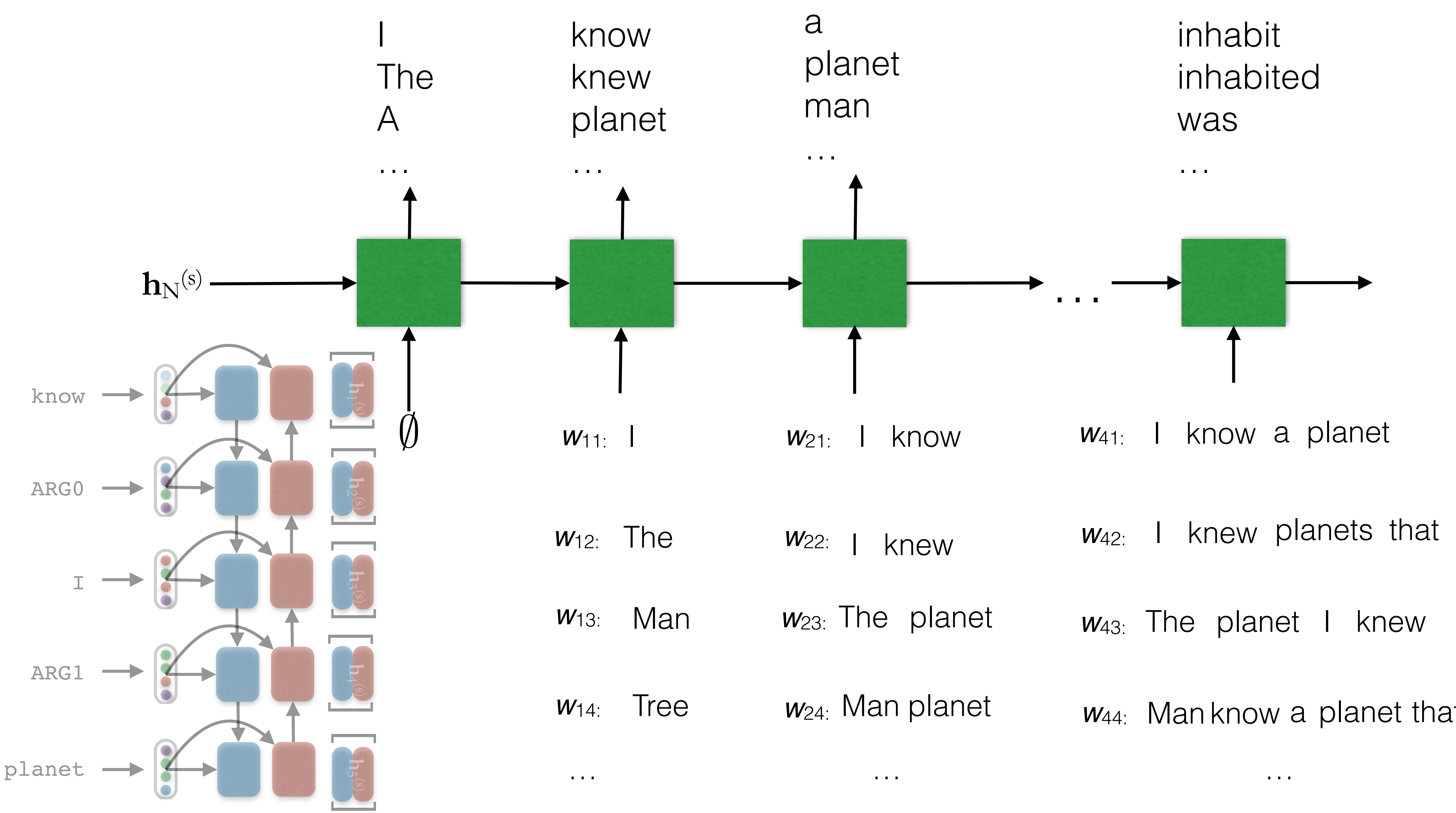
Decoding

Beam search (Left-to-Right)

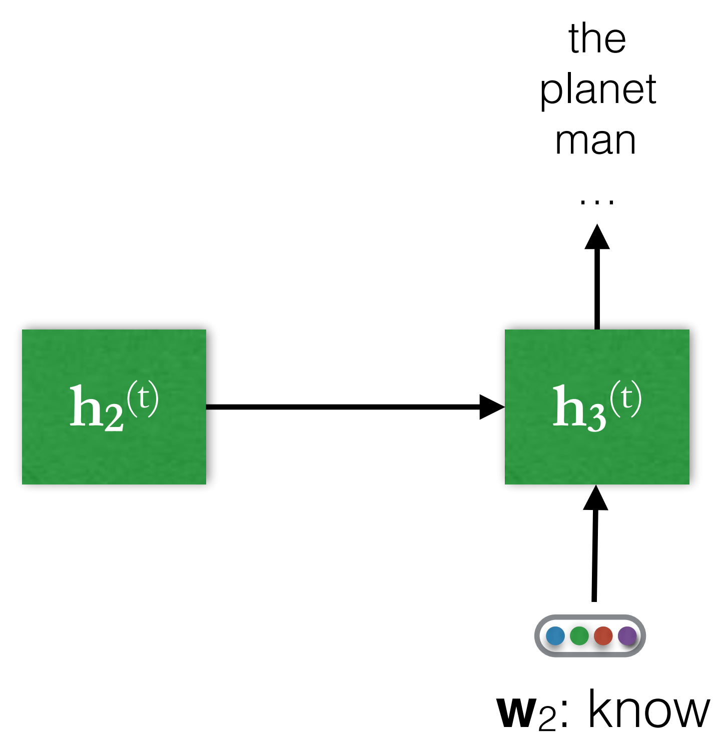


Decoding

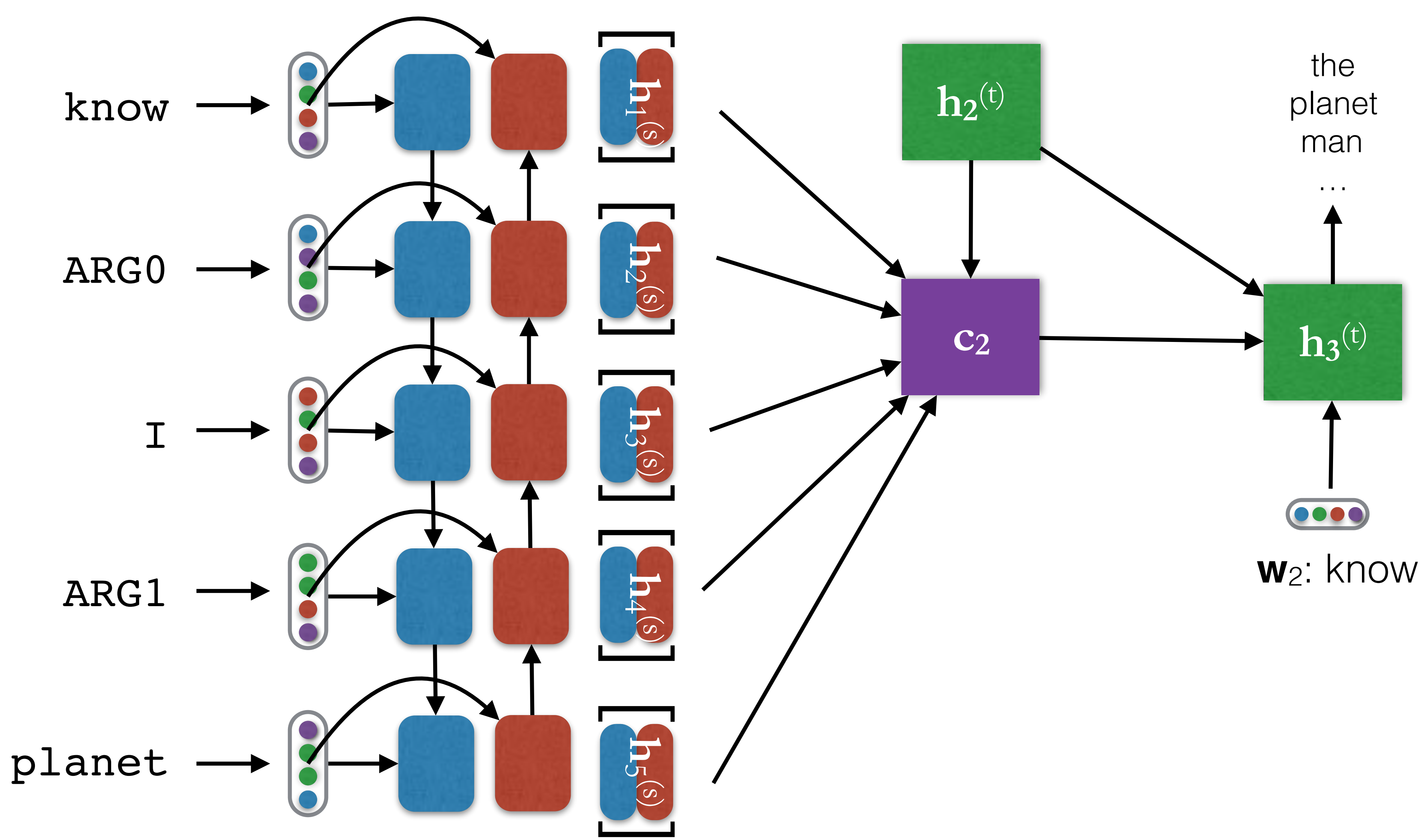
Beam search (Left-to-Right)



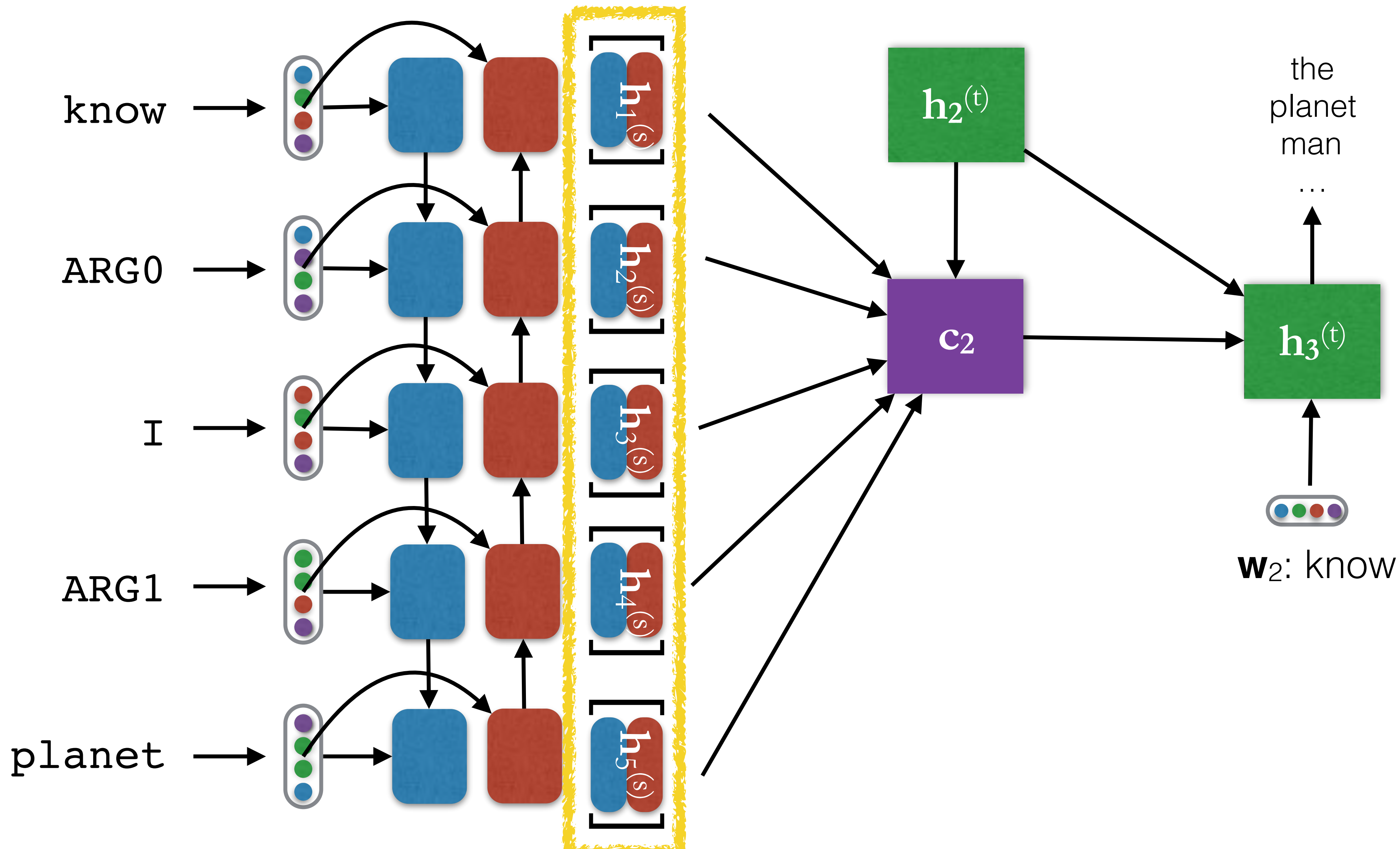
Attention



Attention

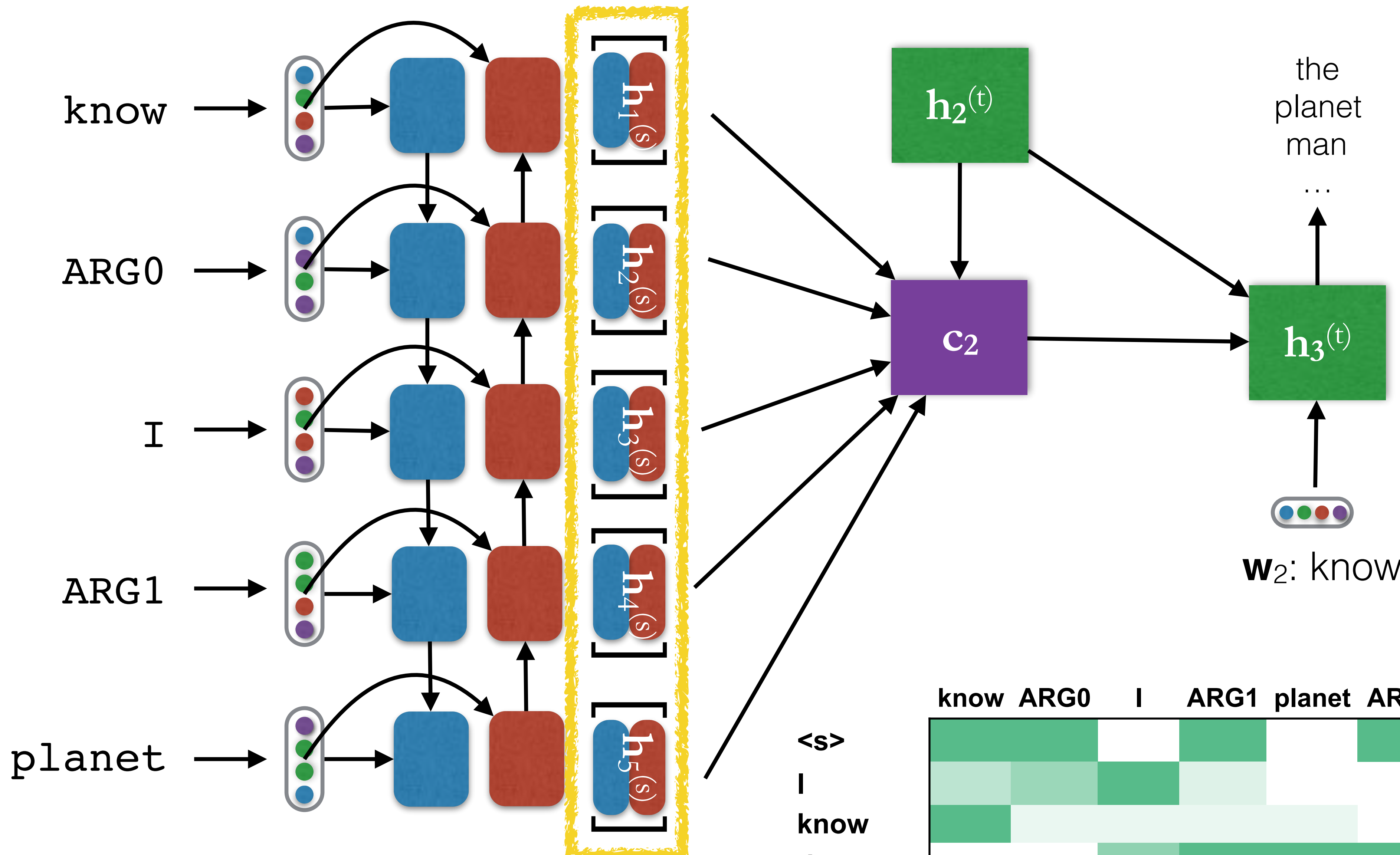

know
ARG0
I
ARG1
planet
h1(s)
h2(s)
h3(s)
h4(s)
h5(s)
h2(t)
c2
h3(t)
the
planet
man
...
w2: know

Attention

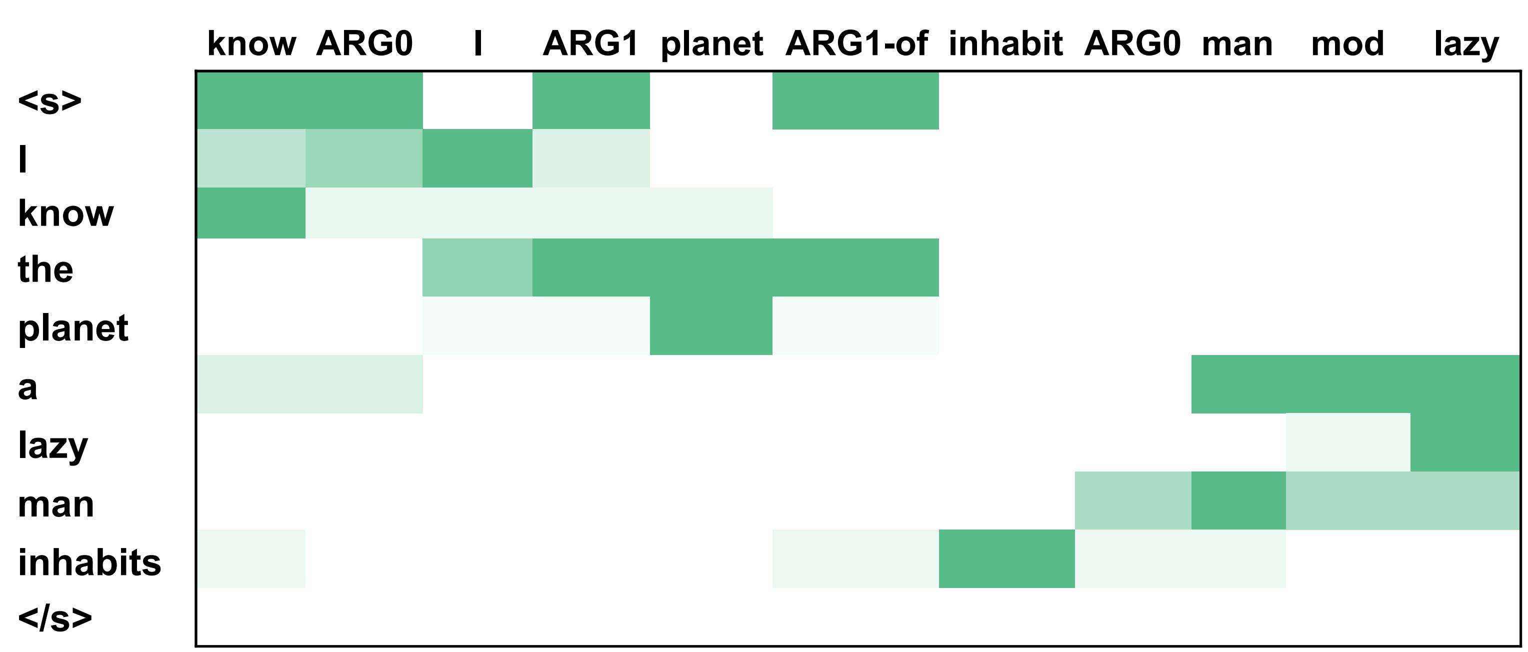


$$\mathbf{a}_i = softmax(\mathbf{f}_i(\mathbf{h}^{(s)}, \mathbf{h}_{i-1}^{(t)}))$$

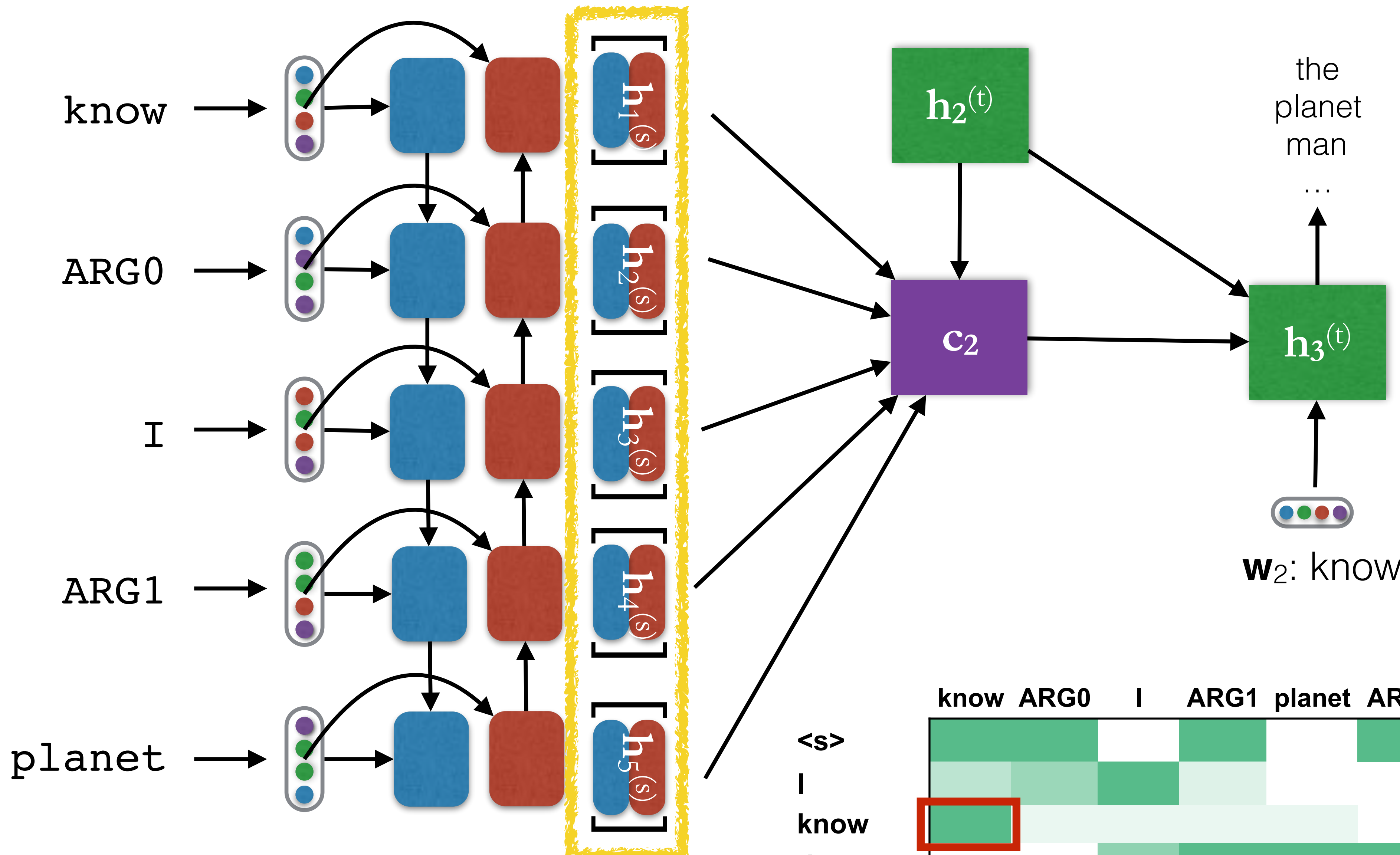
Attention



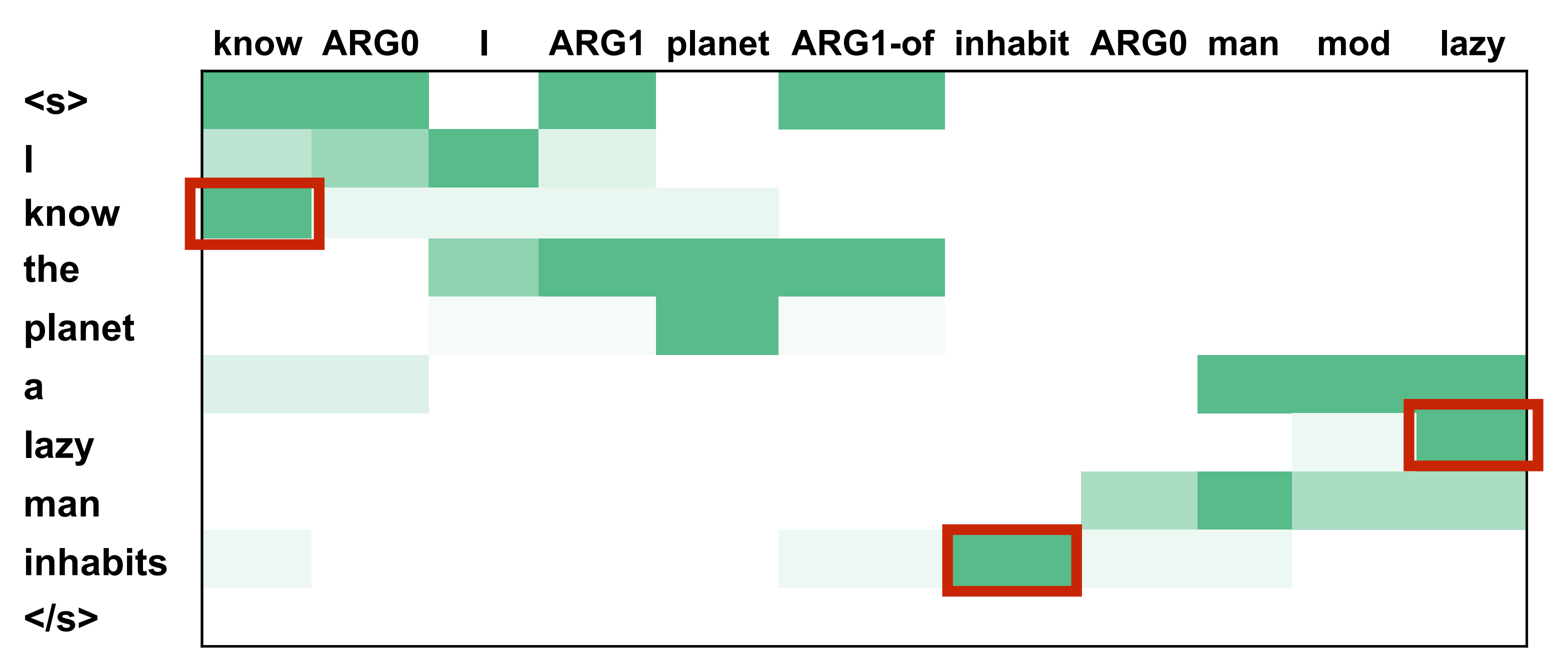
$$\mathbf{a}_i = softmax(\mathbf{f}_i(\mathbf{h}^{(s)}, \mathbf{h}_{i-1}^{(t)}))$$



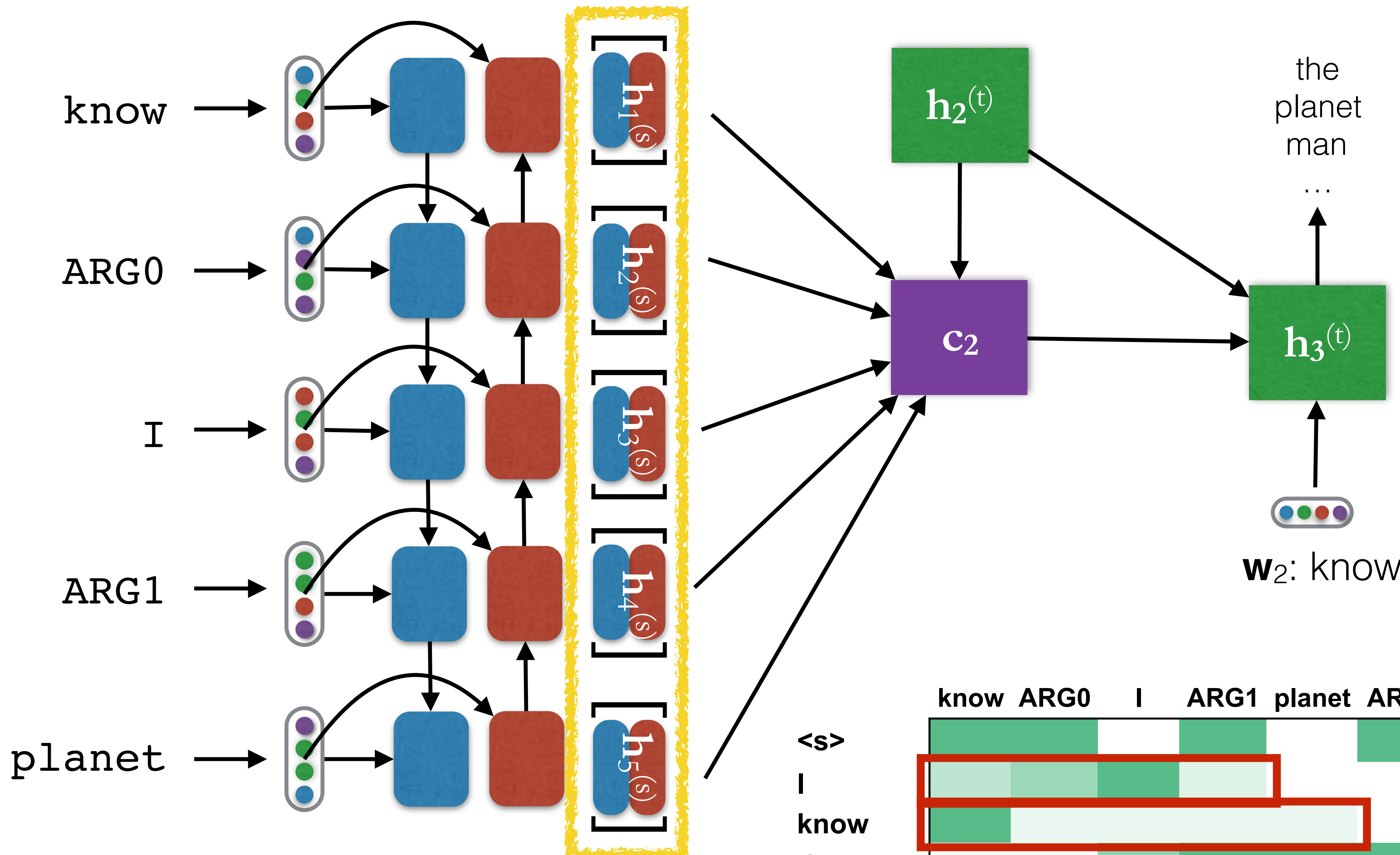
Attention

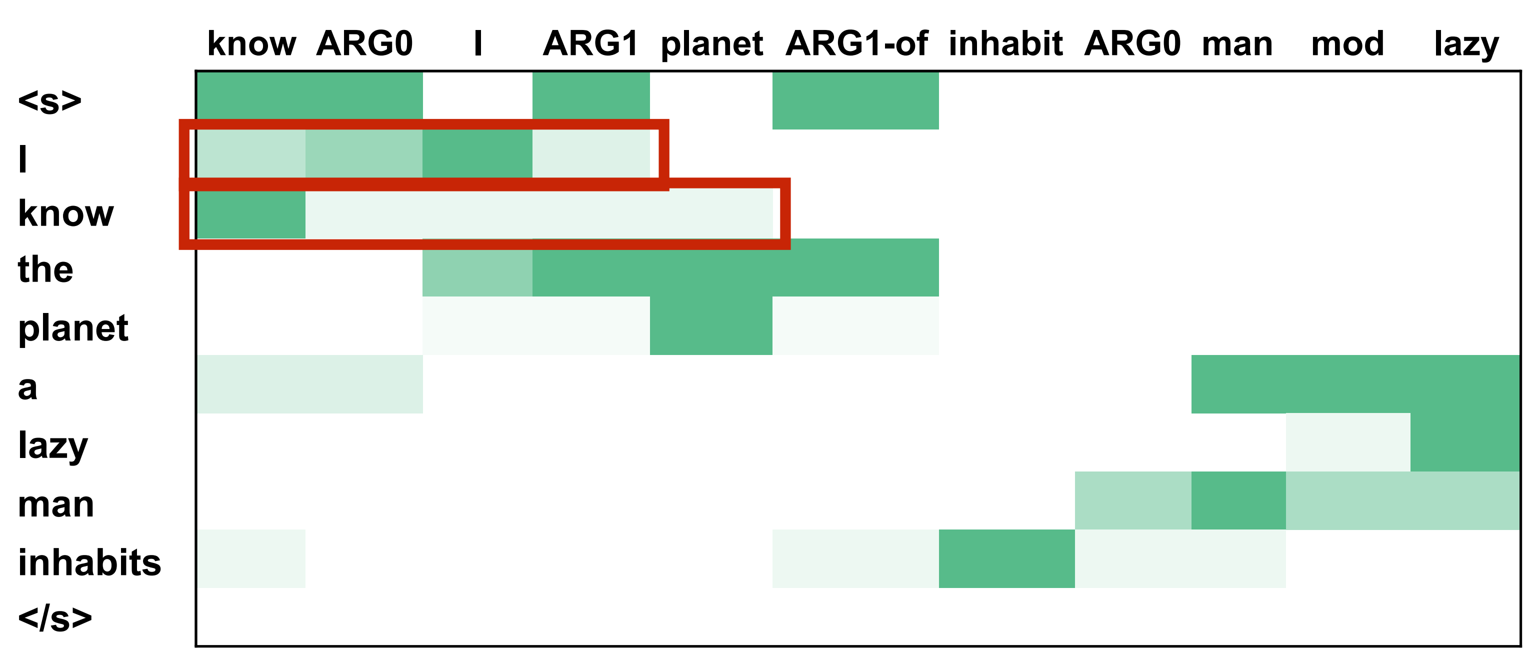


$$\mathbf{a}_i = softmax(\mathbf{f}_i(\mathbf{h}^{(s)}, \mathbf{h}^{(t)}_{i-1}))$$



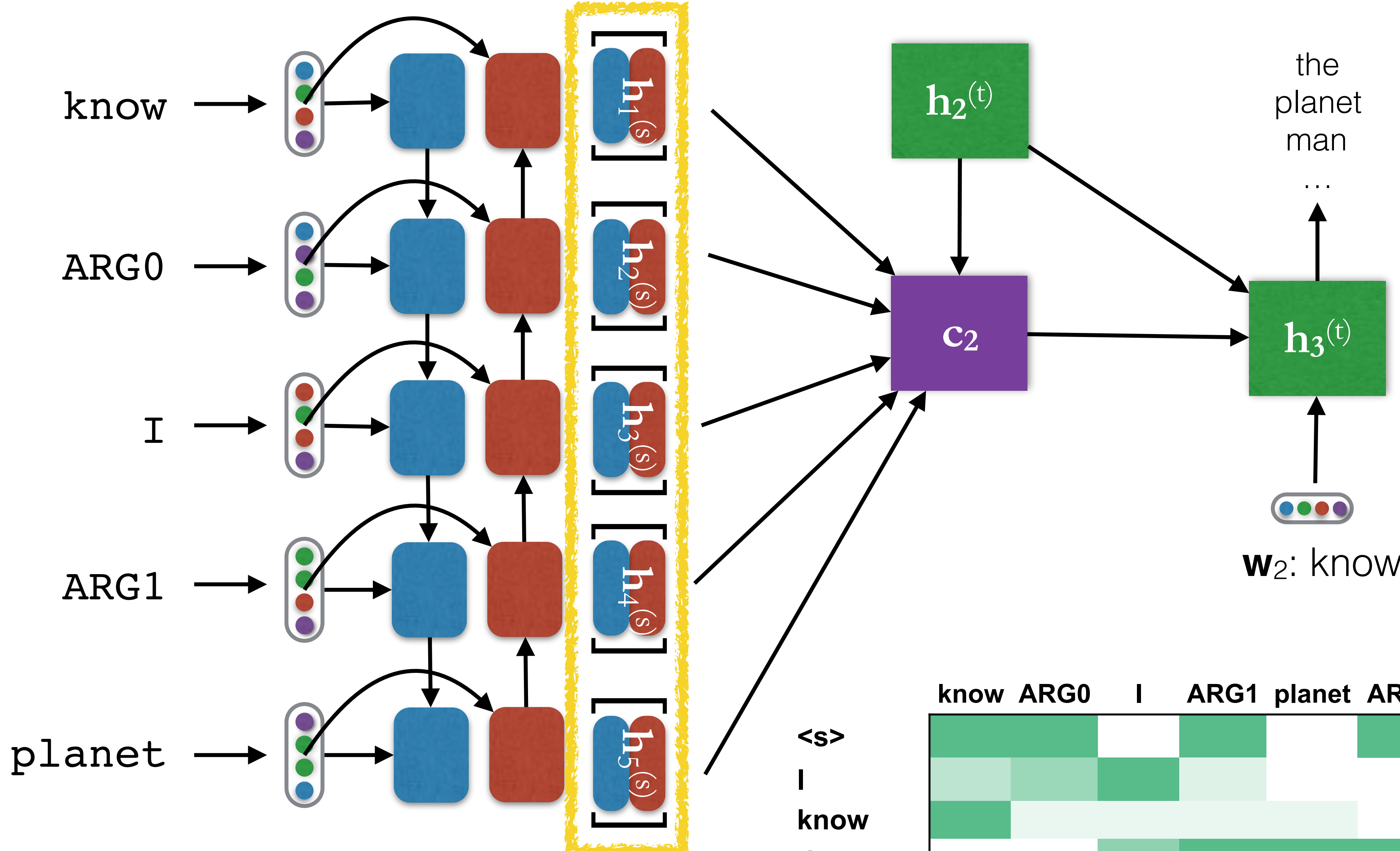
Attention



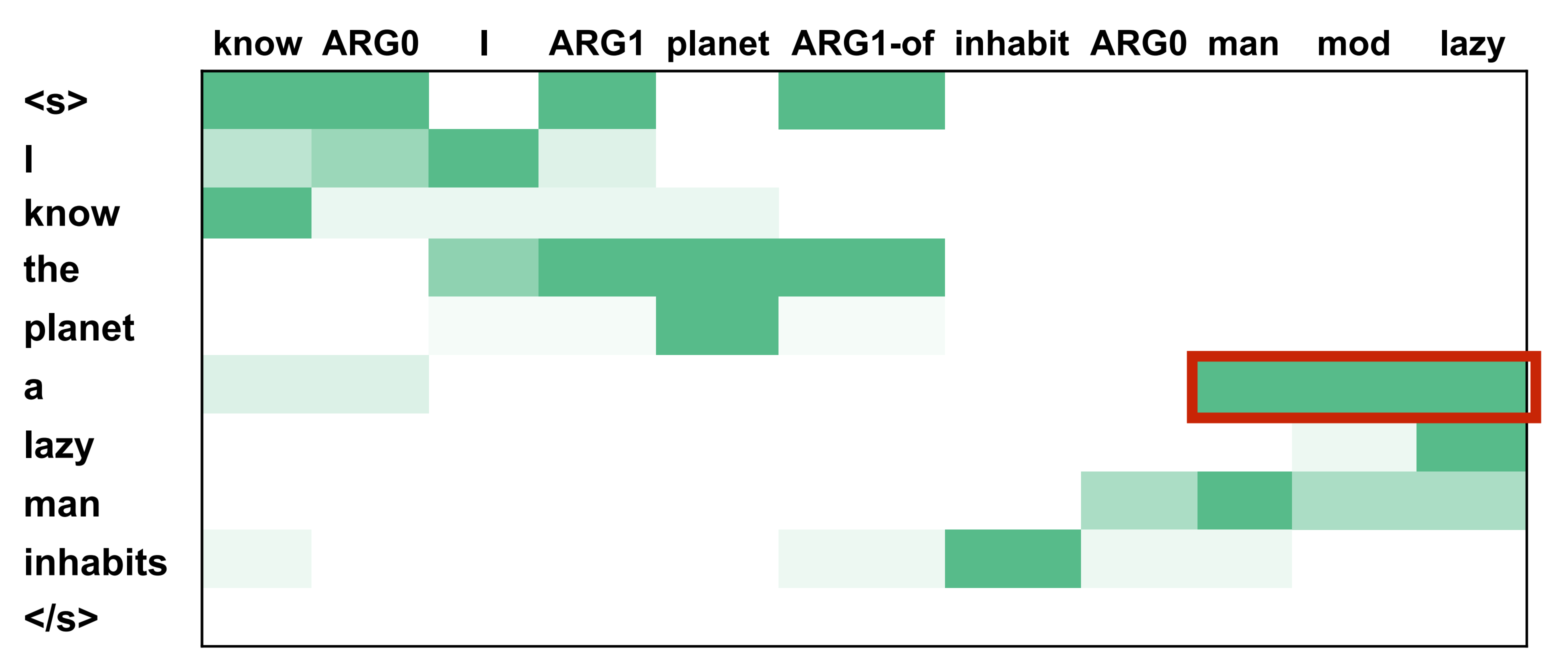


$$\mathbf{a}_i = softmax(\mathbf{f}_i(\mathbf{h}^{(s)}, \mathbf{h}_{i-1}^{(t)}))$$

Attention



$$\mathbf{a}_i = softmax(\mathbf{f}_i(\mathbf{h}^{(s)}, \mathbf{h}_{i-1}^{(t)}))$$



Issues to Address

Max-probability search

$$\mathbf{w}^* = \arg\max_{w} sc\big(\mathbf{t}^{(s)}, \mathbf{w}\big)$$

where $sc\big(\mathbf{t}^{(s)}, \mathbf{w}\big) = p\big(\mathbf{w}|\mathbf{t}^{(s)}\big)$

Issues to Address

Max-probability search

$$\mathbf{w}^* = \arg\max_{w} sc\big(\mathbf{t}^{(s)}, \mathbf{w}\big)$$

where $sc\big(\mathbf{t}^{(s)}, \mathbf{w}\big) = p\big(\mathbf{w}|\mathbf{t}^{(s)}\big)$

Issues

- short / similar outputs
- no guarantee that input is covered

Issues to Address

Issues to Address

Max-probability search

- Length Penalty

$$sc\big(\mathbf{t}^{(s)}, \mathbf{w}\big) = \frac{log\big(p\big(\mathbf{w}|\mathbf{t}^{(s)}\big)\big)}{|\mathbf{t}^{(s)}|^{\alpha}}$$

Issues to Address

Max-probability search

- Length Penalty

$$sc(\mathbf{t}^{(s)}, \mathbf{w}) = \frac{log(p(\mathbf{w}|\mathbf{t}^{(s)}))}{|\mathbf{t}^{(s)}|^{\alpha}}$$

- Coverage Penalty

$$cp(\mathbf{t}^{(s)}, \mathbf{w}) = \beta * \sum_{i=1}^{|\mathbf{t}^{(s)}|} log(min(\sum_{j=1}^{|\mathbf{w}|} a_{i,j}, 1.0))$$

Issues to Address

Max-probability search

- Length Penalty

$$sc(\mathbf{t}^{(s)}, \mathbf{w}) = \frac{log(p(\mathbf{w}|\mathbf{t}^{(s)}))}{|\mathbf{t}^{(s)}|^{\alpha}}$$

- Coverage Penalty

$$cp(\mathbf{t}^{(s)}, \mathbf{w}) = \beta * \sum_{i=1}^{|\mathbf{t}^{(s)}|} log(min(\sum_{j=1}^{|\mathbf{w}|} a_{i,j}, 1.0))$$

- Integrating in model
  - Neural Checklist Model (Kiddon et al, EMNLP 2016)
  - Coverage Model (Tu et al, ACL 2016)
- Structural Biases (Cohn et al, NAACL 2016)
  - Fertility, HMM bias

Issues to Address

Issues to Address

Sparsity

- Anonymize NE tokens

Issues to Address

Sparsity

- Anonymize NE tokens

```
state ARG0 person_name_0 ARG1
keep ARG0 country_name_1 …
```

President Obama stated that **UK** should keep …

person_name_0 stated that **country_name_1** should keep …

Issues to Address

Sparsity

- Anonymize NE tokens

```
state ARG0 person_name_0 ARG1
keep ARG0 country_name_1 …
```

President Obama stated that **UK** should keep …

person_name_0 stated that **country_name_1** should keep …

- Copy from input

Issues to Address

Sparsity

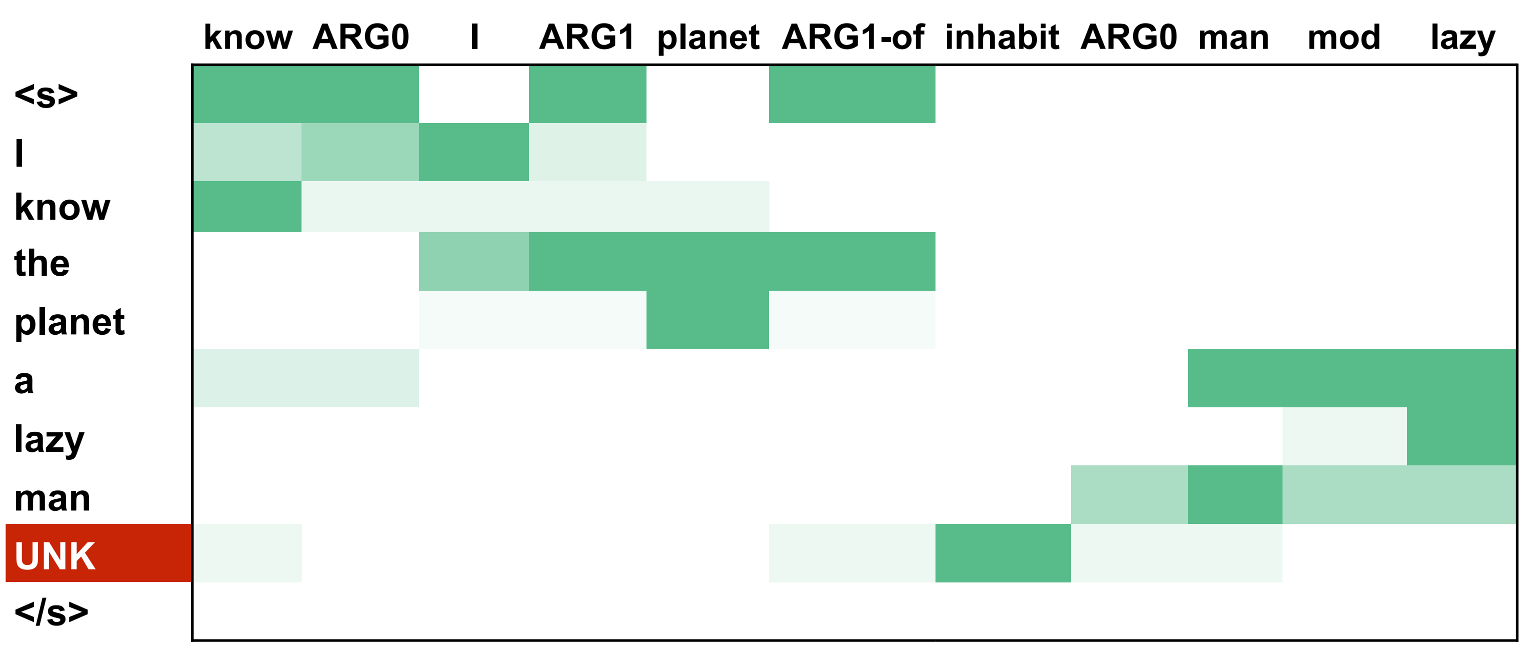
- Anonymize NE tokens

state ARG0 person_name_0 ARG1
keep ARG0 country_name_1 …

President Obama stated that **UK** should keep …

person_name_0 stated that **country_name_1** should keep …

- Copy from input



Issues to Address

Sparsity

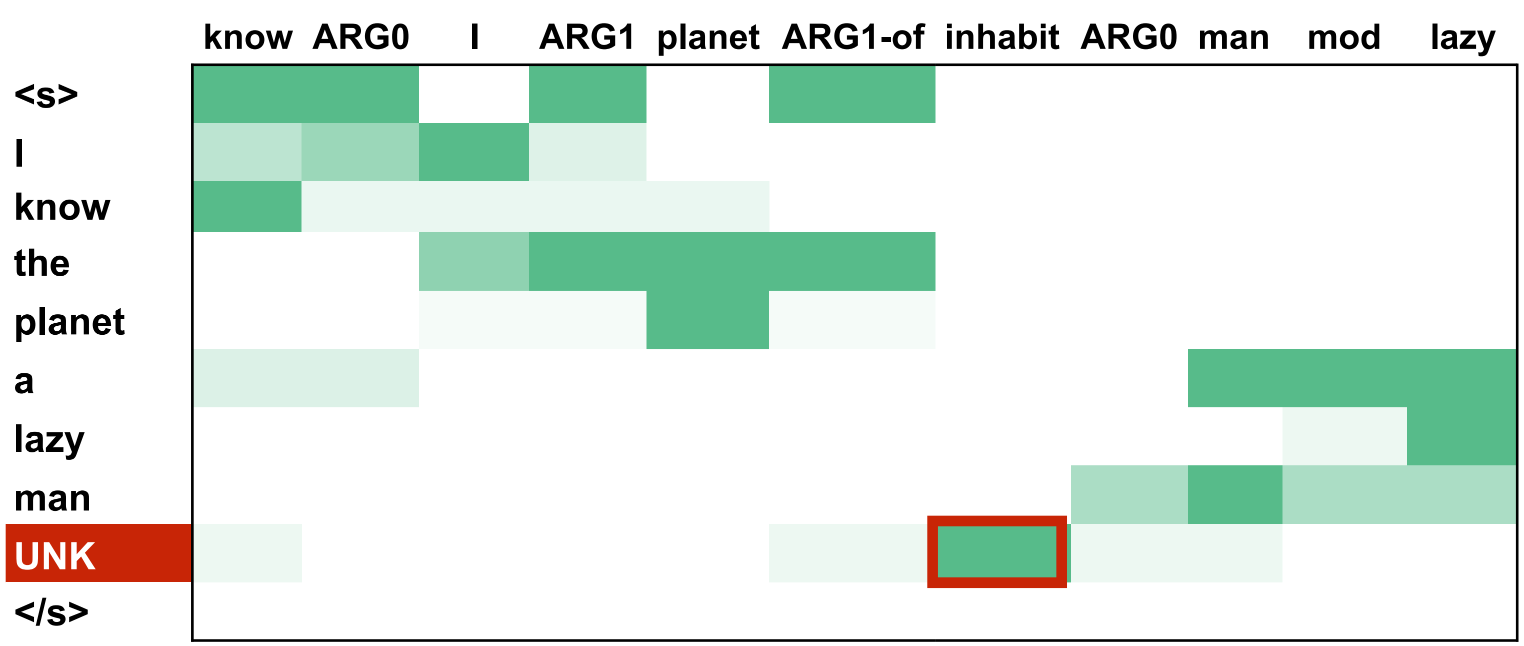
- Anonymize NE tokens

```
state ARG0 person_name_0 ARG1
keep ARG0 country_name_1 …
```

President Obama stated that **UK** should keep …

person_name_0 stated that **country_name_1** should keep …

- Copy from input



Issues to Address

Sparsity

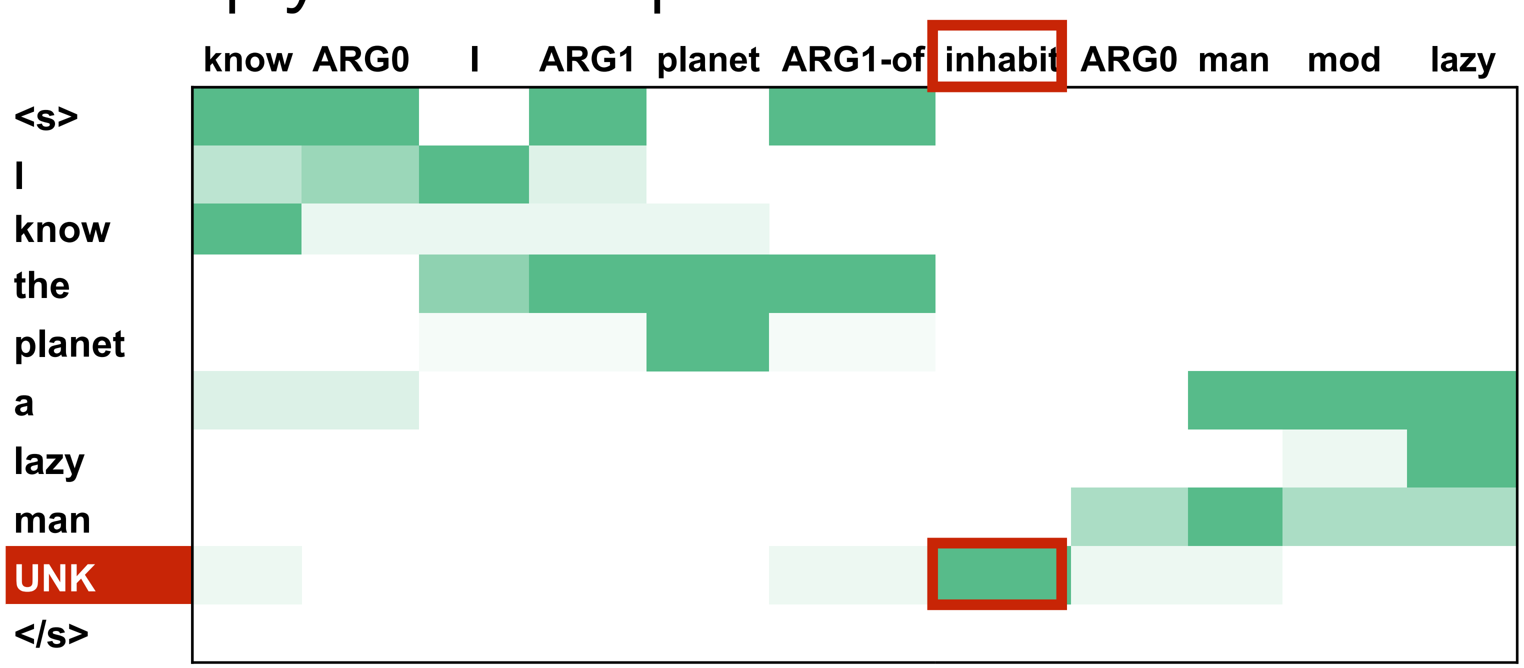
- Anonymize NE tokens

```
state ARG0 person_name_0 ARG1
keep ARG0 country_name_1 …
```

President Obama stated that **UK** should keep …

person_name_0 stated that **country_name_1** should keep …

- Copy from input



Issues to Address

Sparsity

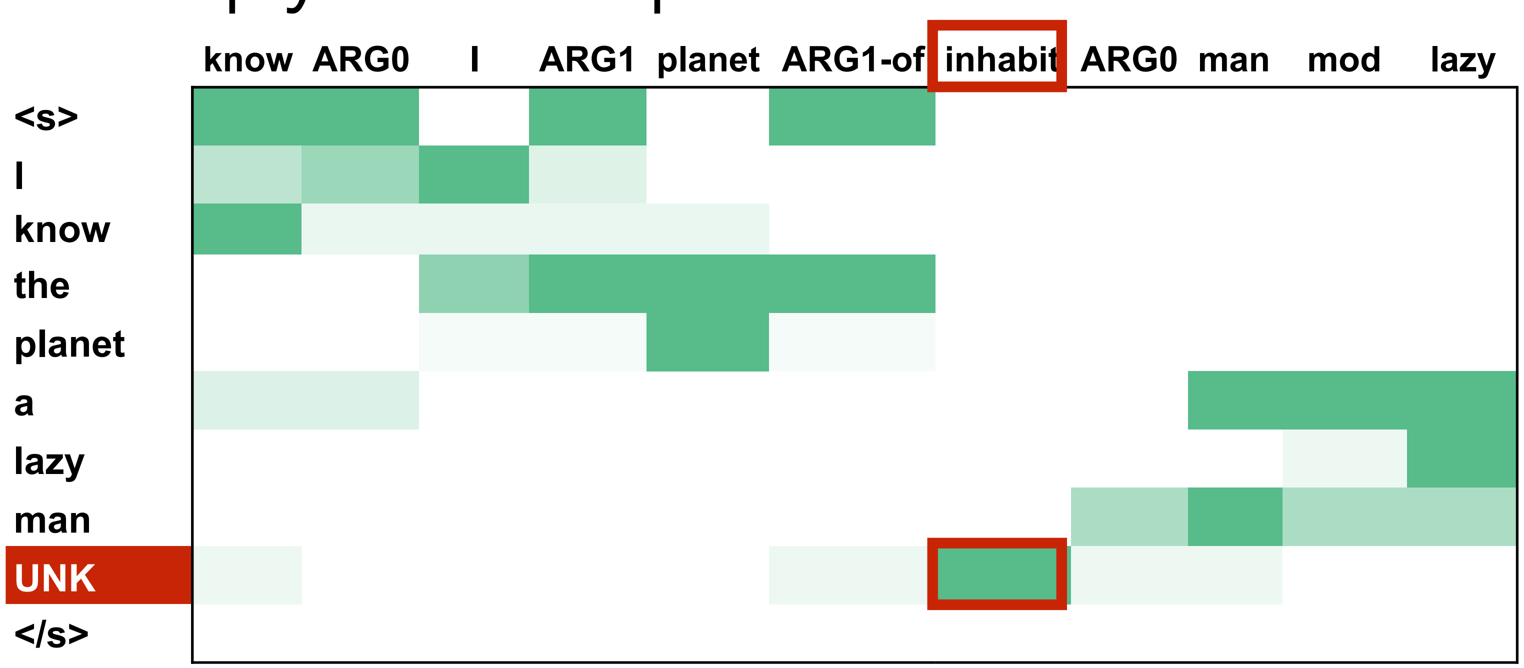
- Anonymize NE tokens

`state ARG0 person_name_0 ARG1 keep ARG0 country_name_1 …`

President Obama stated that **UK** should keep …

person_name_0 stated that **country_name_1** should keep …

- Copy from input



input	output	prob
`inhabit`	**inhabits**	**0.6**
	inhabit	0.2
	inhabiting	0.1
	…	…

Issues to Address

Sparsity

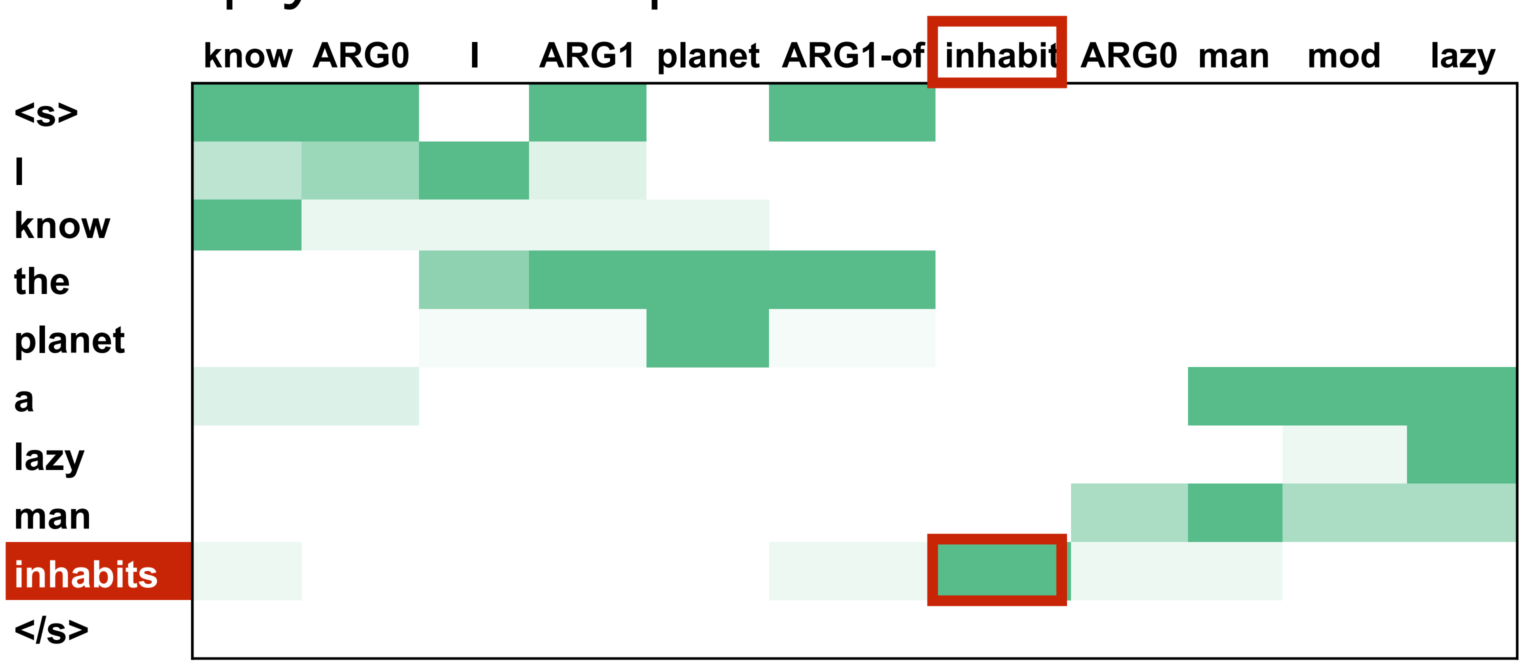
- Anonymize NE tokens

state ARG0 person_name_0 ARG1
keep ARG0 country_name_1 …

President Obama stated that UK should keep …

person_name_0 stated that country_name_1 should keep …

- Copy from input



input	output	prob
inhabit	**inhabits**	**0.6**
	inhabit	0.2
	inhabiting	0.1
	…	…

Issues to Address

Sparsity

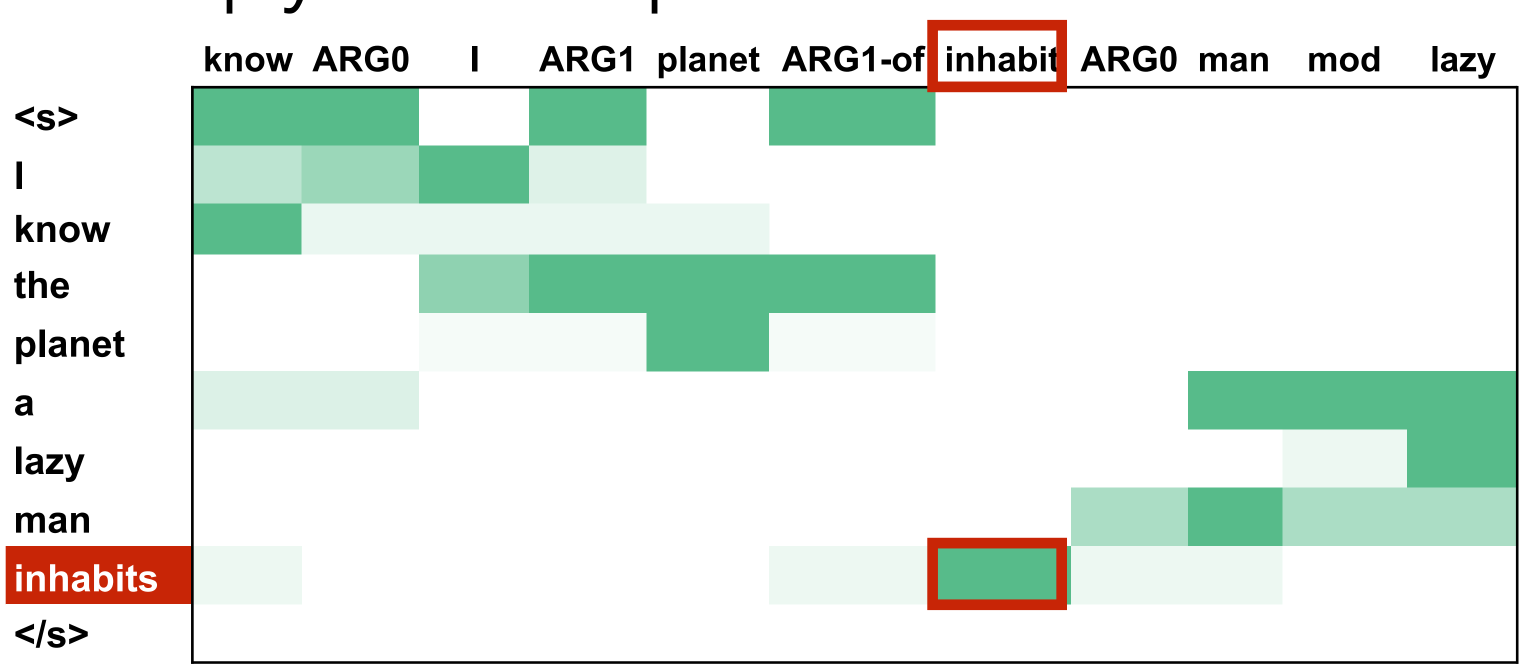
- Anonymize NE tokens

`state ARG0 person_name_0 ARG1 keep ARG0 country_name_1 …`

President Obama stated that **UK** should keep …

person_name_0 stated that **country_name_1** should keep …

- Copy from input



input	output	prob
`inhabit`	**inhabits**	**0.6**
	inhabit	0.2
	inhabiting	0.1
	…	…

- Data Augmentation (Sennrich et al, ACL 2016)

Open Questions

Open Questions

Representations

- Probably shouldn't treat all inputs as strings…

Open Questions

Representations

- Probably shouldn't treat all inputs as strings…

Loss on some intermediate / latent goal

- Don't want just good-looking string of [X_language]…

Open Questions

Representations

- Probably shouldn't treat all inputs as strings…

Loss on some intermediate / latent goal

- Don't want just good-looking string of [X_language]…

Document Plans

- Maybe shouldn't treat output as stream of strings…

Thank You