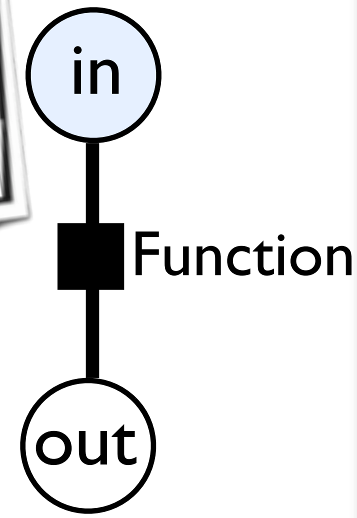
Graphical Models over Multiple Strings

Markus Dreyer and Jason Eisner
Center for Language and Speech Processing (CLSP)
Center of Excellence in Language and Speech Processing (COE)
Computer Science Department (CS)
Johns Hopkins University (JHU)

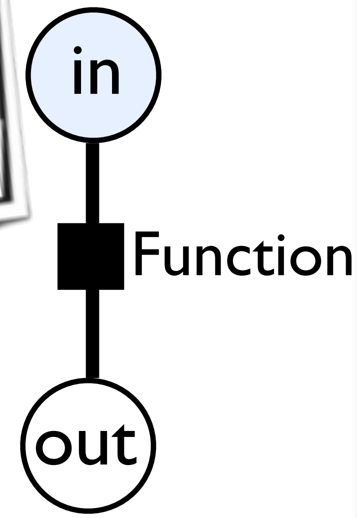
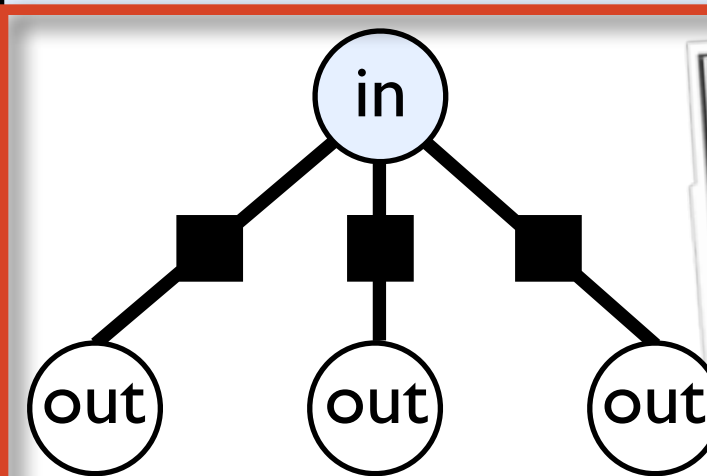
EMNLP 2009

human language technology
center of excellence

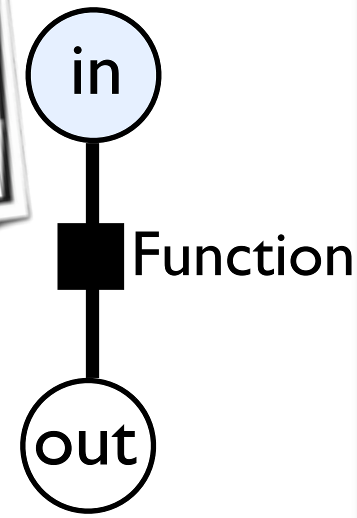
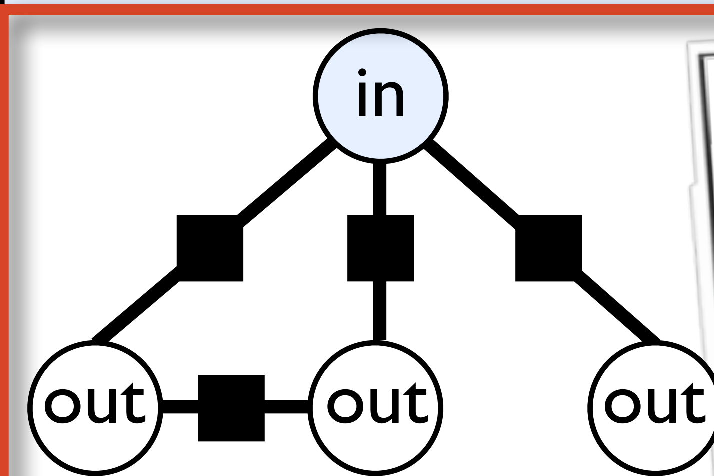
Motivation

	single prediction	**joint prediction**
simple variables	**in**: text, **out**: topic ID 	
complex variables		this talk goes here!

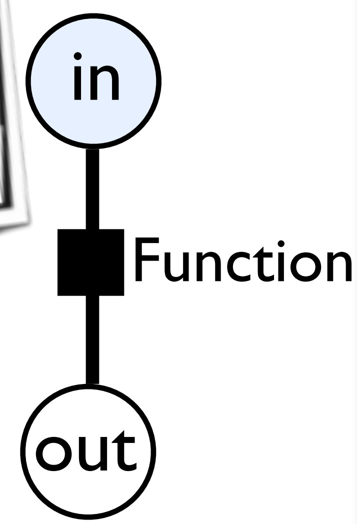
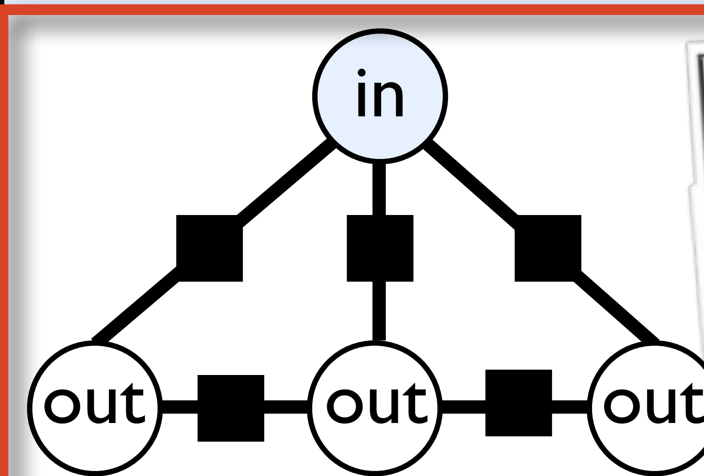
Motivation

	single prediction	**joint prediction**
simple variables	**in**: text, **out**: topic ID 	 **in**: text, **out**: tag sequence *(CRF, ...)*
complex variables		this talk goes here!

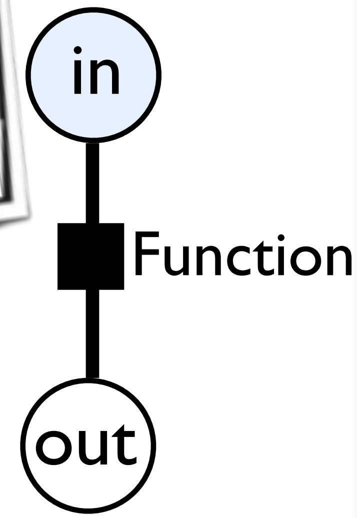
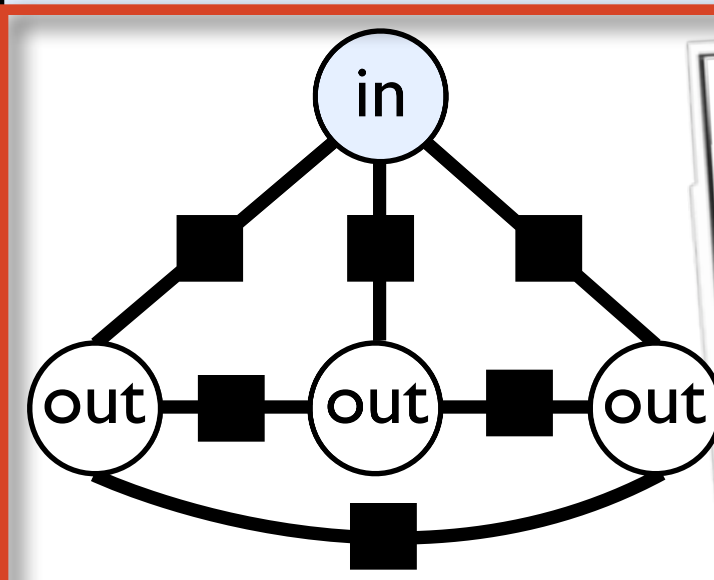
Motivation

	single prediction	**joint prediction**
simple variables	**in**: text, **out**: topic ID 	 **in**: text, **out**: tag sequence *(CRF, ...)*
complex variables		this talk goes here!

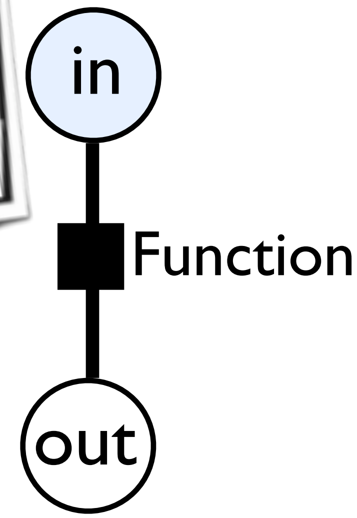
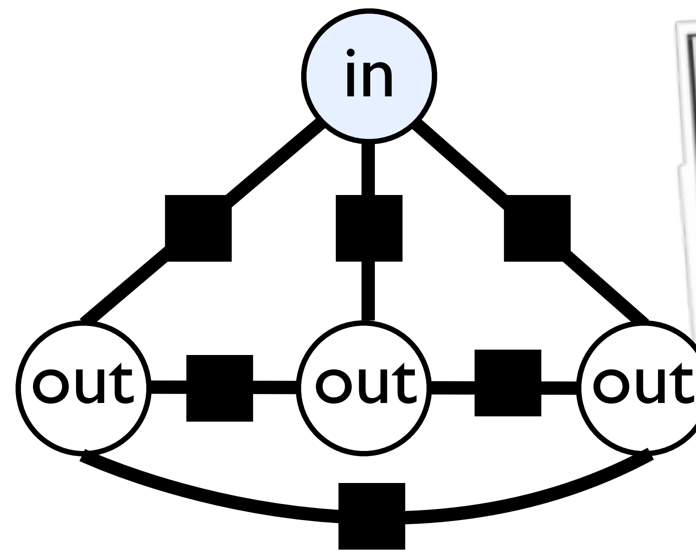
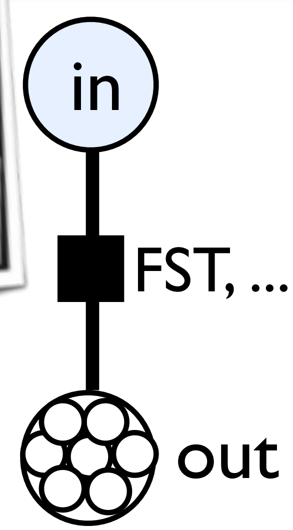
Motivation

	single prediction	joint prediction
simple variables	**in**: text, **out**: topic ID 	 **in**: text, **out**: tag sequence *(CRF, ...)*
complex variables		this talk goes here!

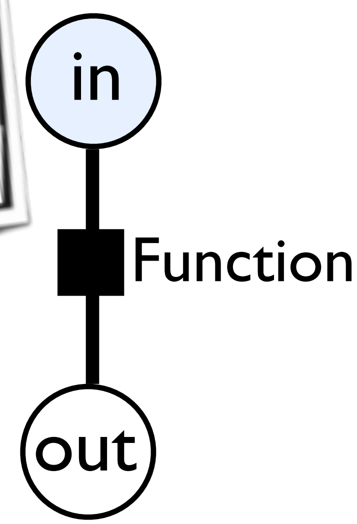
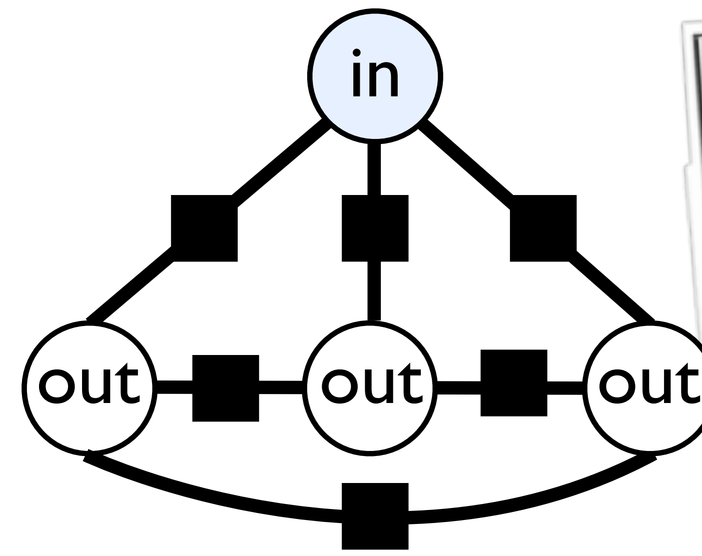
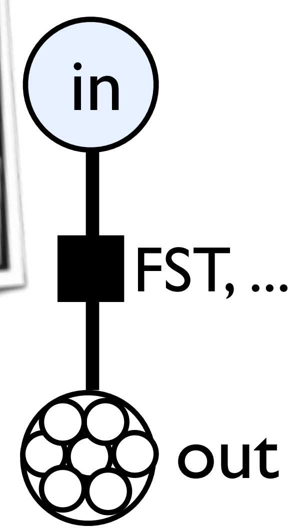
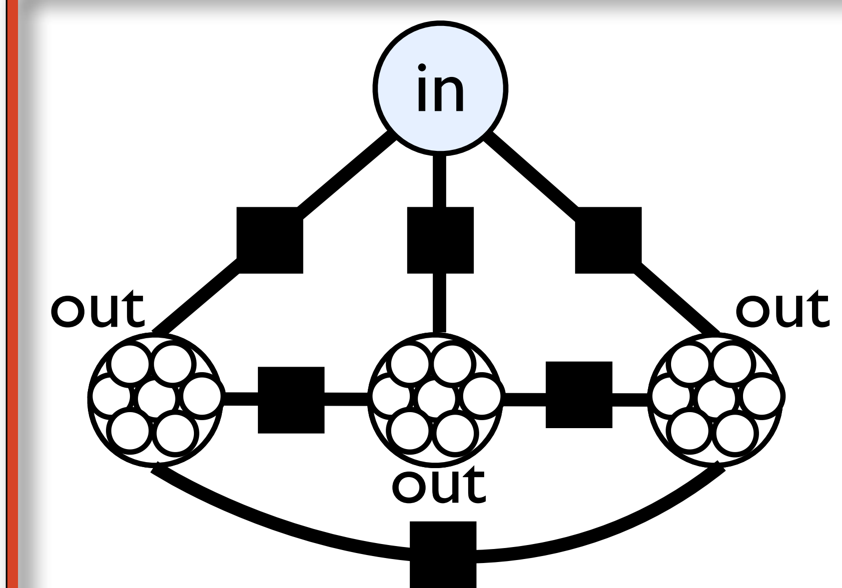
Motivation

	single prediction	**joint prediction**
simple variables	**in**: text, **out**: topic ID 	 **in**: text, **out**: tag sequence *(CRF, ...)*
complex variables		this talk goes here!

Motivation

	single prediction	**joint prediction**
simple variables	**in**: text, **out**: topic ID 	 **in**: text, **out**: tag sequence *(CRF, ...)*
complex variables	**in**: word, **out**: trans-literation, ... 	this talk goes here!

Motivation

	single prediction	joint prediction
simple variables	**in**: text, **out**: topic ID 	 **in**: text, **out**: tag sequence *(CRF, ...)*
complex variables	**in**: word, **out**: trans-literation, ... 	

Motivation. *Example tasks*

Morphology

infinitive	brech**en**

	present		past	
	singular	plural	singular	plural
1st	brech**e** ?	brech**en** ?	brach ?	br**a**ch**en** ?
2nd	br**i**ch**st** ?	brech**t** ?	br**a**ch**st** ?	brach**t** ?
3rd	br**i**ch**t** ?	brech**en** ?	br**a**ch ?	br**a**ch**en** ?

Motivation. *Example tasks*

Morphology

infinitive	brech**en**

1st	brech**e**	brech**en**	br**a**ch	br**a**ch**en**
2nd	br**i**ch**st**	brech**t**	br**a**ch**st**	br**a**ch**t**
3rd	br**i**ch**t**	brech**en**	br**a**ch	br**a**ch**en**
	singular	plural	singular	plural
	present		past	

Motivation. *Example tasks*

Morphology

infinitive	brech**en**

1st	brech**e**	brech**en**	br**a**ch	br**a**ch**en**
2nd	br**i**ch**st**	brech**t**	br**a**ch**st**	br**a**ch**t**
3rd	br**i**ch**t**	brech**en**	br**a**ch	br**a**ch**en**
	singular	plural	singular	plural
	present		past	

Motivation. *Example tasks*

Morphology

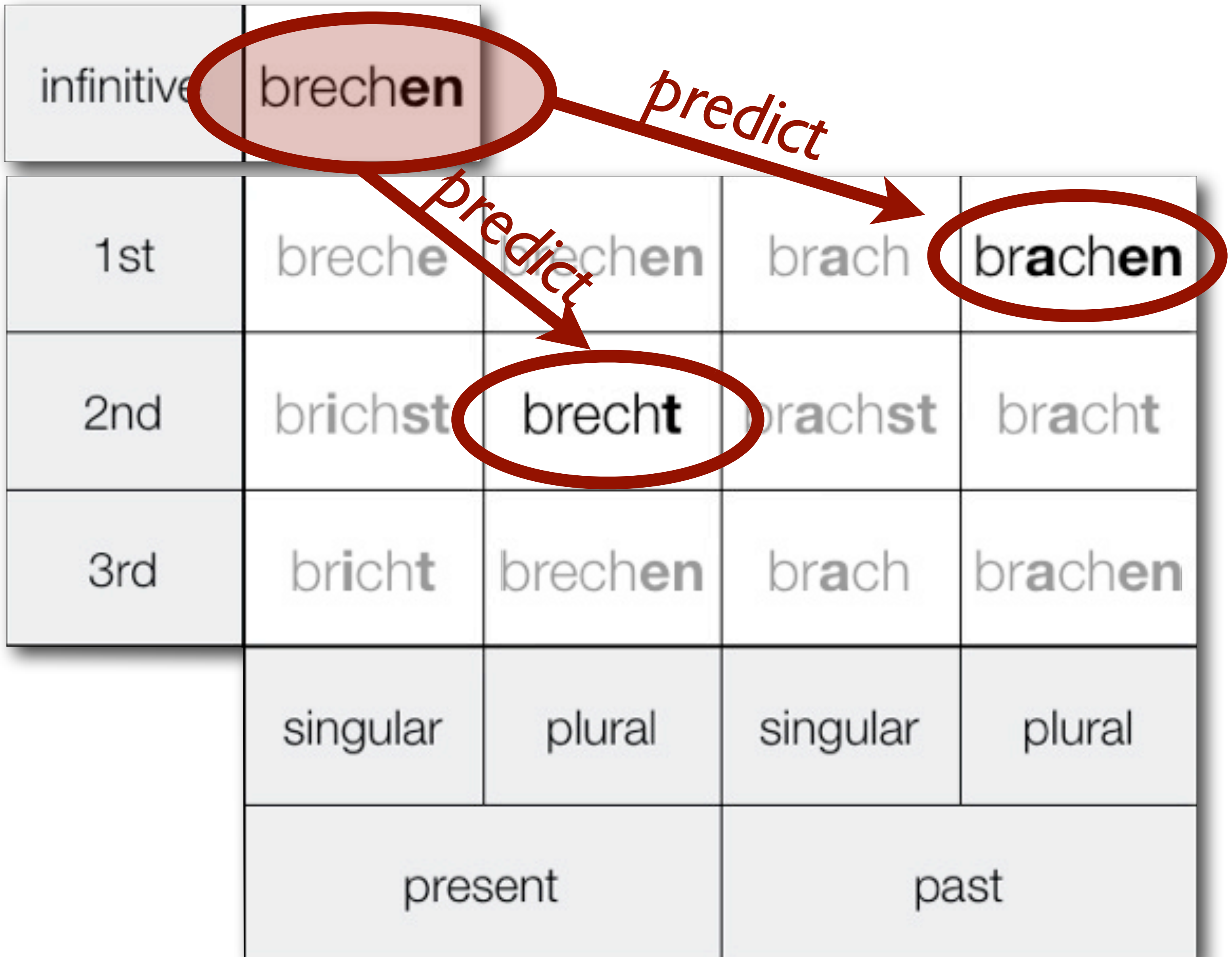
infinitive	brechen			
1st	breche	brechen	brach	brachen
2nd	brichst	brecht	brachst	bracht
3rd	bricht	brechen	brach	brachen
	singular	plural	singular	plural
	present		past	

predict

predict

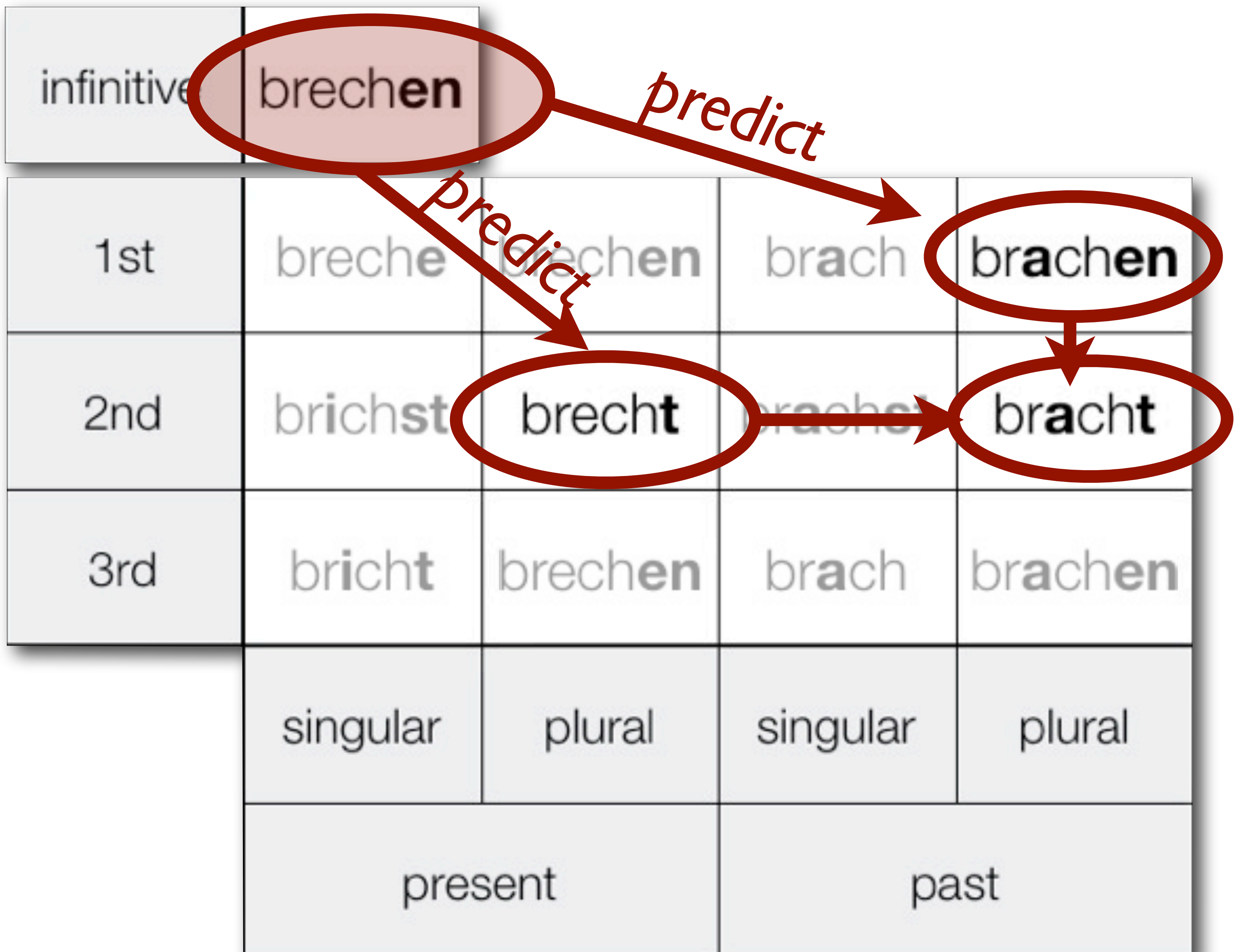
Motivation. *Example tasks*

Morphology



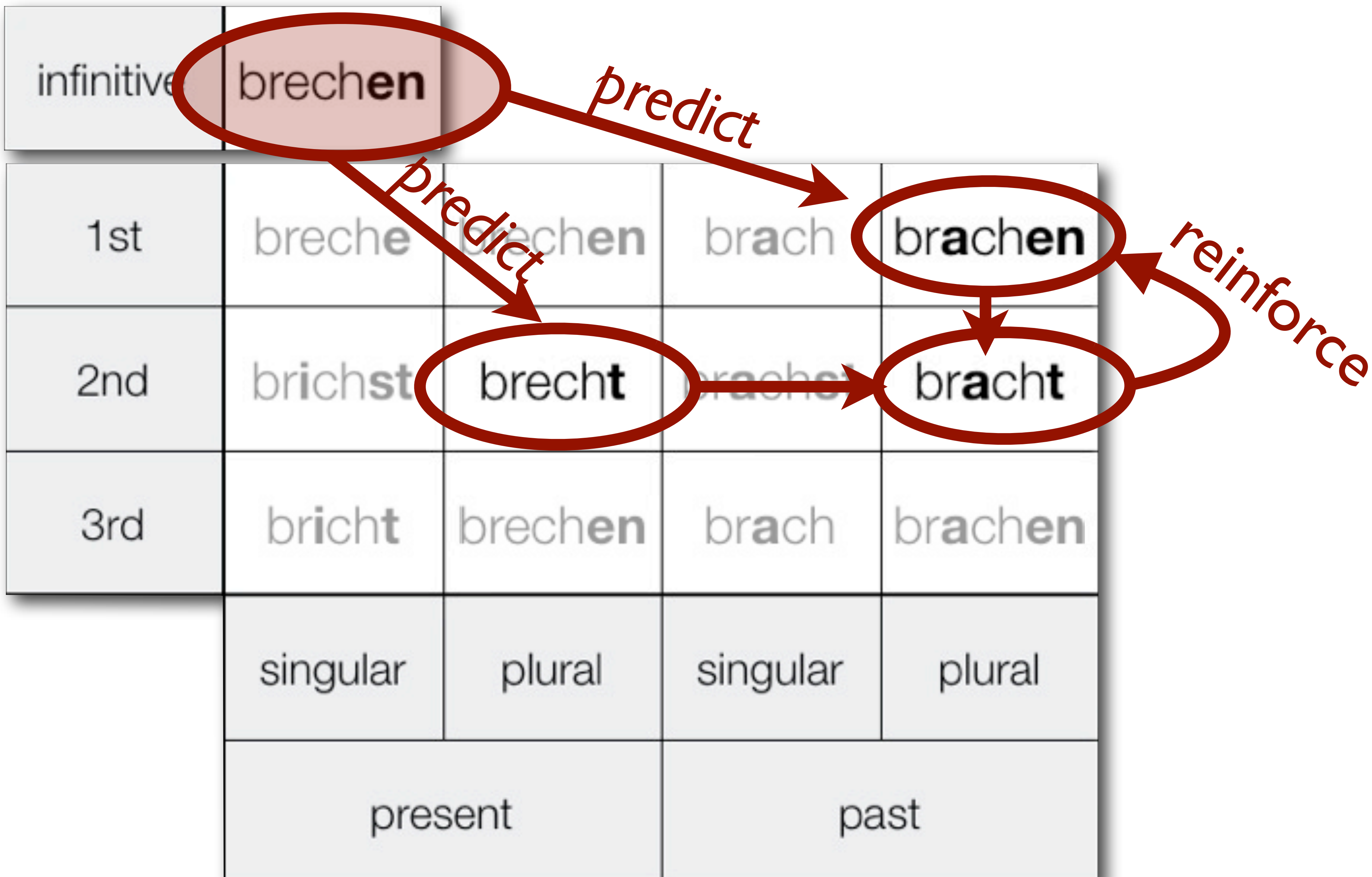
Motivation. *Example tasks*

Morphology



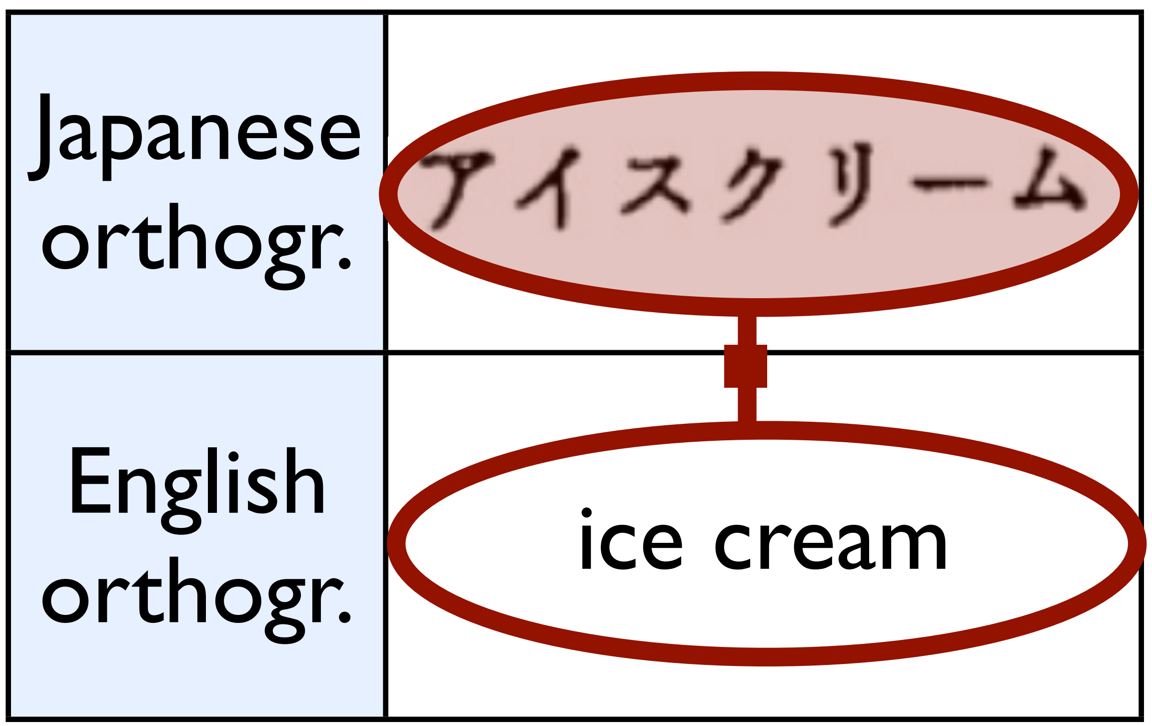
Motivation. *Example tasks*

Morphology



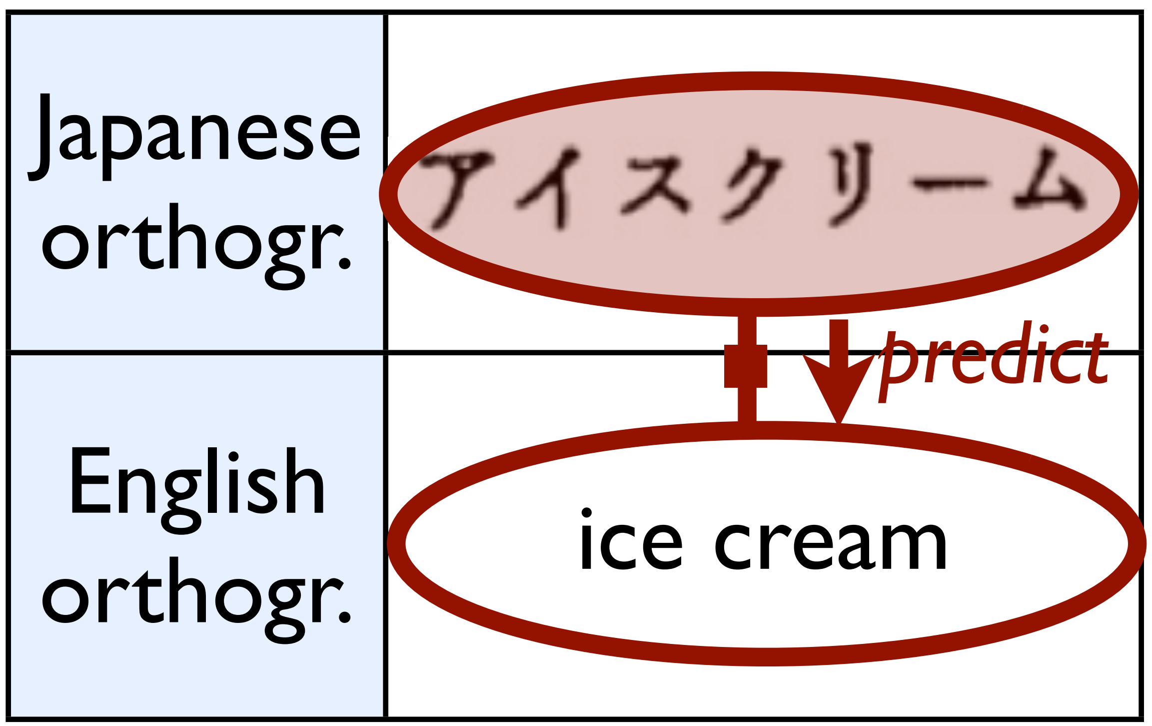
Motivation. *Example tasks*

Transliteration



Motivation. *Example tasks*

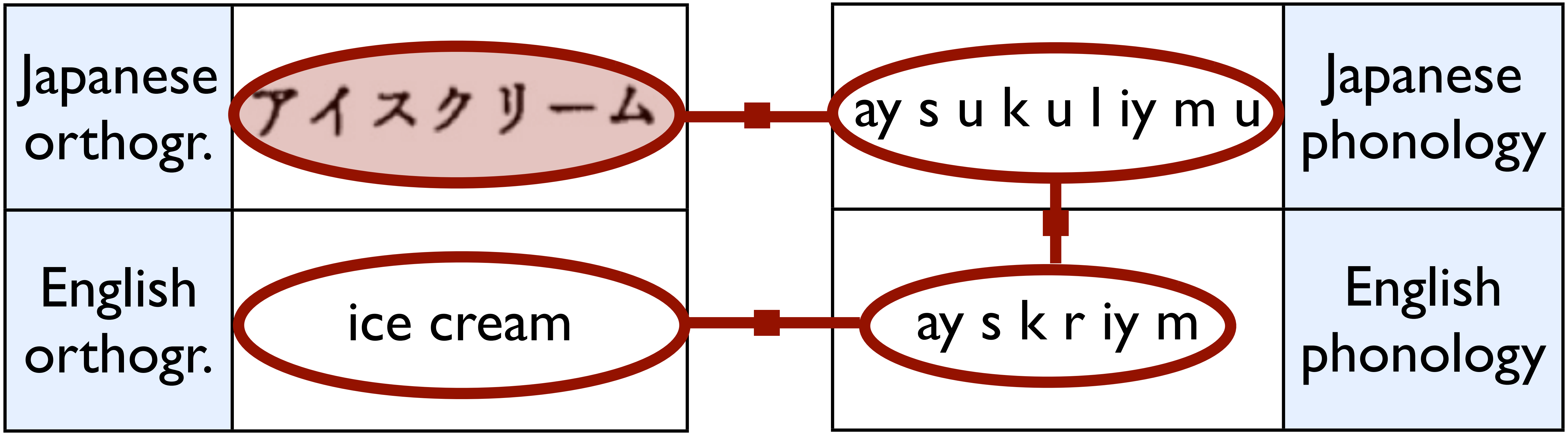
Transliteration



Motivation. *Example tasks*

Transliteration

hidden pronunciations

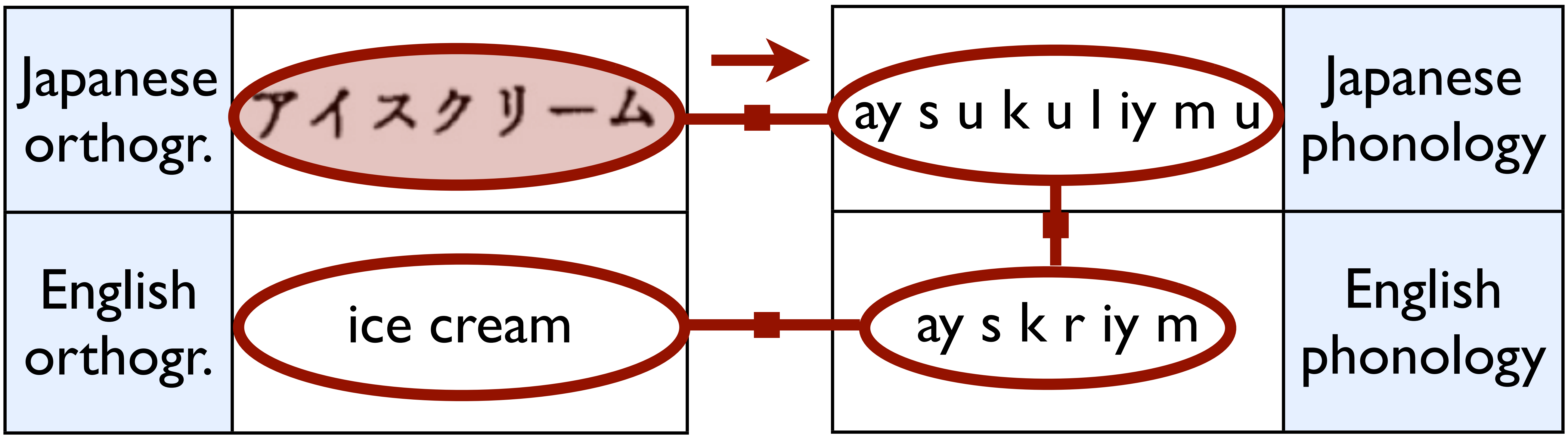


Knight & Graehl 1997

Motivation. *Example tasks*

Transliteration

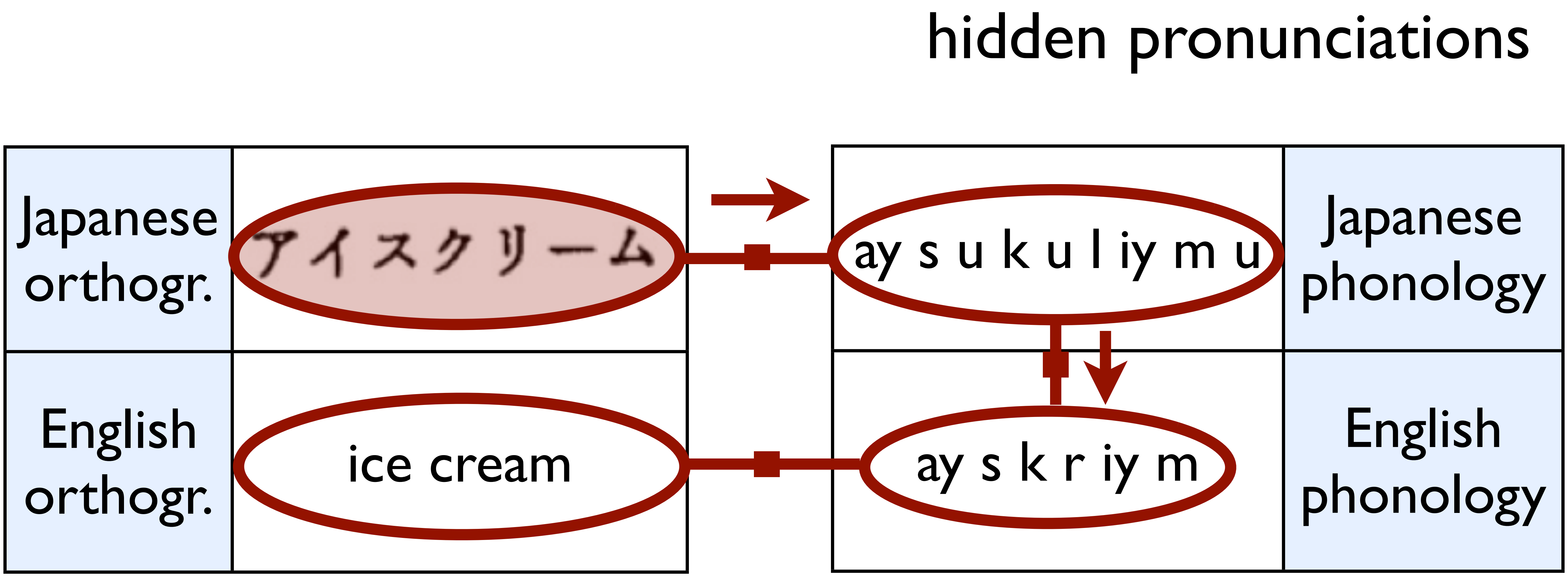
hidden pronunciations



Knight & Graehl 1997

Motivation. *Example tasks*

Transliteration

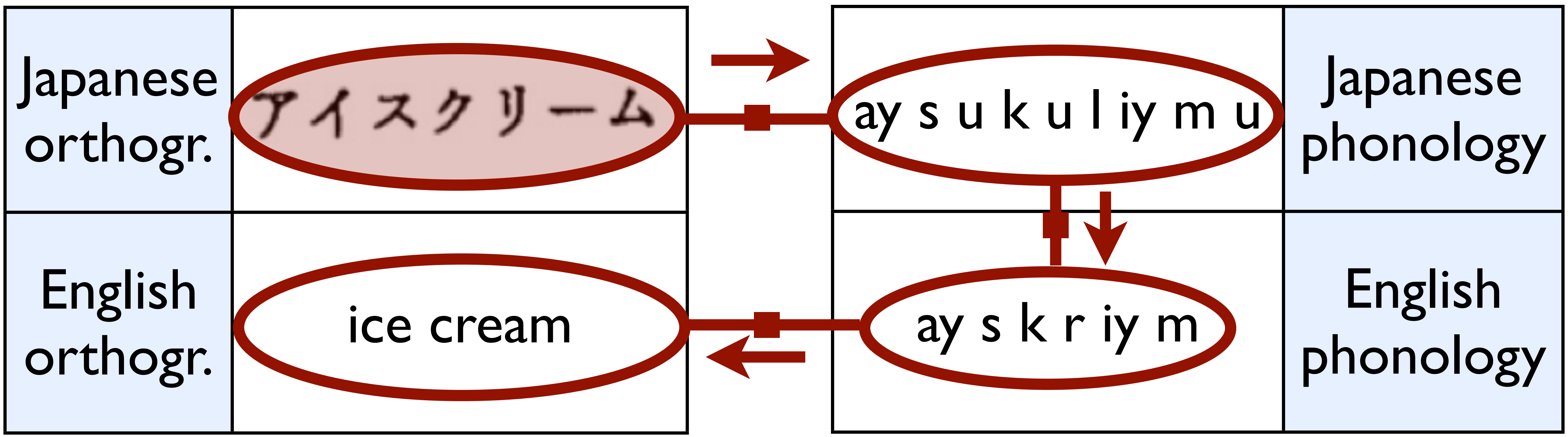


Knight & Graehl 1997

Motivation. *Example tasks*

Transliteration

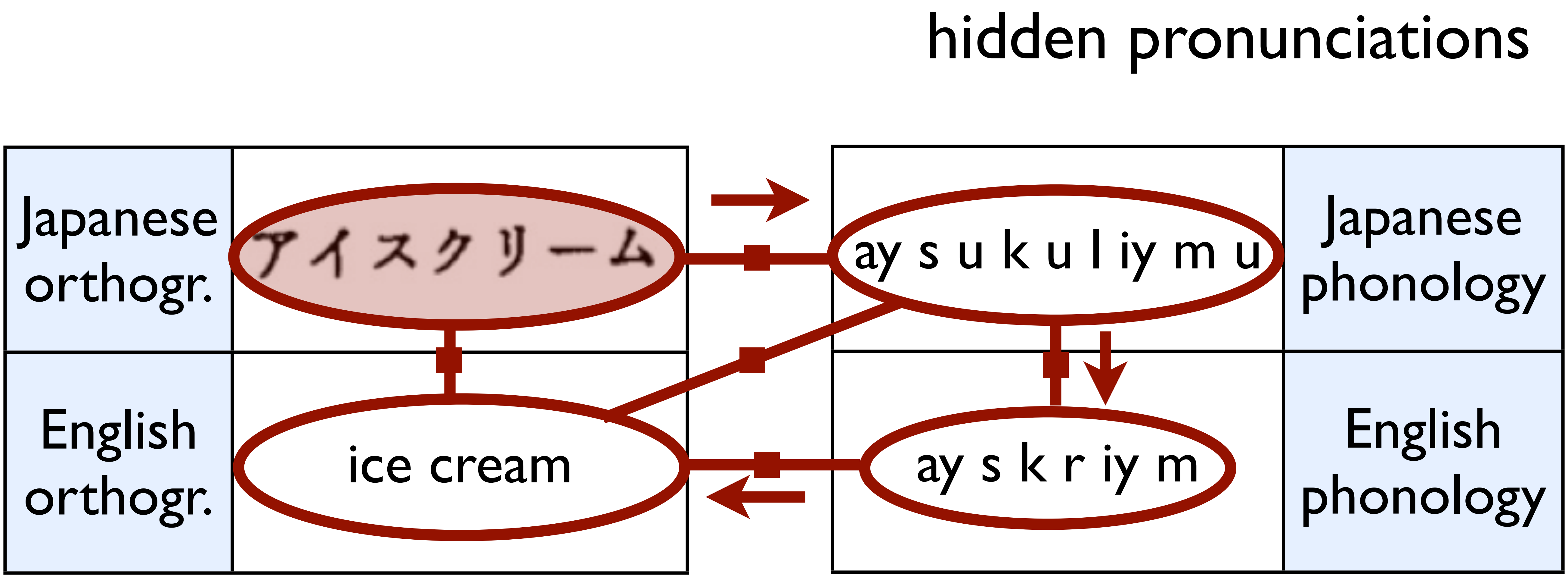
hidden pronunciations



Knight & Graehl 1997

Motivation. *Example tasks*

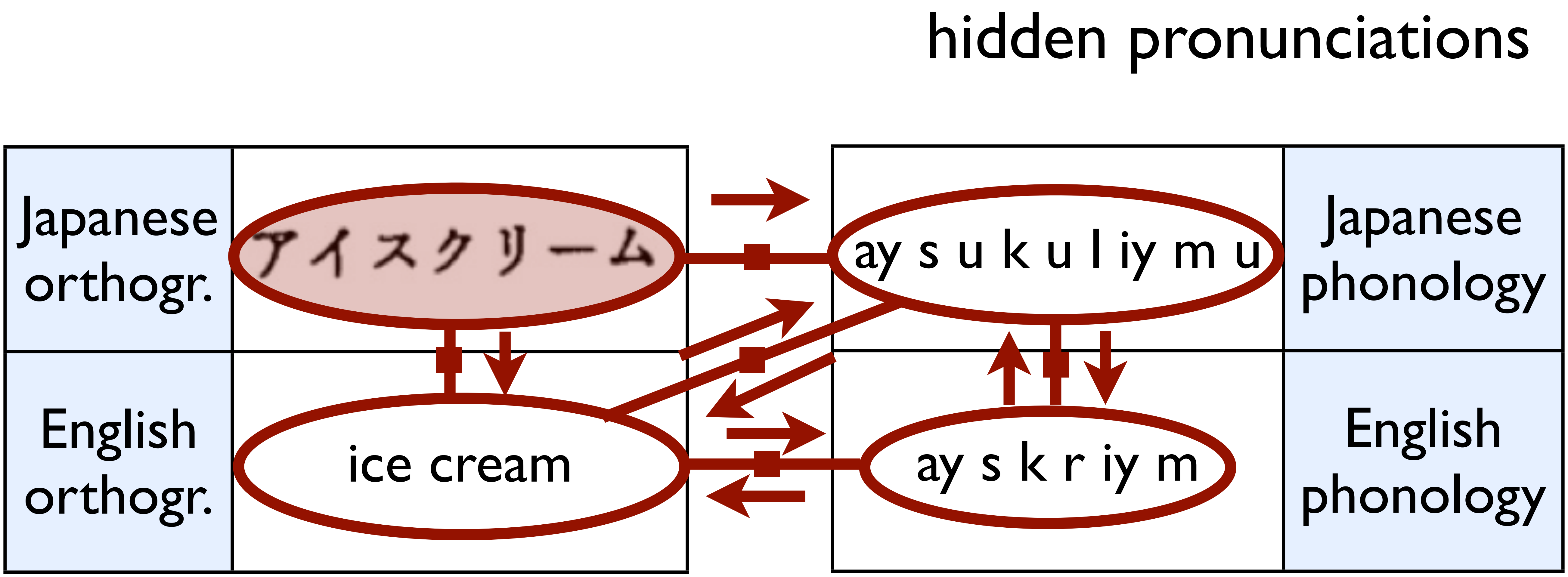
Transliteration



Add **arbitrary** piecewise factors!

Motivation. *Example tasks*

Transliteration



Add **arbitrary** piecewise factors!

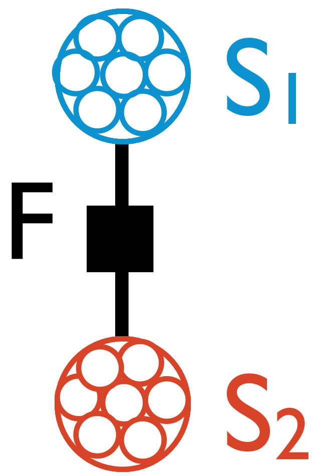
Motivation. *Example tasks*

- Further examples:
  - Cognate modeling
  - Multiple-string alignment
  - System combination

Overview

- Motivation
- **Model**
- Inference & Approximations
- Experiments
- Conclusions

Model. *Getting started: 2 strings*

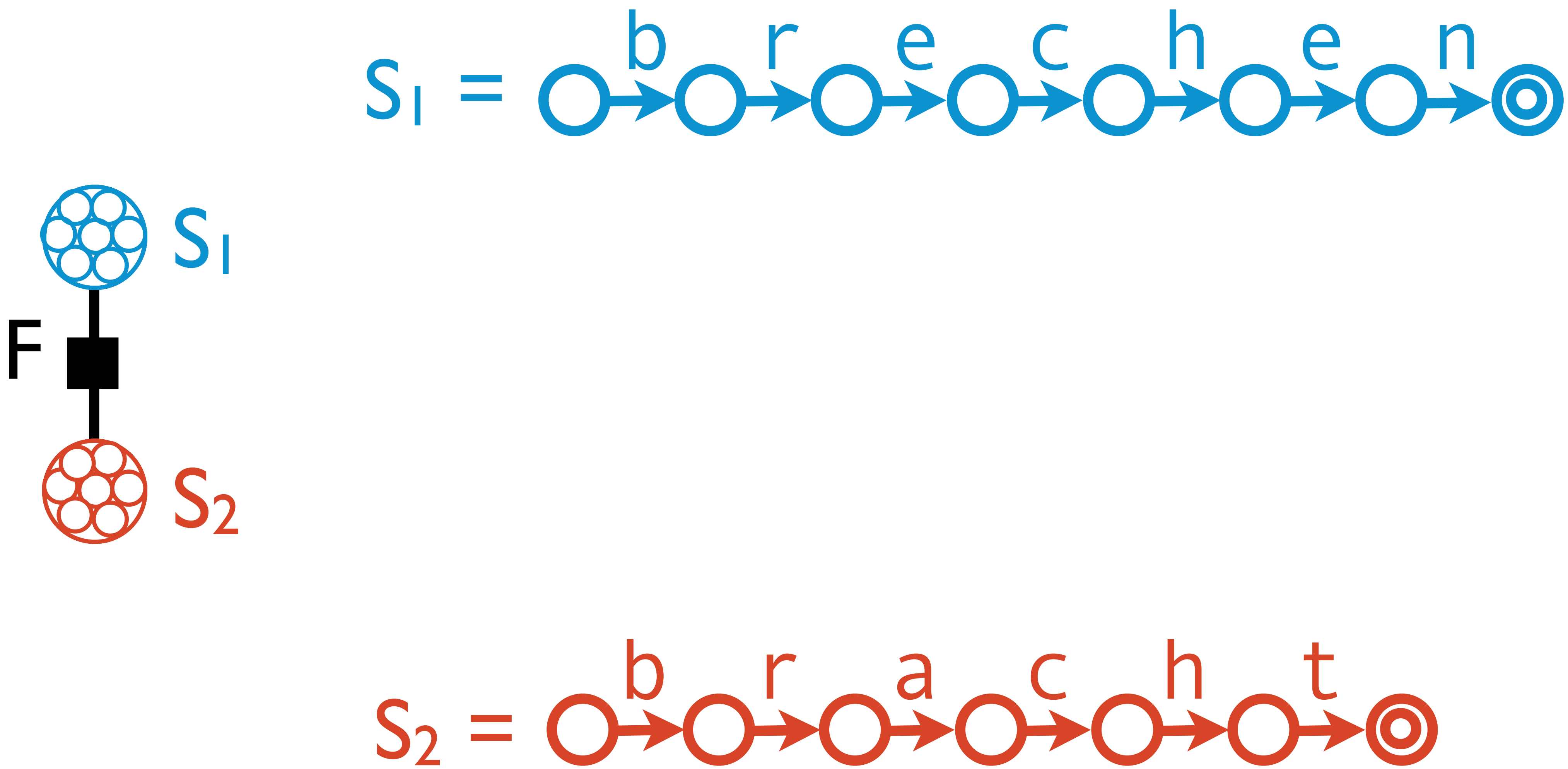


- Suppose we have a **probability distribution over two string variables** S_1 and S_2
- Construct **weighted finite-state transducer F** that can assign a **score** to any values of the strings s_1, s_2.

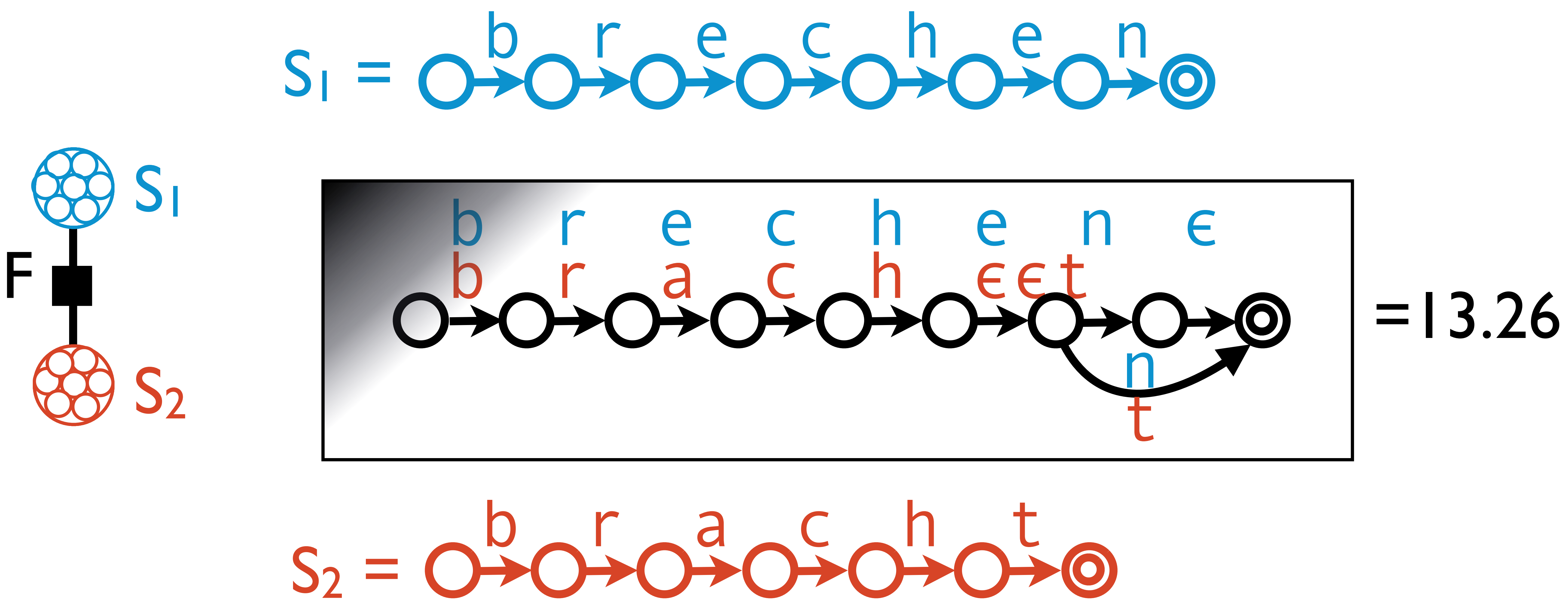
$$\Pr(s_1,s_2) = 1/Z\ F(s_1,s_2)$$

Dreyer, Smith & Eisner, 2008

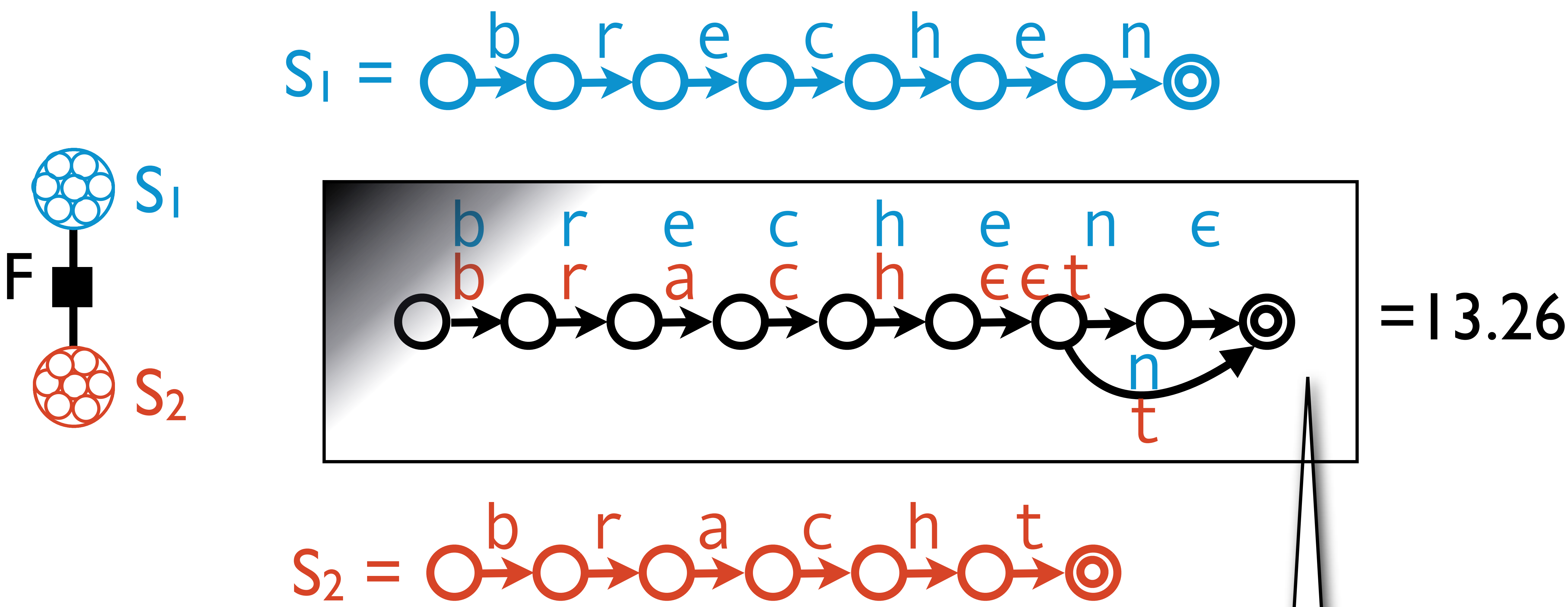
Model. *2 strings: An example*



Model. *2 strings: An example*

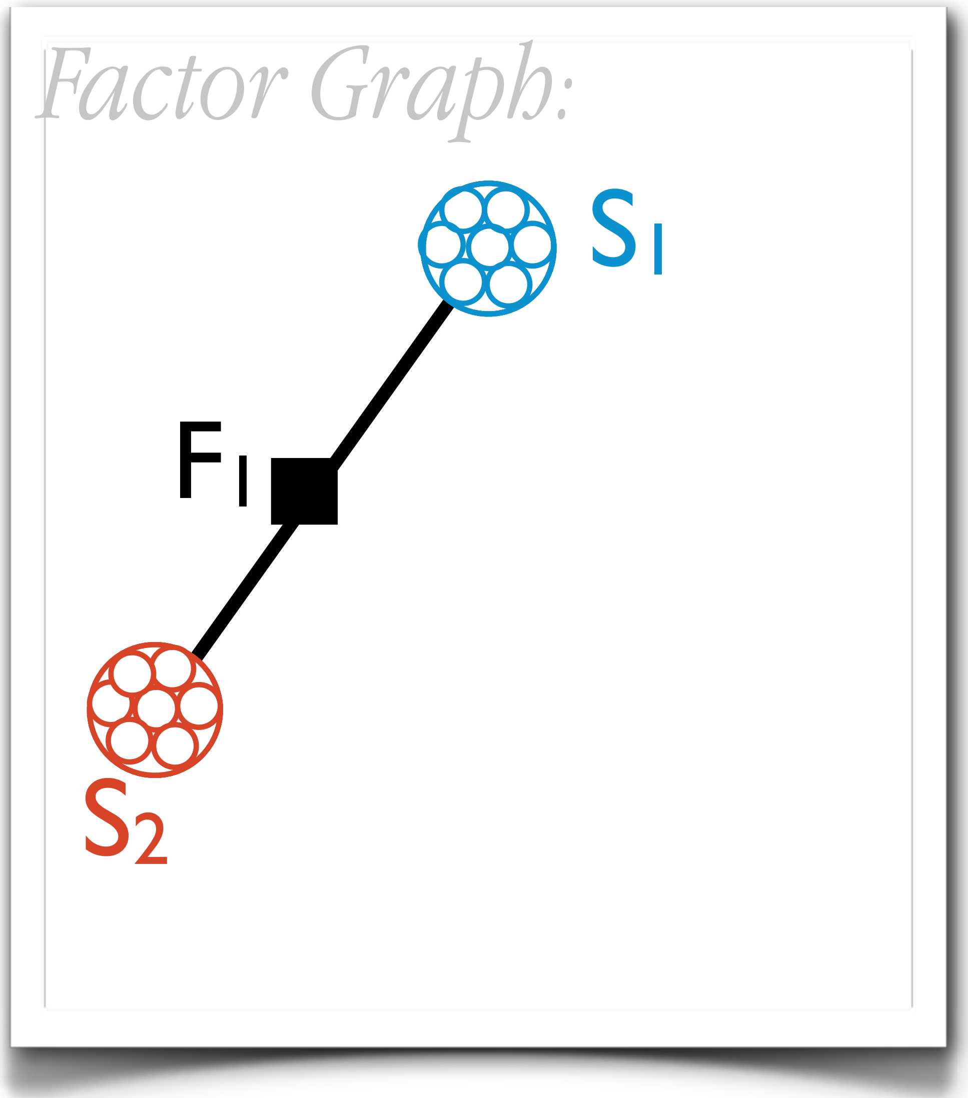


Model. *2 strings: An example*



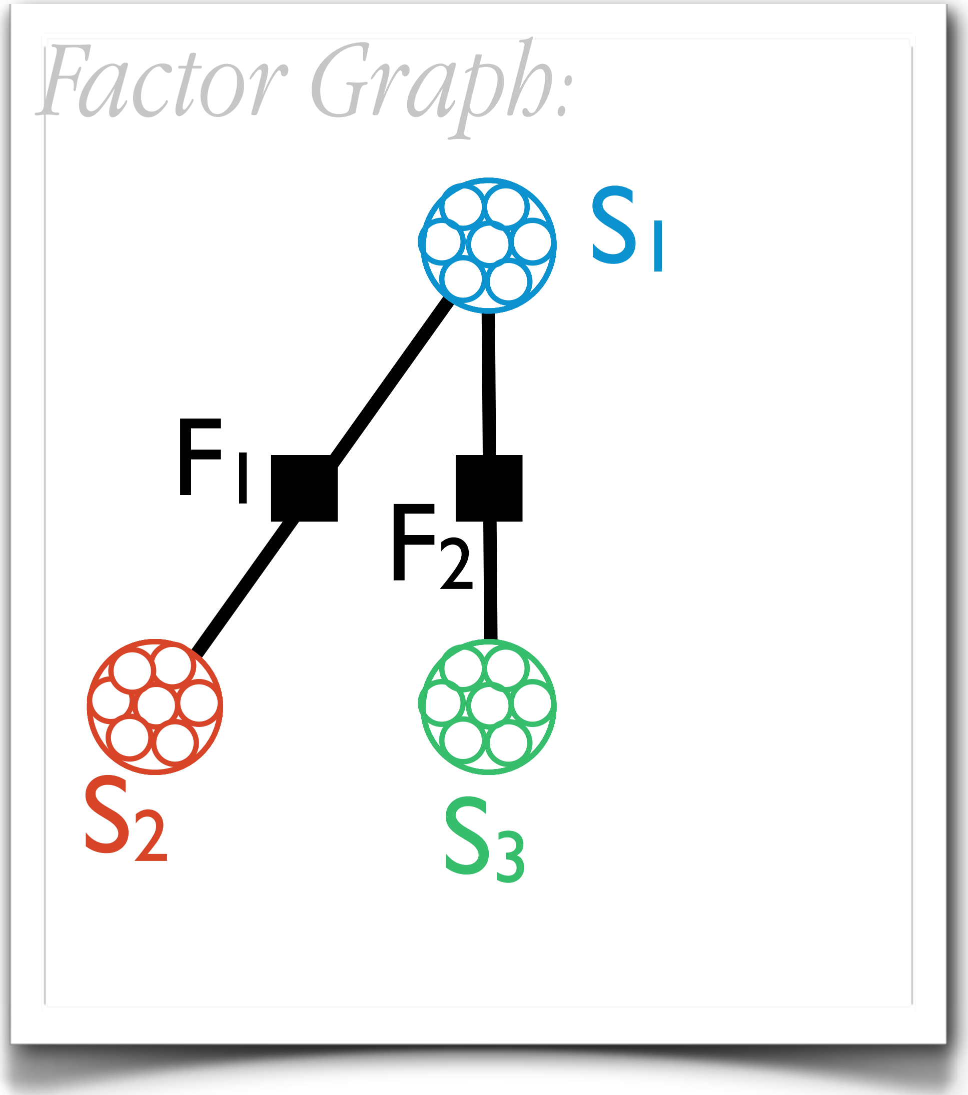
Transducer F computes score by looking at **all** alignments

Model. *Factor graph examples*



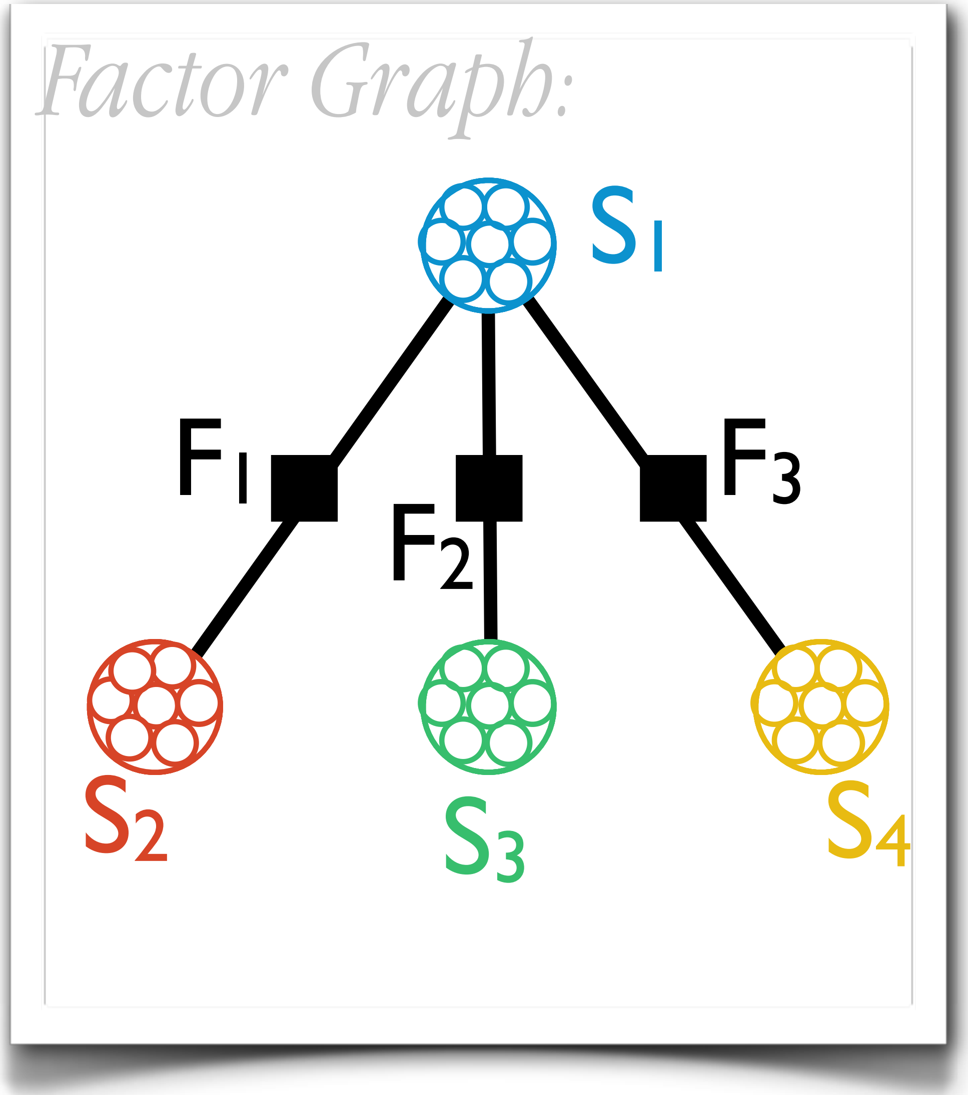
$$\Pr(s_1, s_2) = 1/Z \times F_1(s_1, s_2)$$

Model. *Factor graph examples*



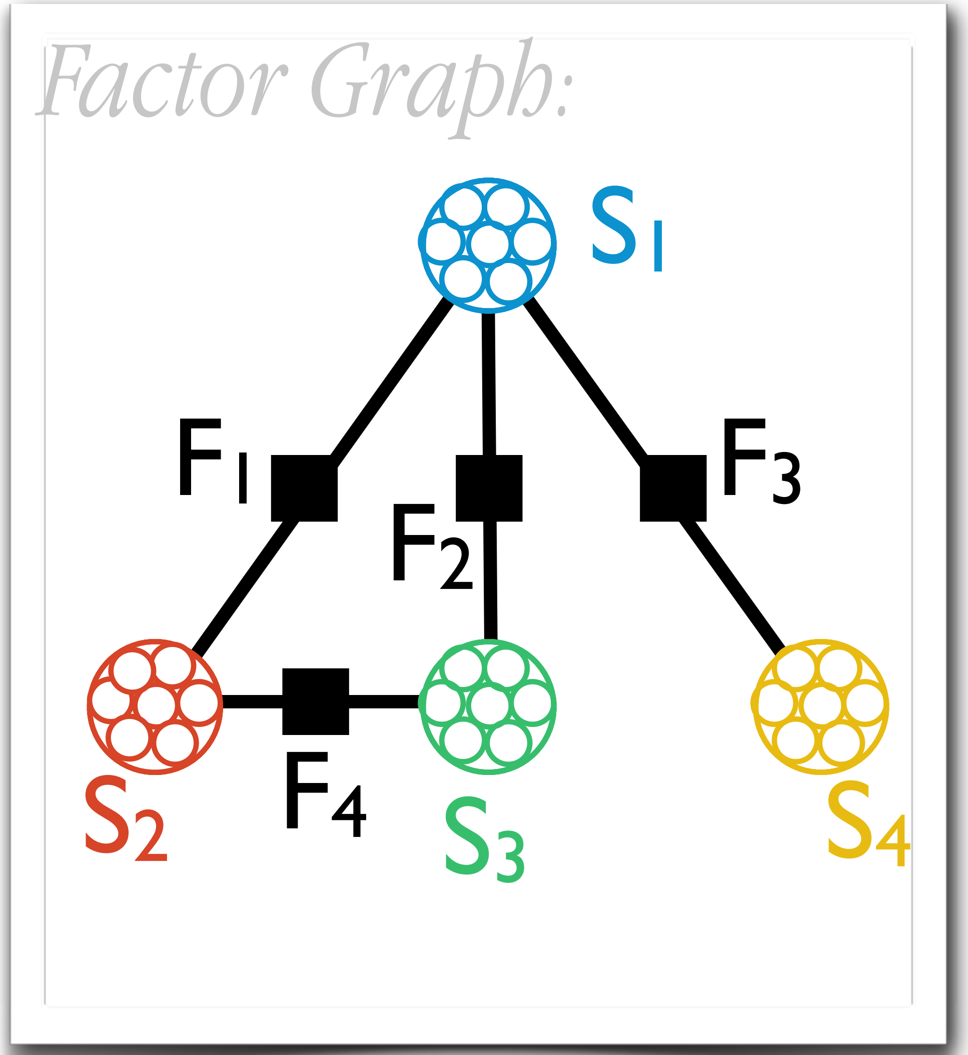
$$\Pr(s_1, s_2, s_3) = 1/Z \times F_1(s_1, s_2) \times F_2(s_1, s_3)$$

Model. *Factor graph examples*



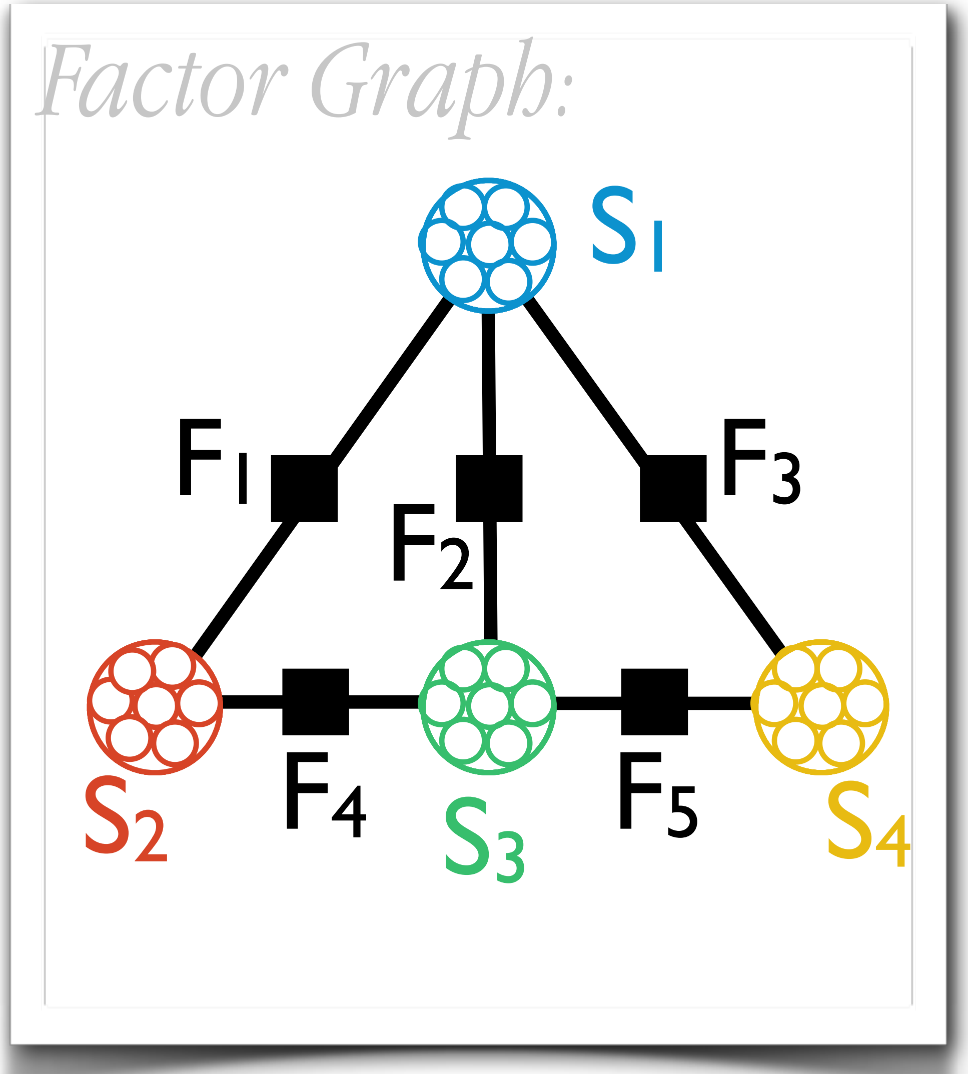
$$\Pr(s_1, s_2, s_3, s_4) = 1/Z$$
$$\times F_1(s_1, s_2)$$
$$\times F_2(s_1, s_3)$$
$$\times F_3(s_1, s_4)$$

Model. *Factor graph examples*



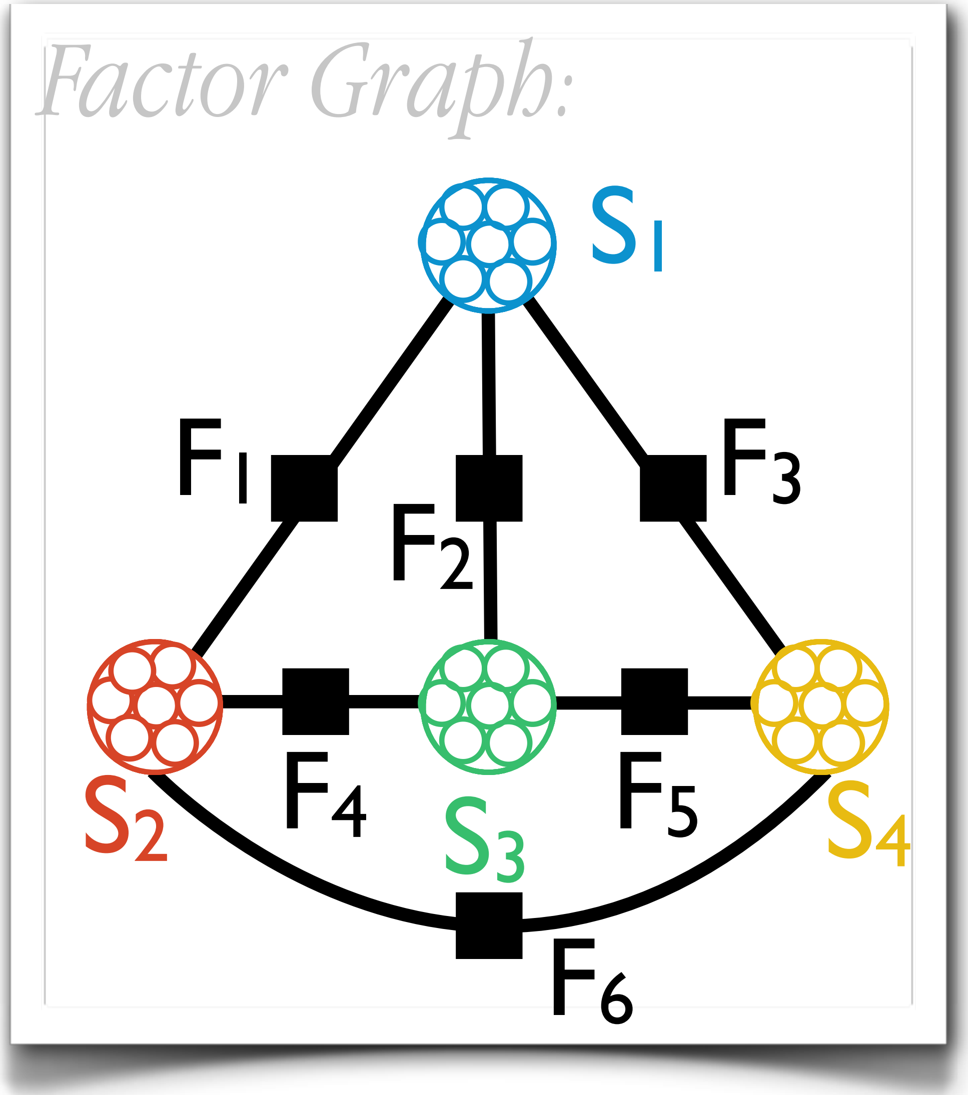
$$\Pr(s_1, s_2, s_3, s_4) = 1/Z \times F_1(s_1, s_2) \times F_2(s_1, s_3) \times F_3(s_1, s_4) \times F_4(s_2, s_3)$$

Model. *Factor graph examples*



$$
\begin{aligned}
\Pr(s_1, s_2, s_3, s_4) = \; & 1/Z \\
& \times F_1(s_1, s_2) \\
& \times F_2(s_1, s_3) \\
& \times F_3(s_1, s_4) \\
& \times F_4(s_2, s_3) \\
& \times F_5(s_3, s_4)
\end{aligned}
$$

Model. *Factor graph examples*



$$\Pr(s_1, s_2, s_3, s_4) = 1/Z$$
$$\times F_1(s_1, s_2)$$
$$\times F_2(s_1, s_3)$$
$$\times F_3(s_1, s_4)$$
$$\times F_4(s_2, s_3)$$
$$\times F_5(s_3, s_4)$$
$$\times F_6(s_2, s_4)$$

Model. *Summary*

- Our model is formally an undirected **graphical model**,
- in which the **variables are string-valued**, and the **factors** (potential functions) **are finite-state transducers.**

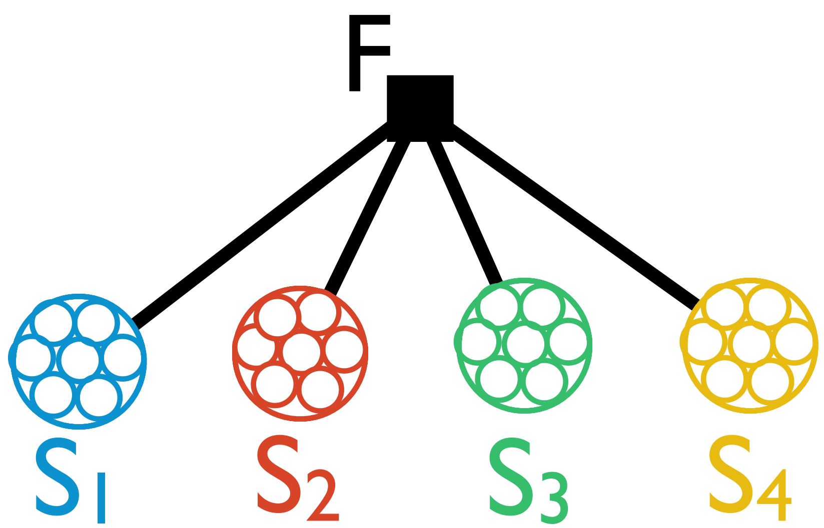
Model. *Less formal description*

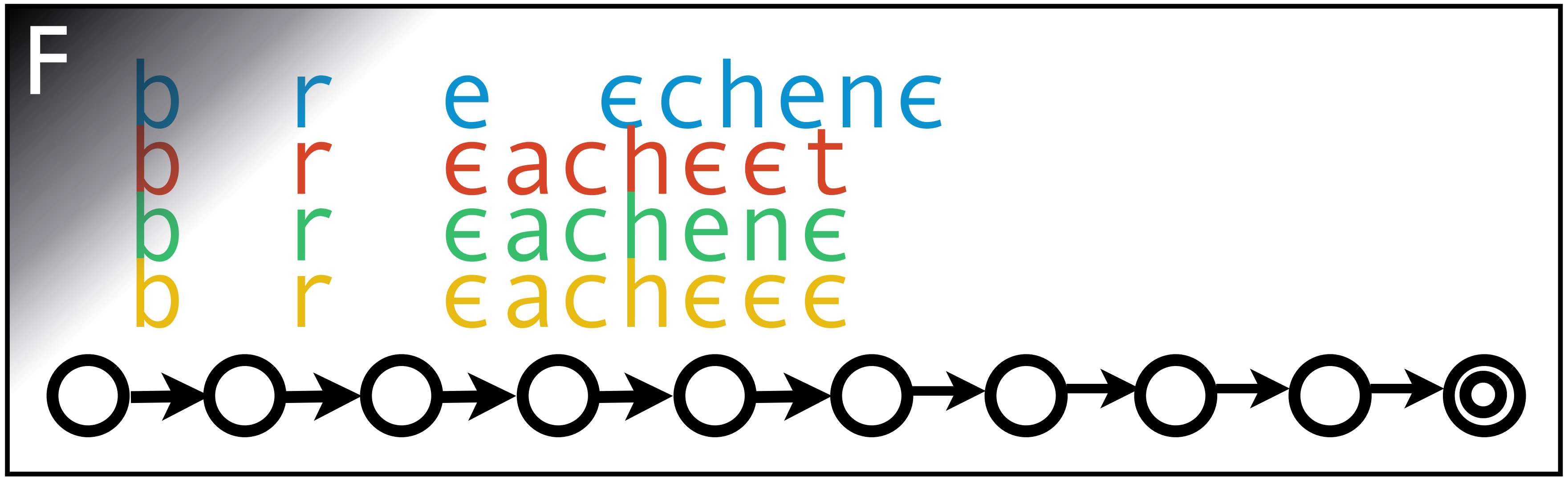
To **model multiple strings** and their various interactions, we

- build many **finite-state transducers**, like the ones we presented last year,
- have each of them look at a different string pair,
- **plug them together** into a big network,
- and **coordinate them** to predict all strings jointly.

Model. *Comparison with k-tape FSM*

- Model k strings with a k-tape finite-state machine?

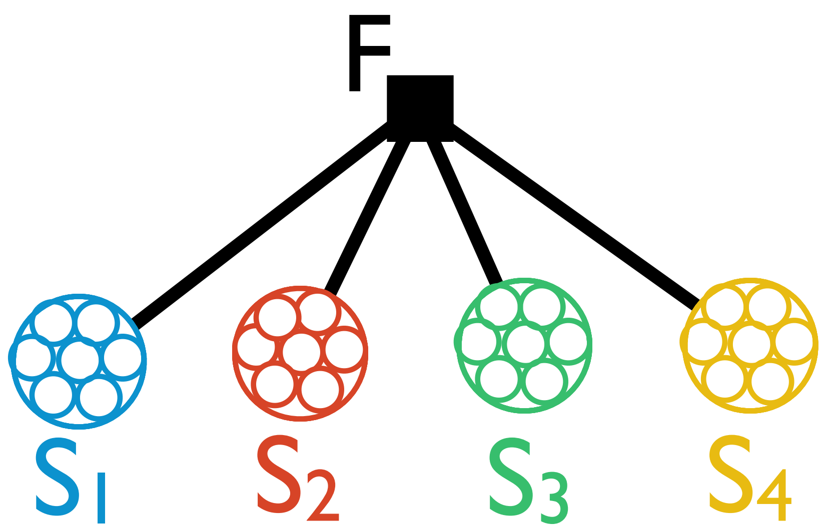


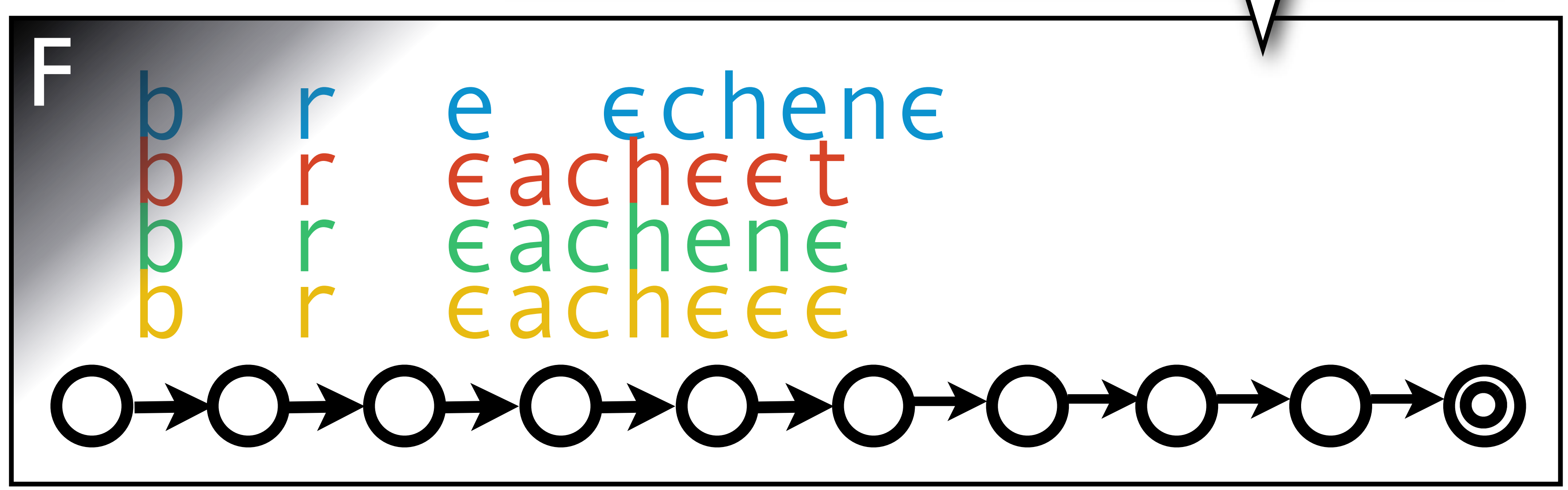


Model. *Comparison with k-tape FSM*

- Model k strings with a k-tape finite-state machine?
- $>26^k$ arcs, intractable!

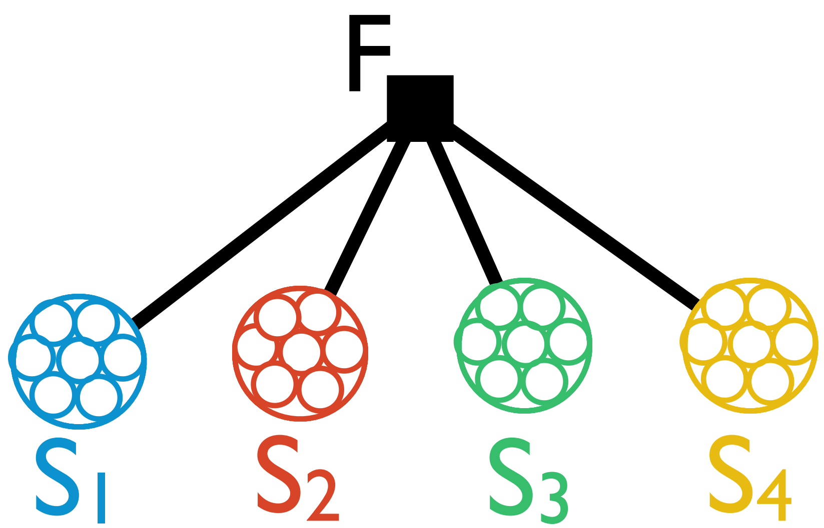
Multiple-sequence alignment

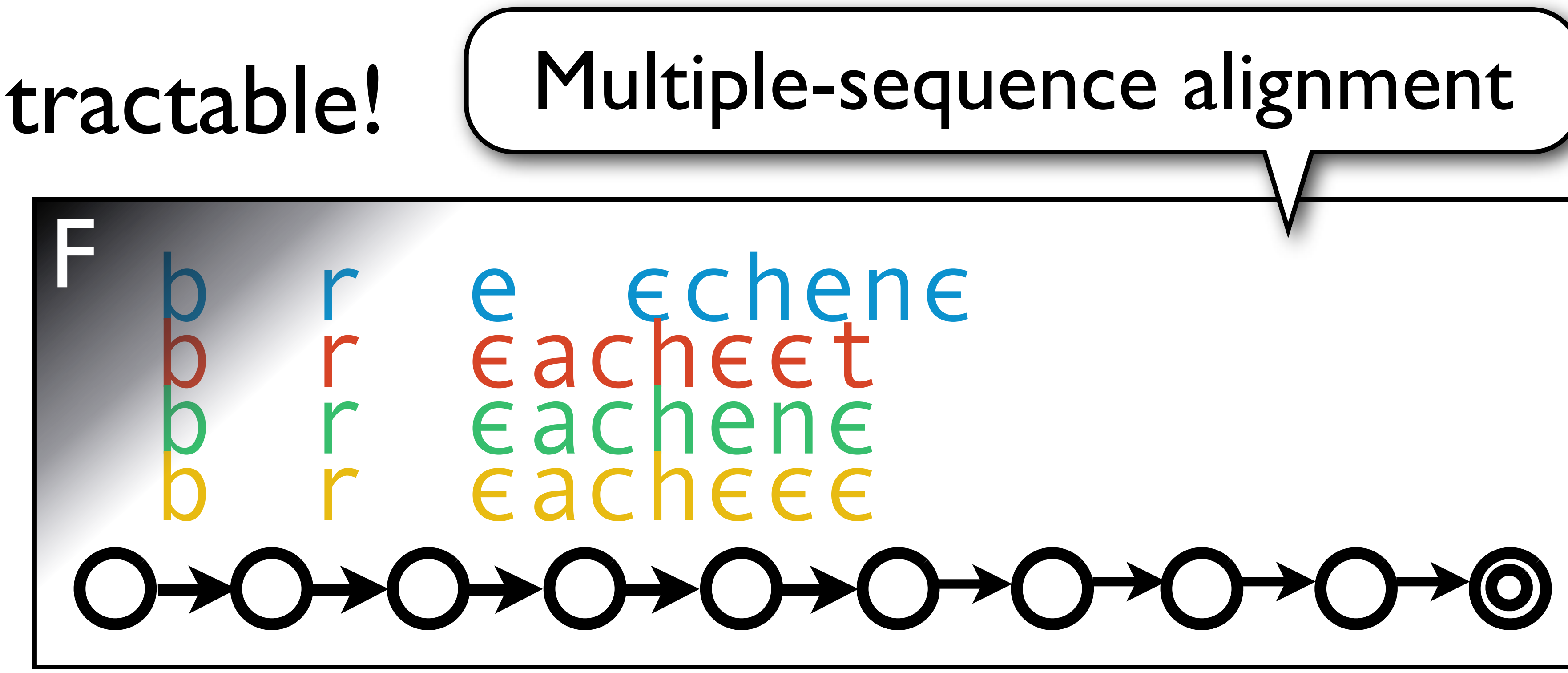




Model. *Comparison with k-tape FSM*

- Model k strings with a k-tape finite-state machine?
- $>26^k$ arcs, intractable!



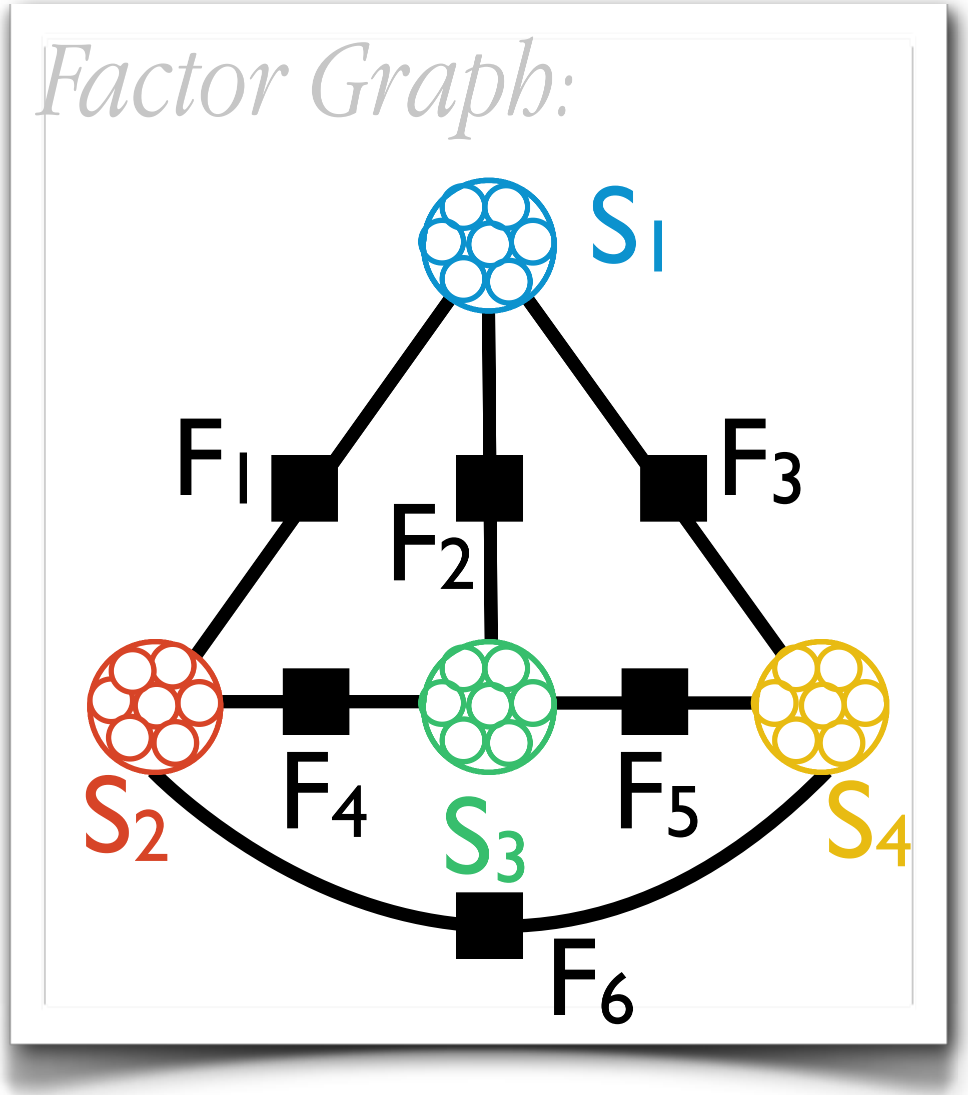


- Factored model more powerful:
  - Encode swaps and other useful models ☺
  - Encode undecidable models ☹

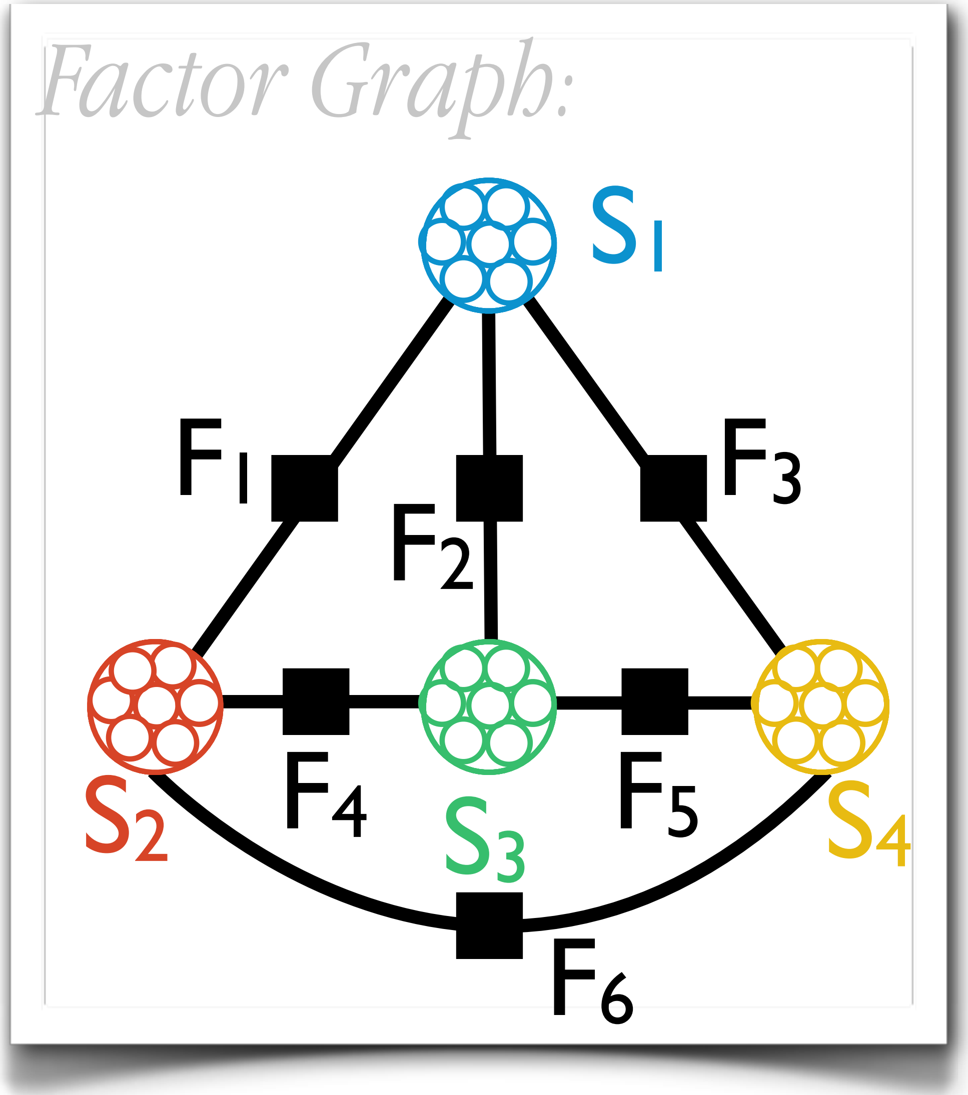
Overview

- Motivation
- Model
- **Inference & Approximations**
- Experiments
- Conclusions

Inference. *Overview*

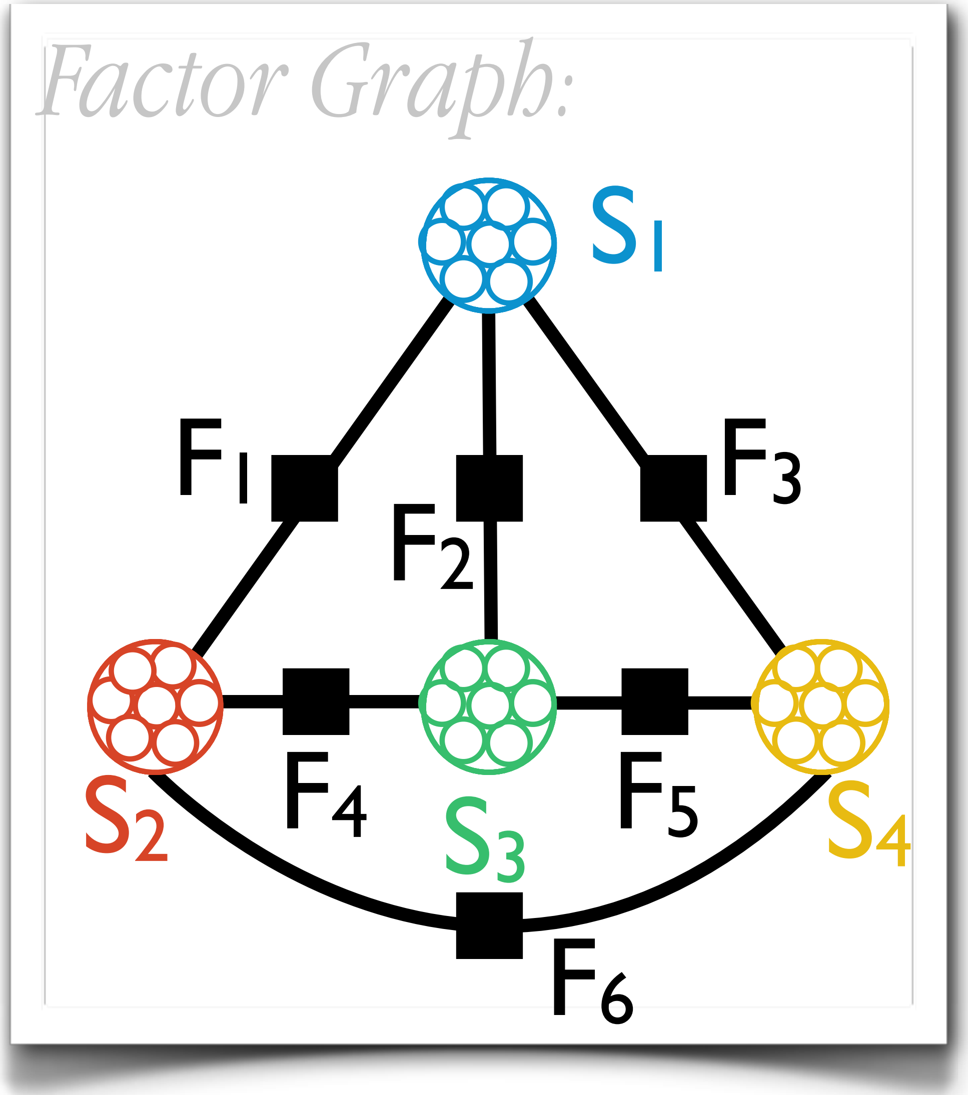


Inference. *Overview*



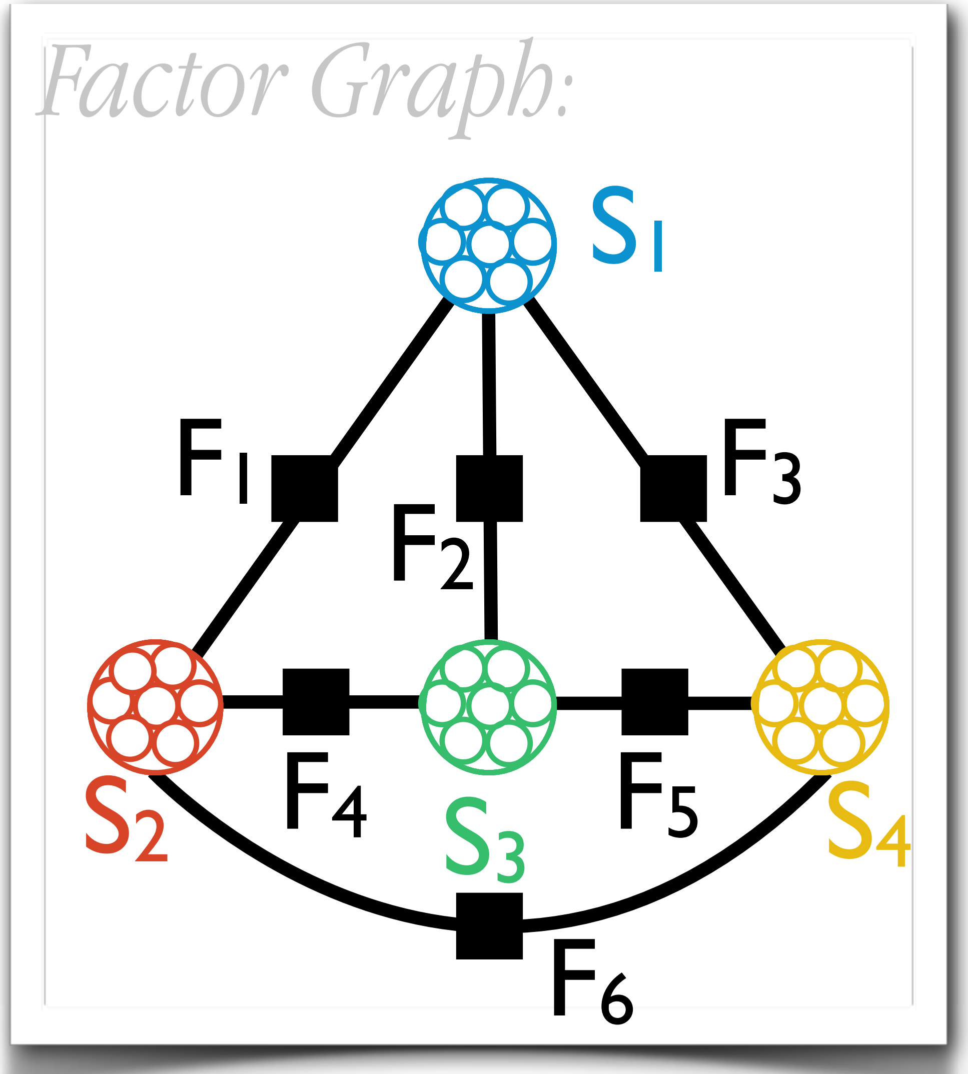
- We run **Belief Propagation (BP)**

Inference. *Overview*



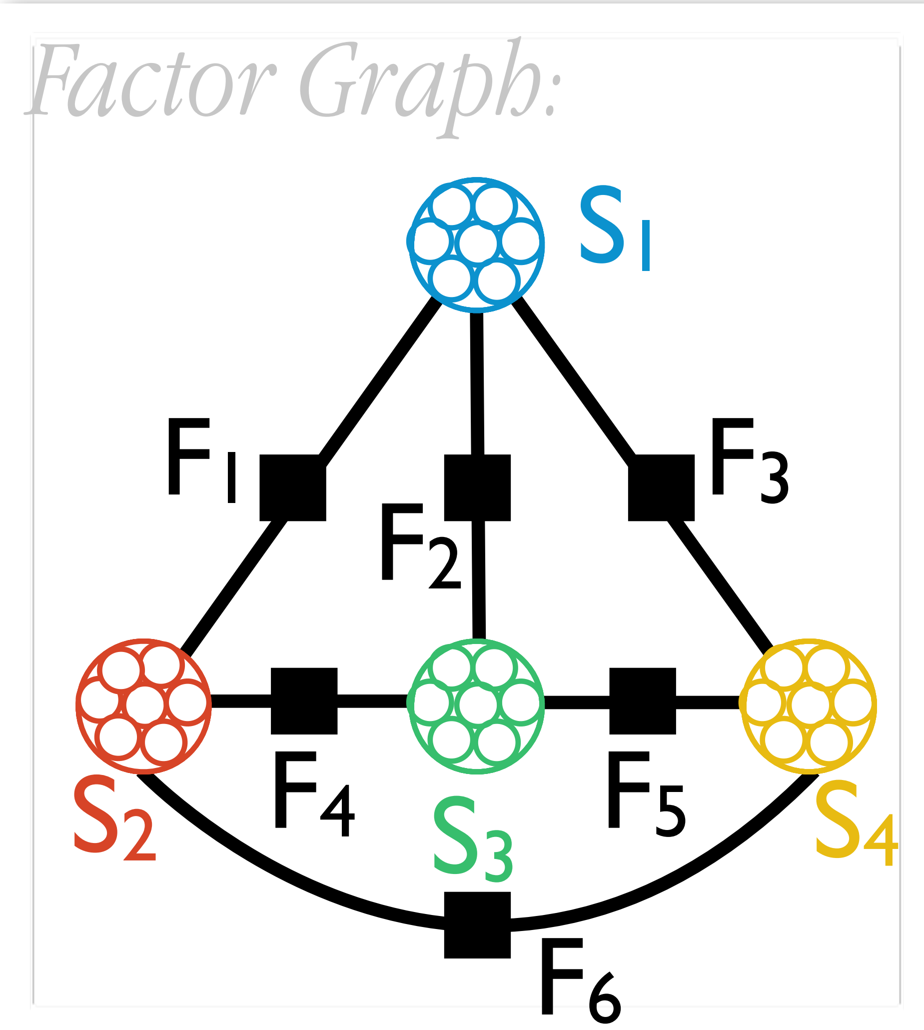
- We run **Belief Propagation (BP)**
- BP is a message-passing algorithm, a **generalization of forward-backward**.

Inference. *Overview*



- We run **Belief Propagation (BP)**
- BP is a message-passing algorithm, a **generalization of forward-backward**.
- BP computes marginals

Inference. *Overview*

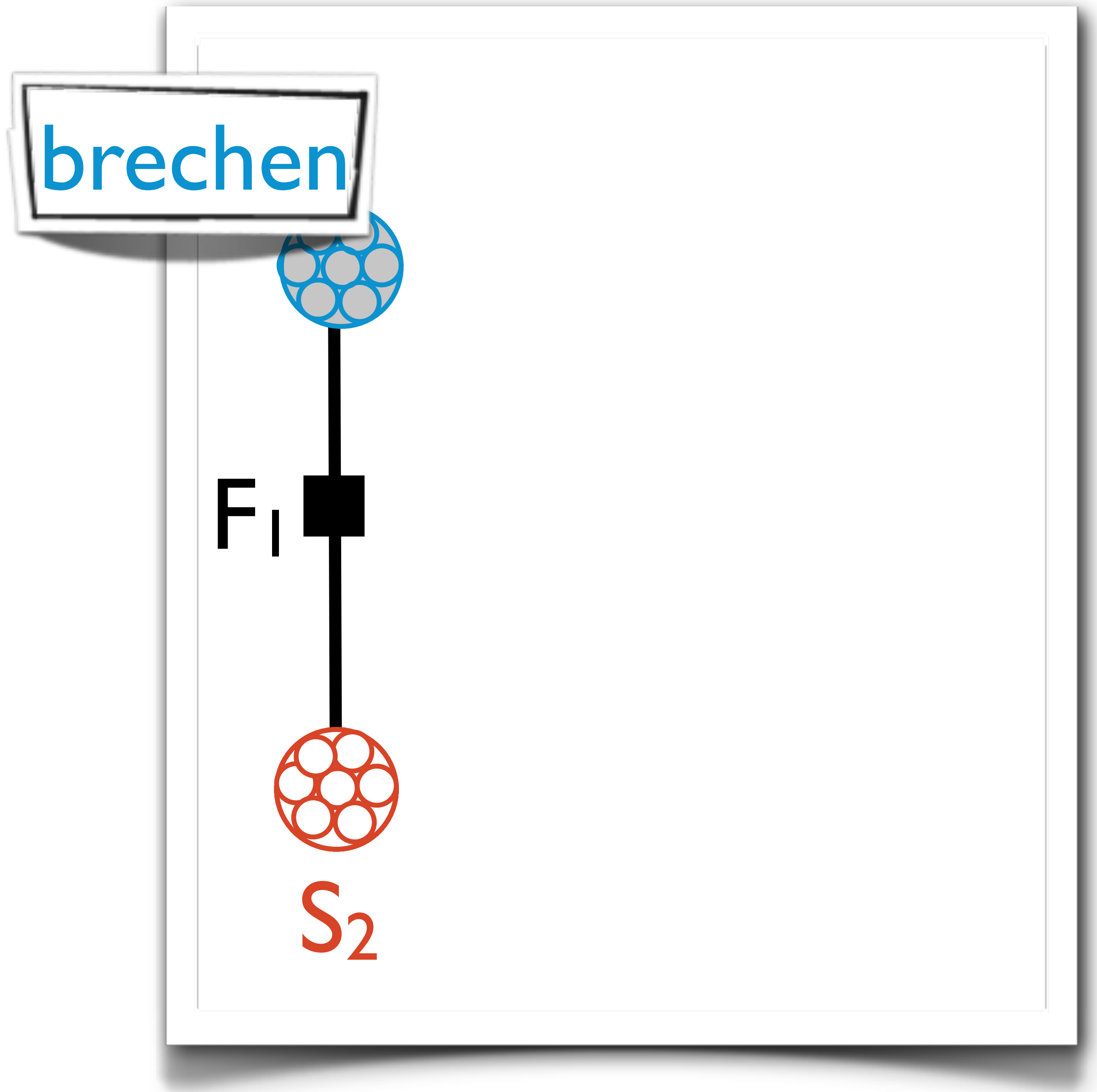


- We run **Belief Propagation (BP)**
- BP is a message-passing algorithm, a **generalization of forward-backward**.
- BP computes marginals

In our version of BP, **all messages and beliefs are finite-state machines**, which is novel.

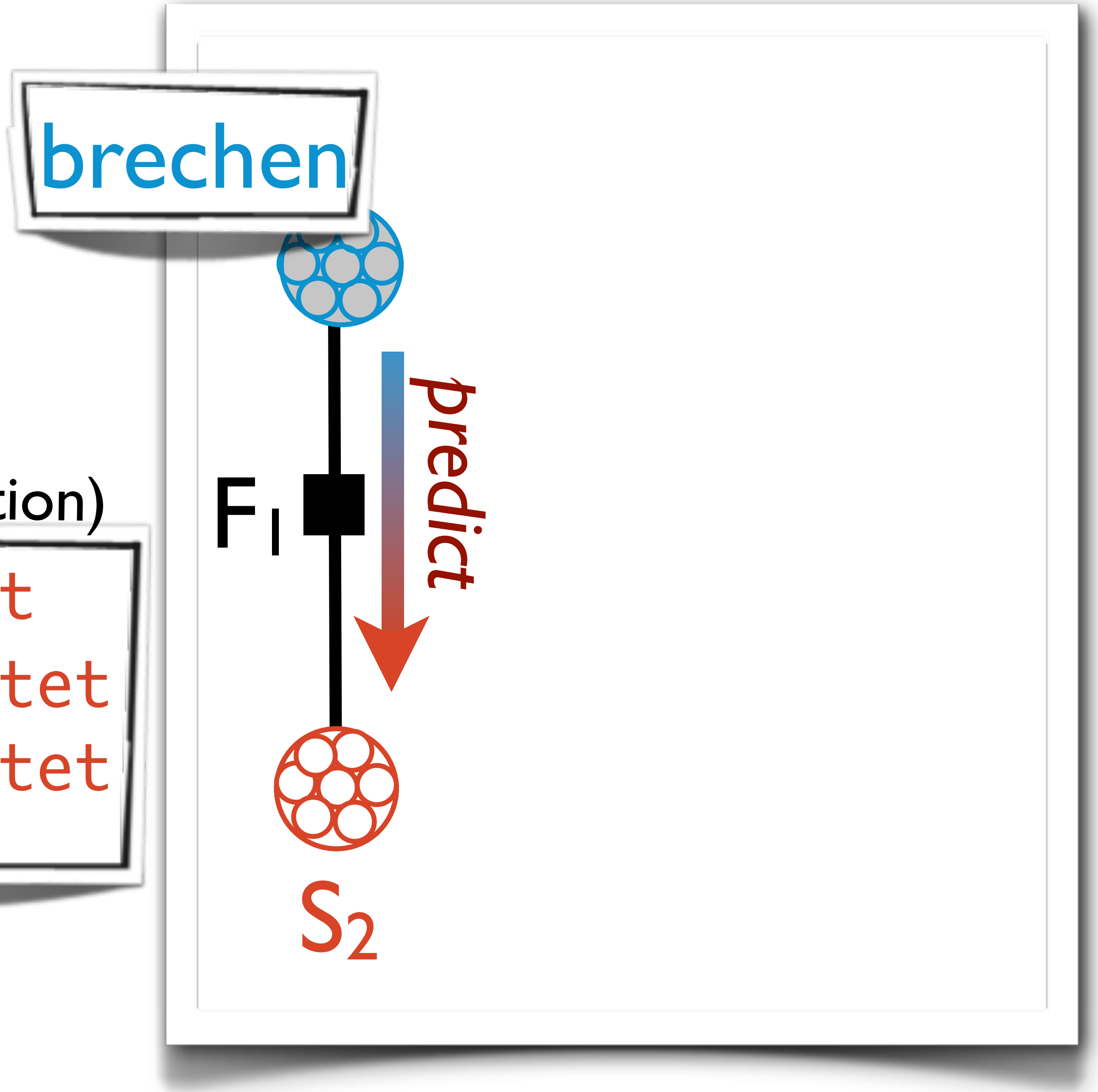
Inference. *Multiple strings*

Example:



Inference. *Multiple strings*

Example:



(whole distribution)

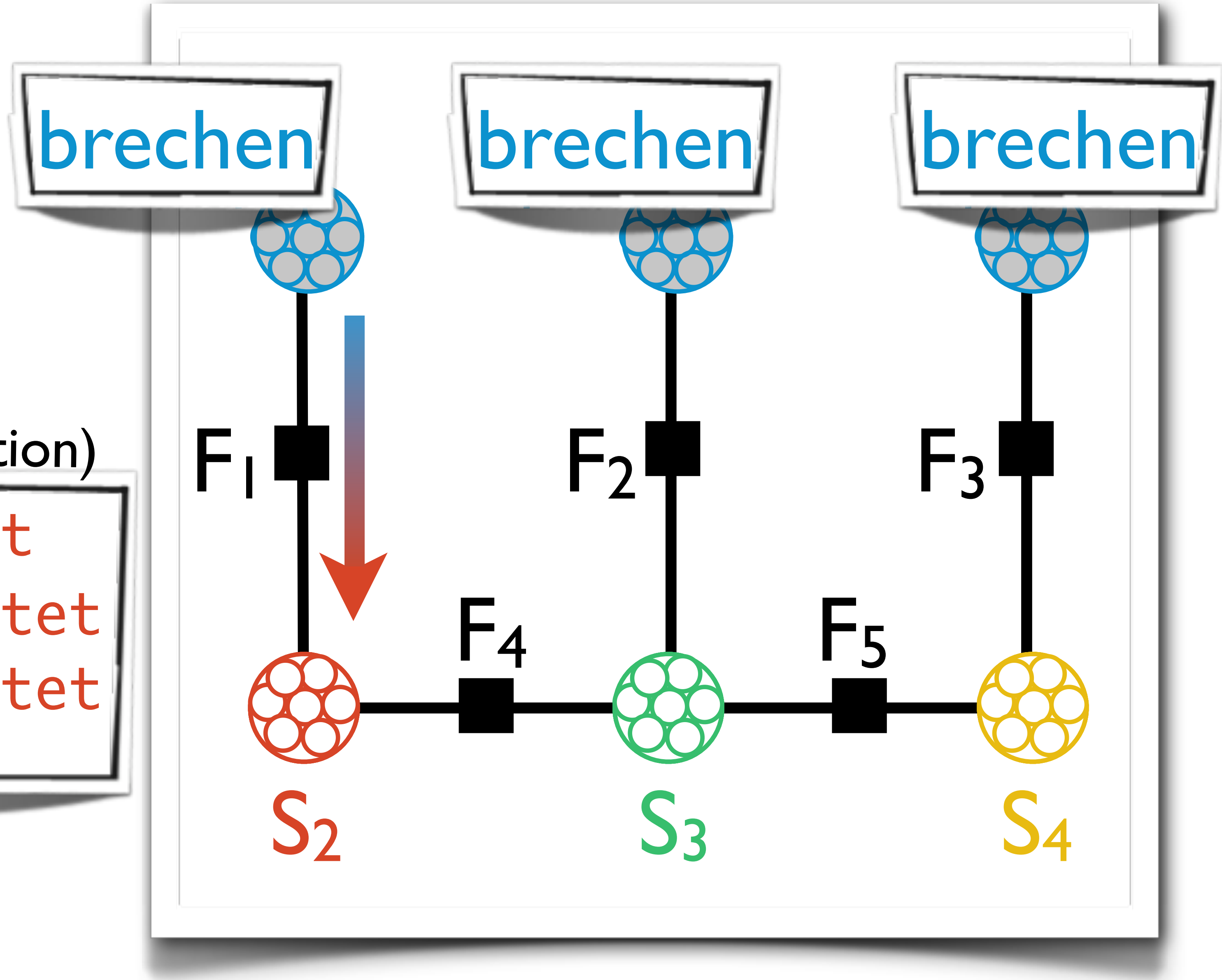
```
0.20 bracht
0.13 brechtet
0.08 brachtet
...
```

Inference. *Multiple strings*

Example:

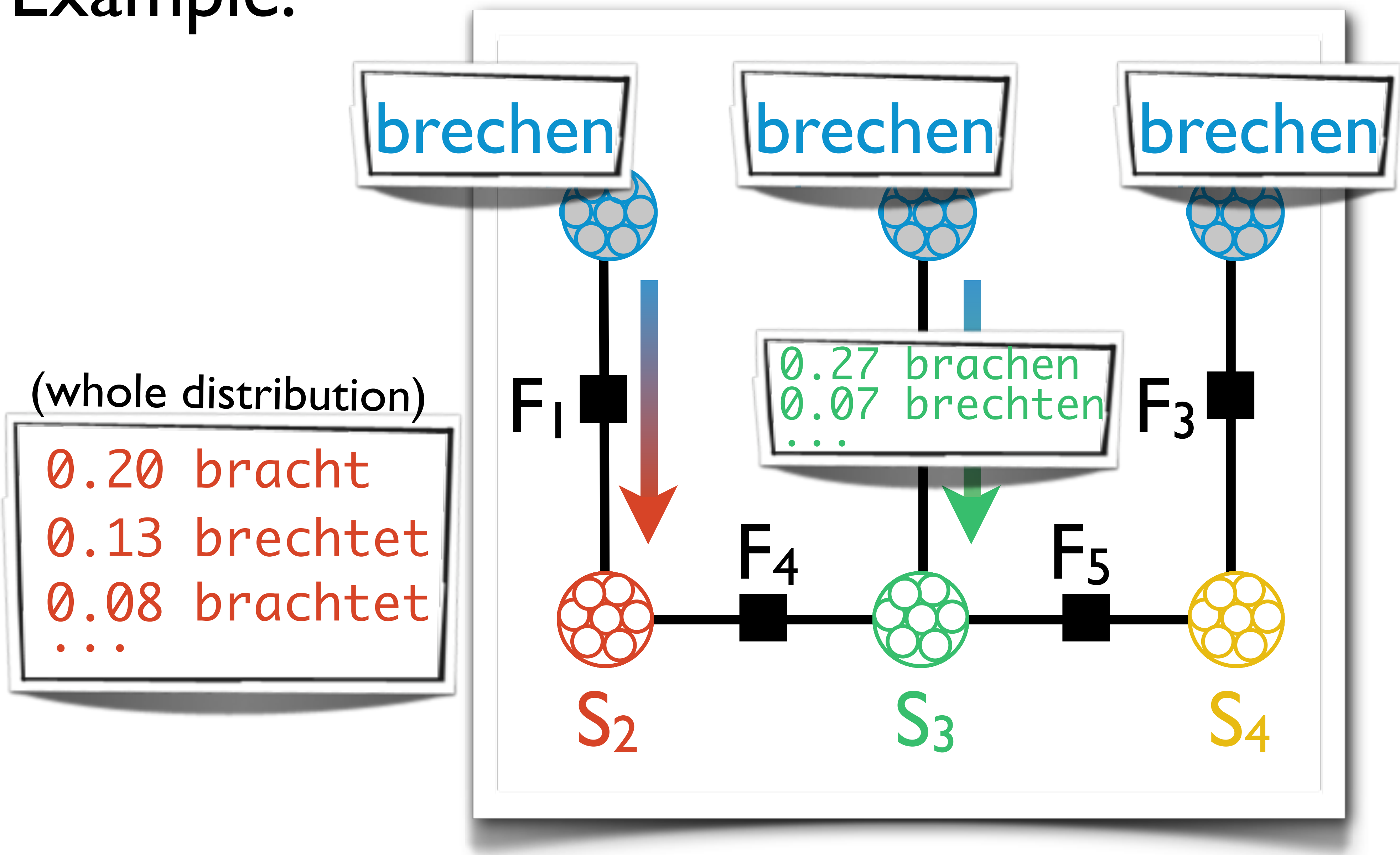
(whole distribution)

```
0.20 bracht
0.13 brechtet
0.08 brachtet
...
```



Inference. *Multiple strings*

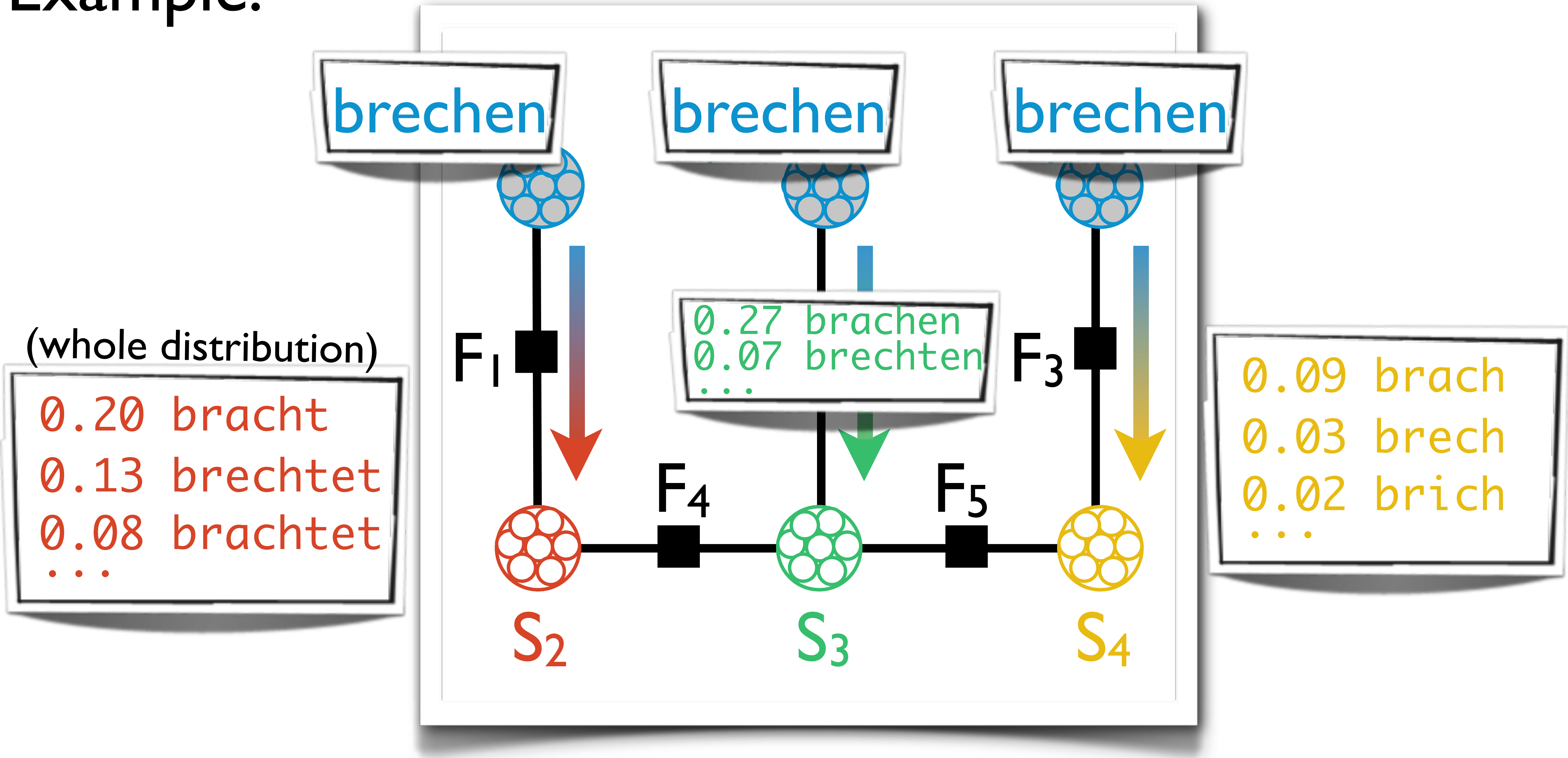
Example:



Inference. *Multiple strings*

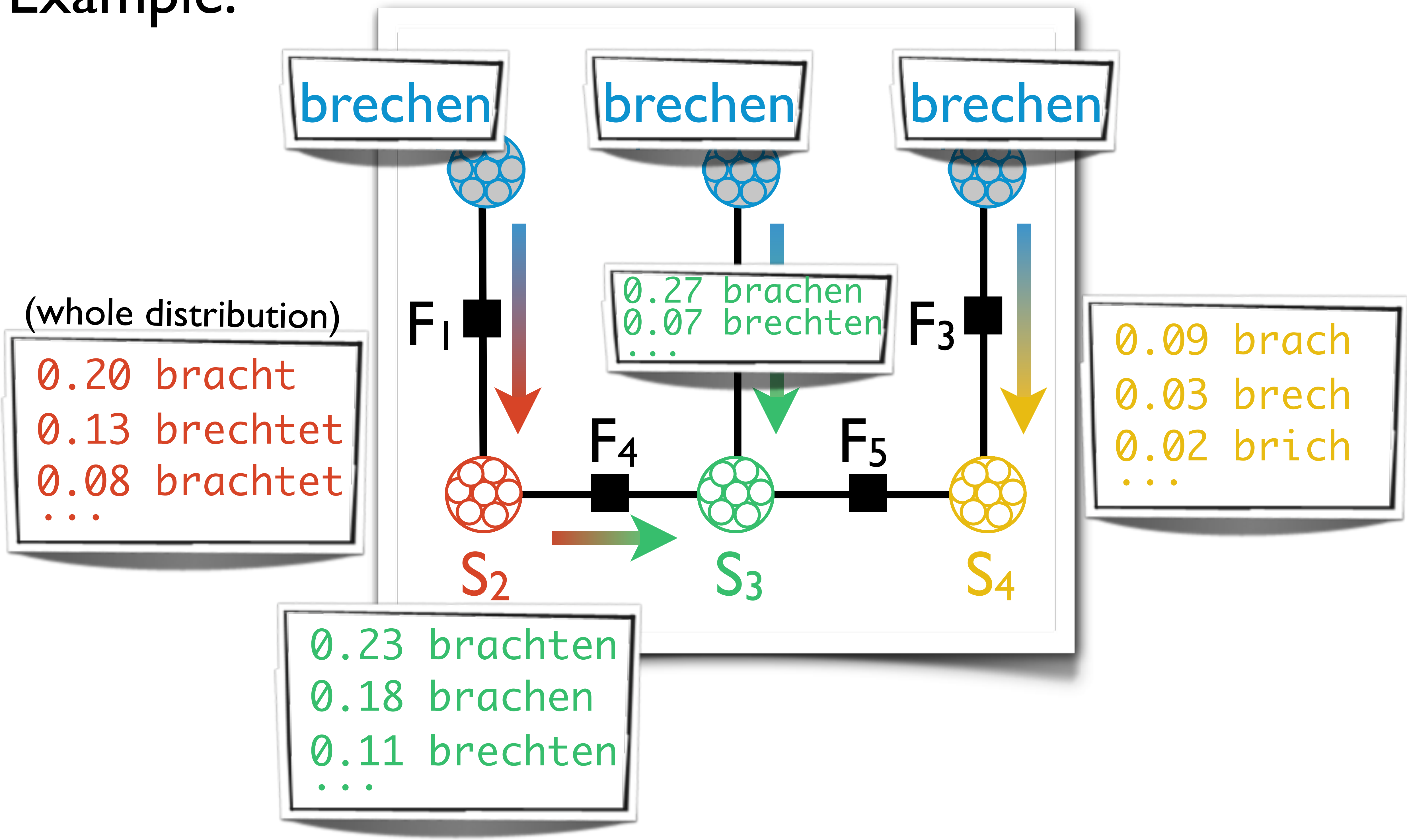
Example:

(whole distribution)



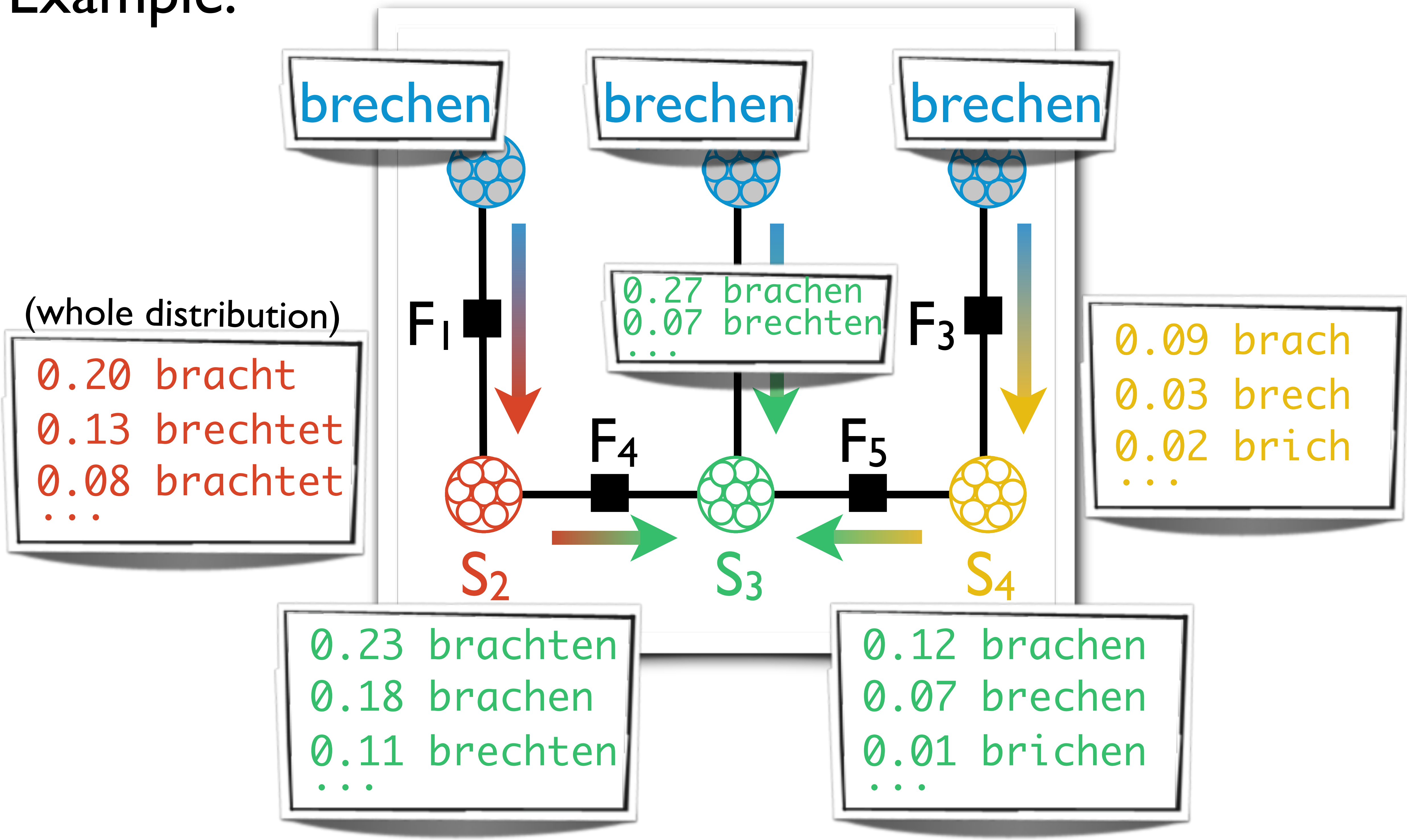
Inference. *Multiple strings*

Example:



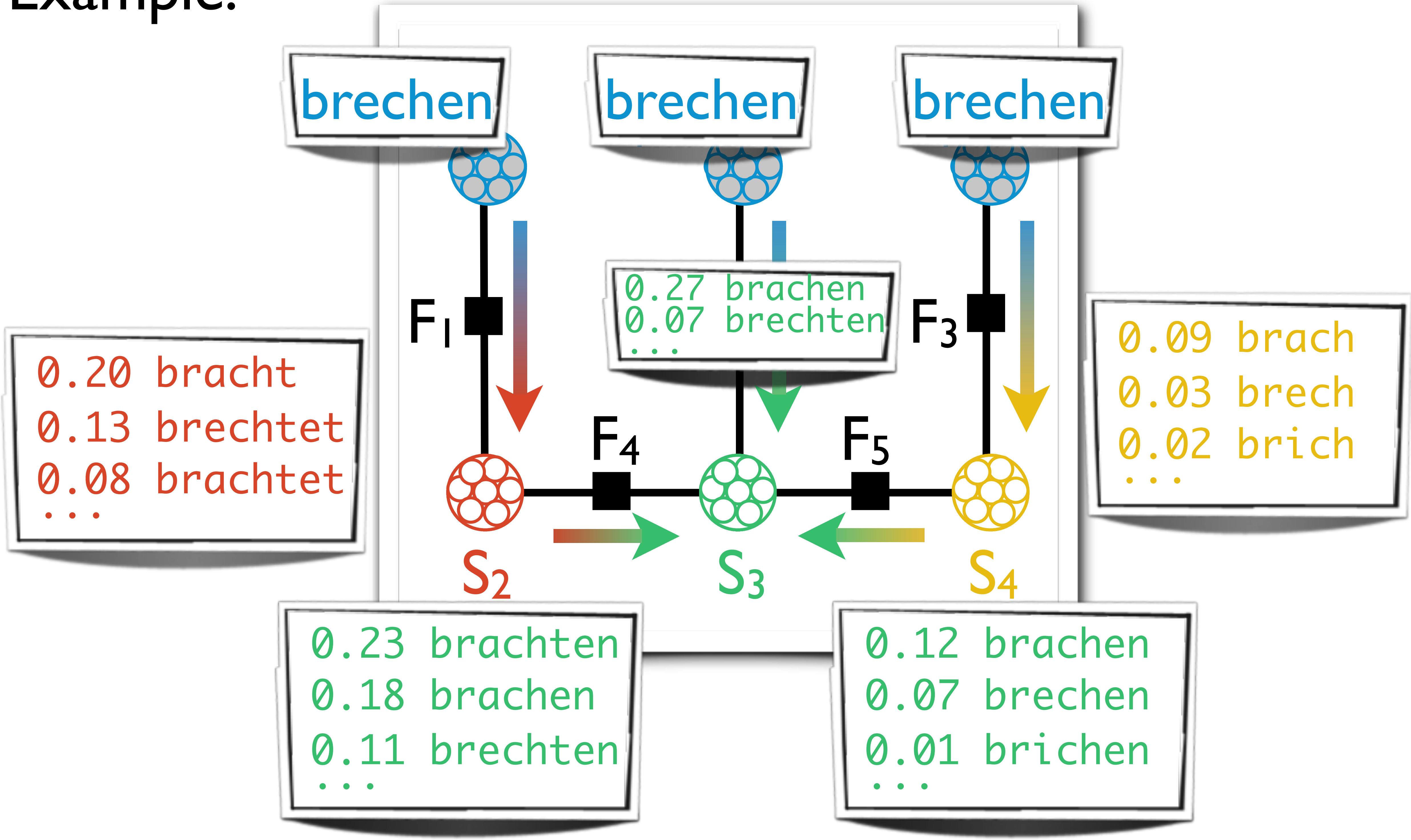
Inference. *Multiple strings*

Example:



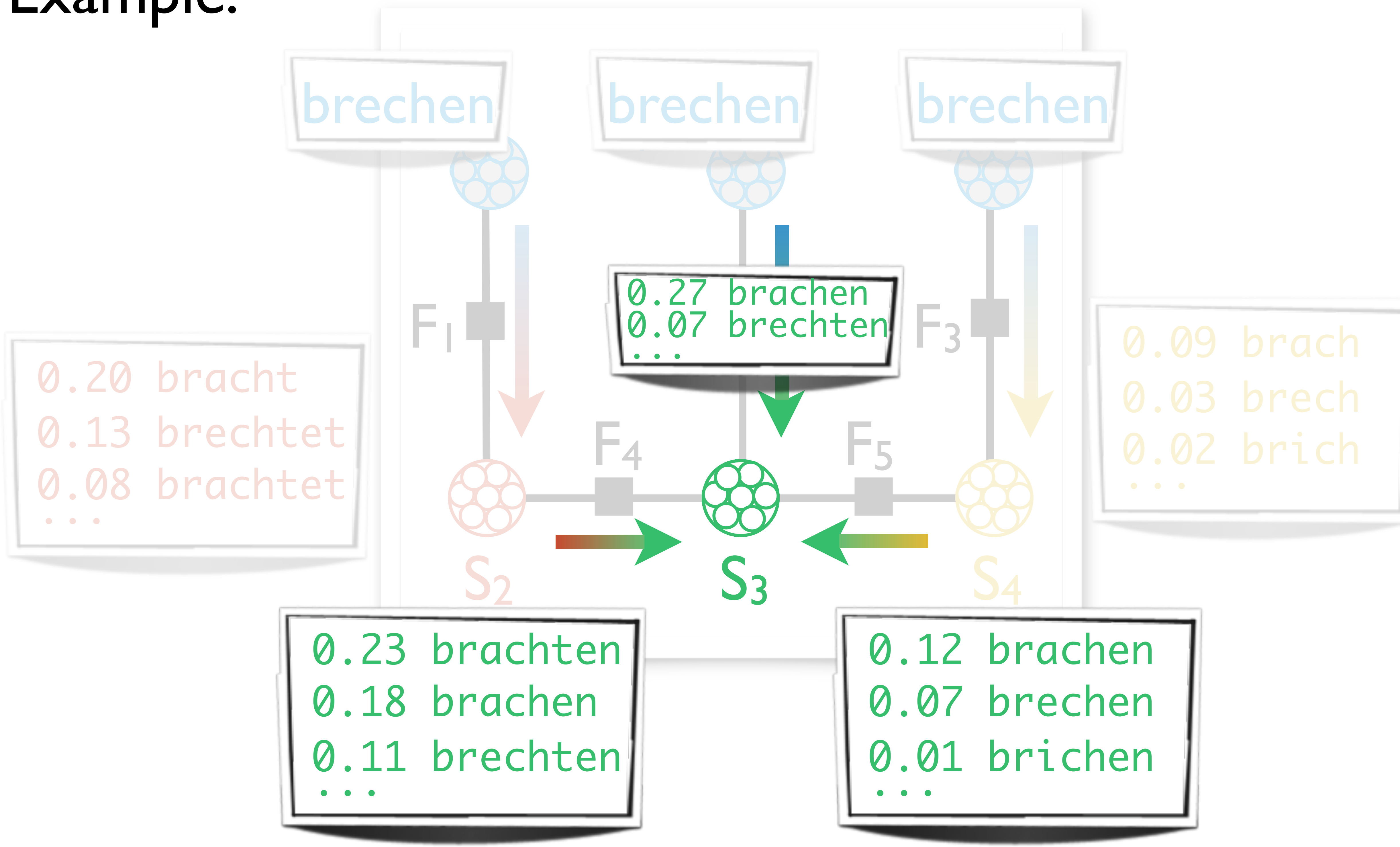
Inference. *Multiple strings*

Example:



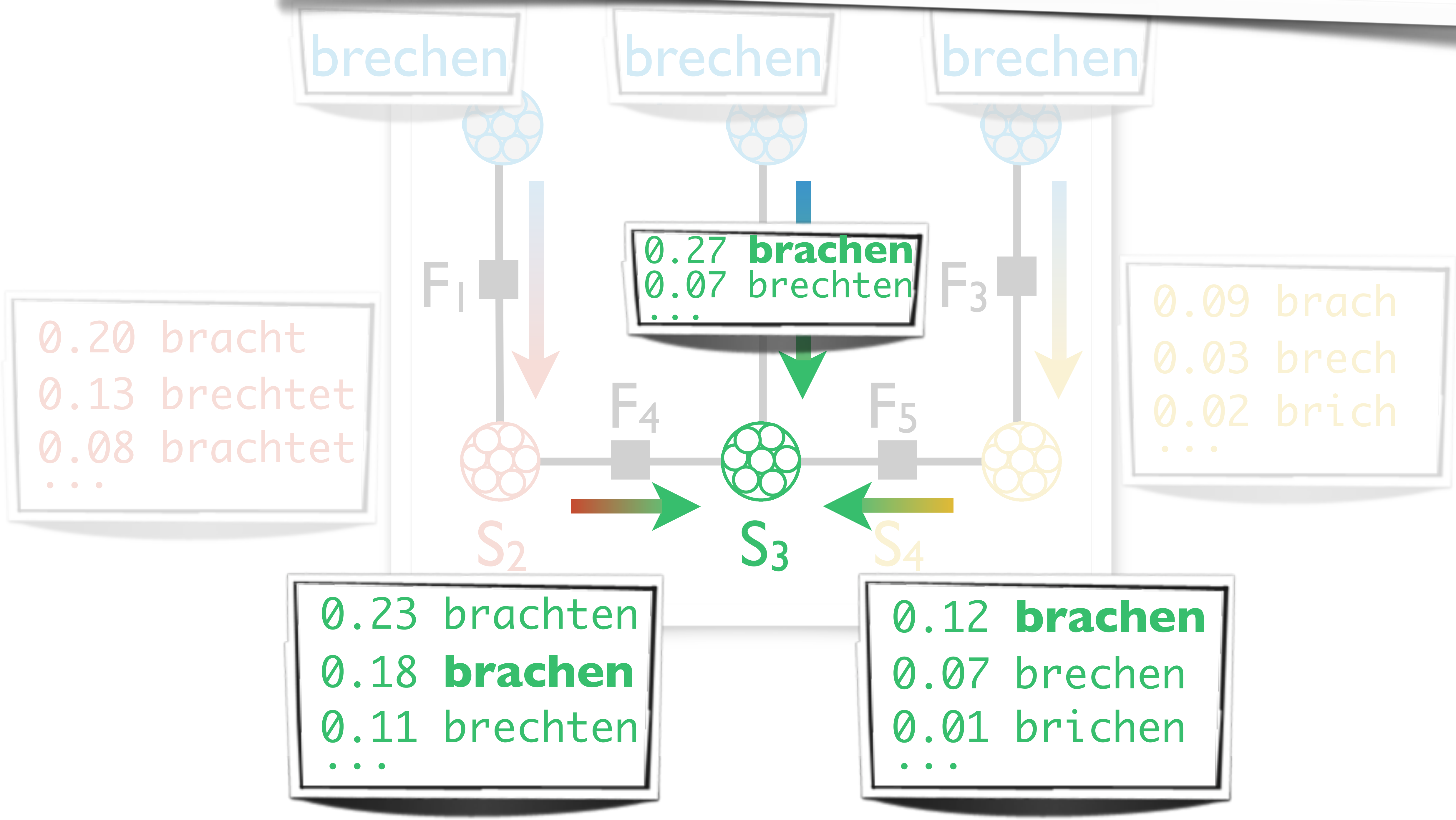
Inference. *Multiple strings*

Example:



Inference. *Multiple strings*

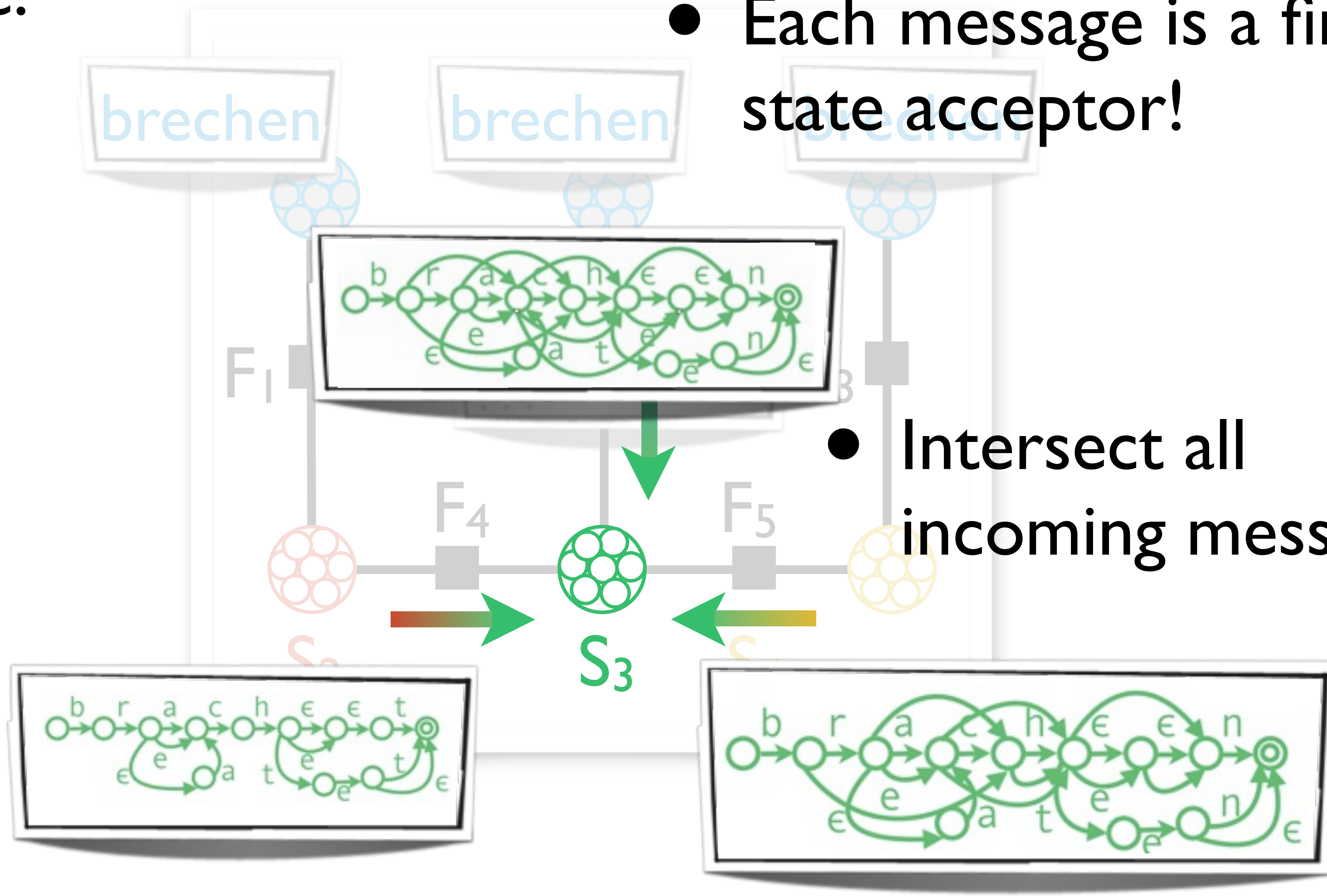
Example: Decoding output for S_3 (consensus): **brachen**



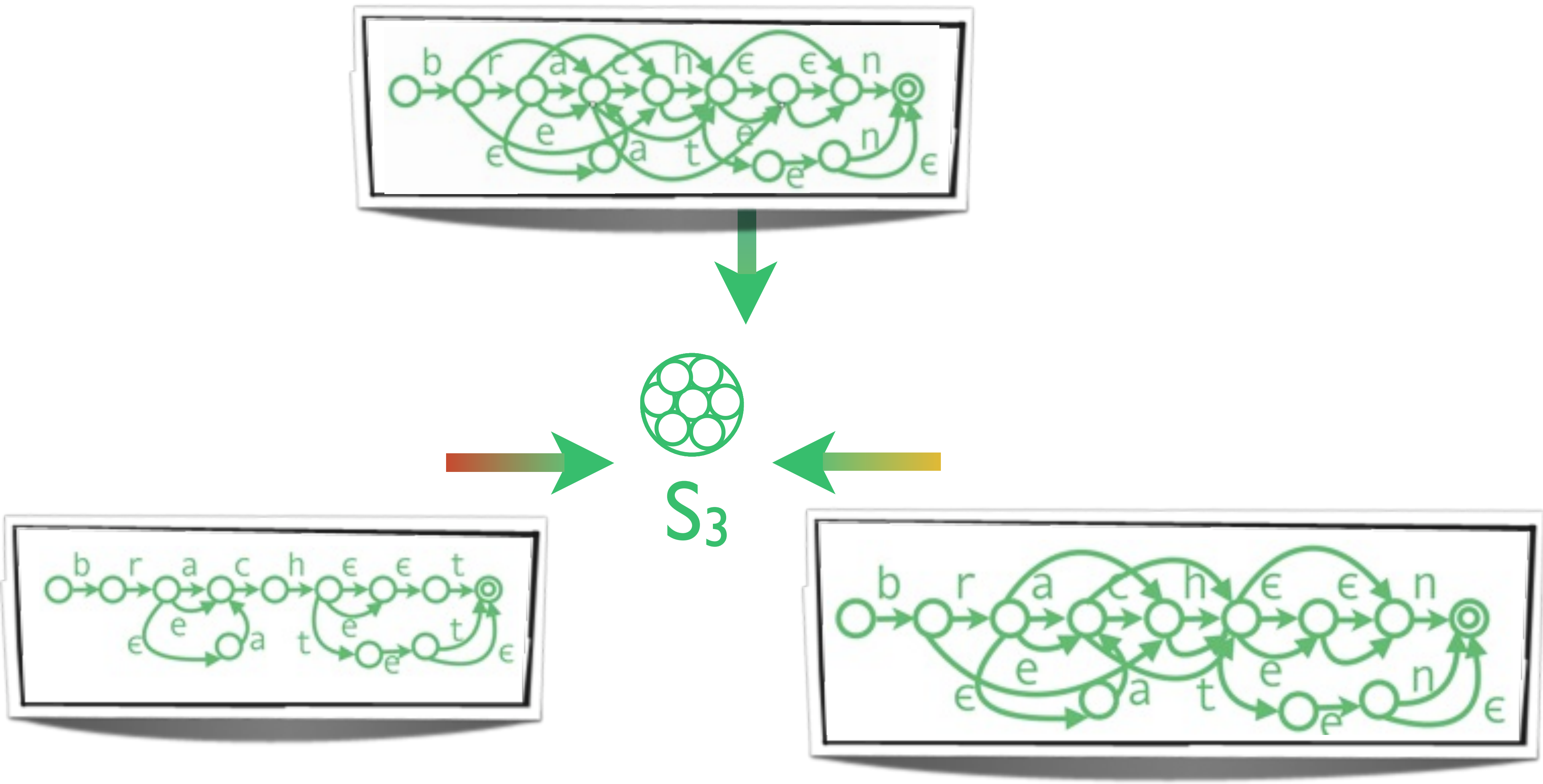
Inference. *Multiple strings*

Example:

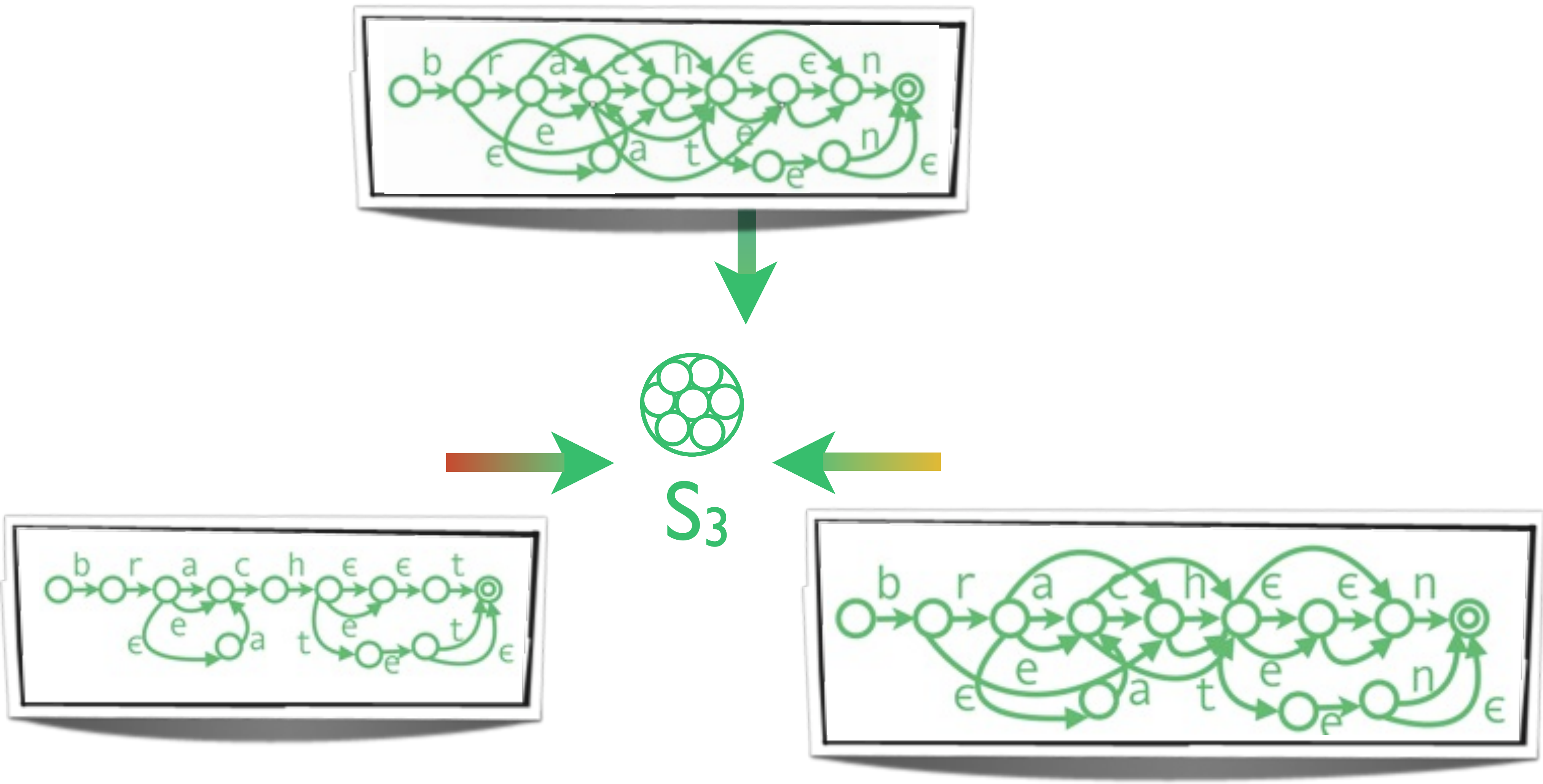
- Each message is a finite-state acceptor!
- Intersect all incoming messages



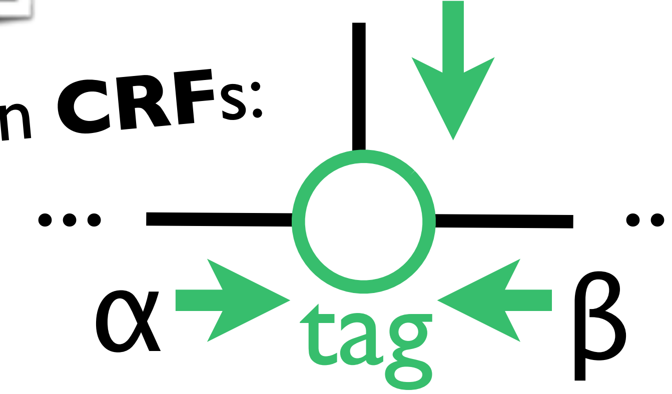
Inference. *Multiple strings*



Inference. *Multiple strings*

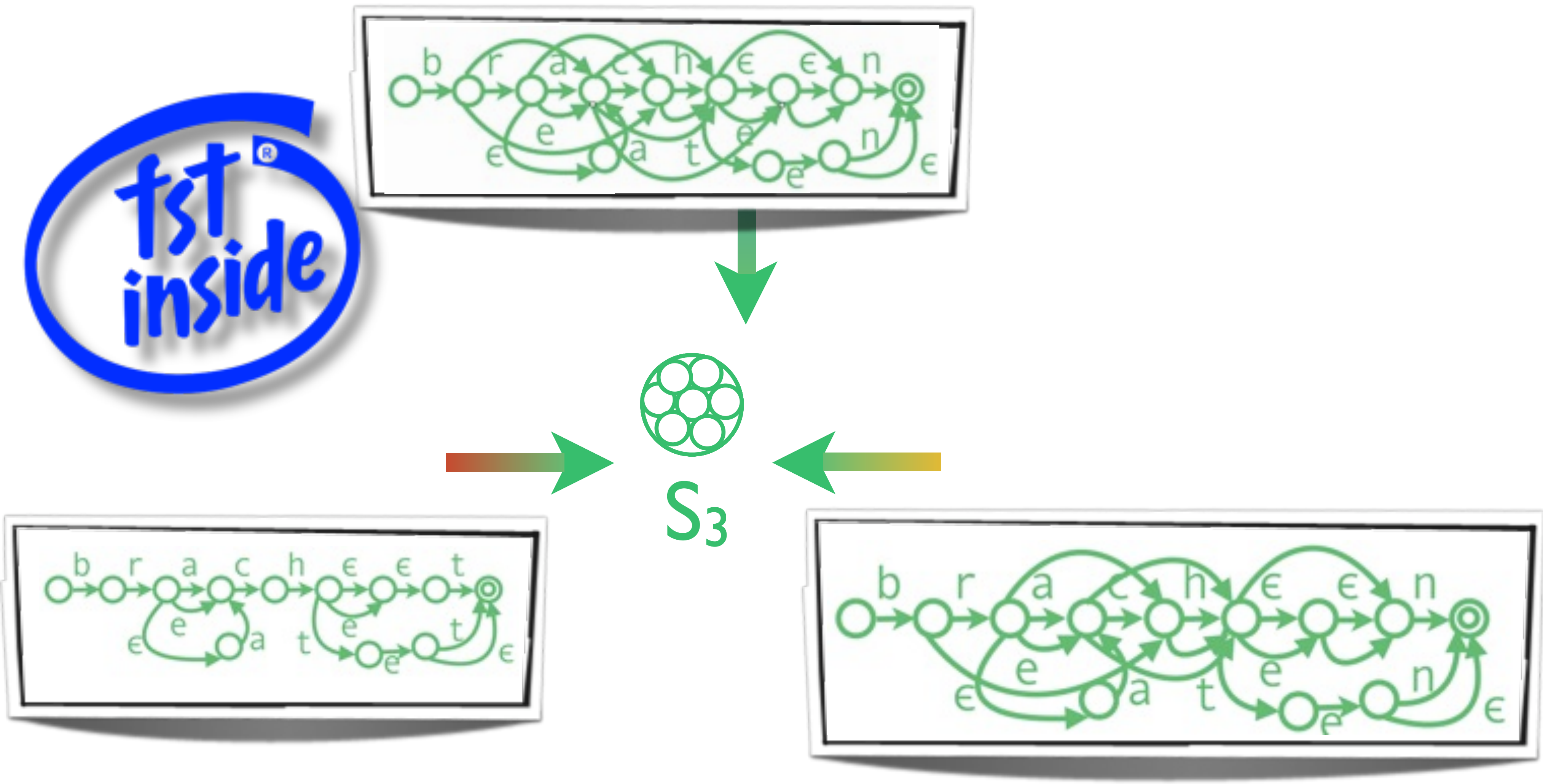


Similar to inference in **CRF**s:

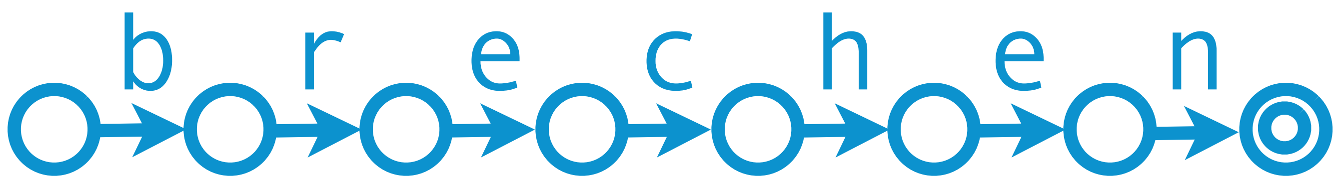


but in CRFs: simple lookup tables,
not finite-state machines!

Inference. *Multiple strings*

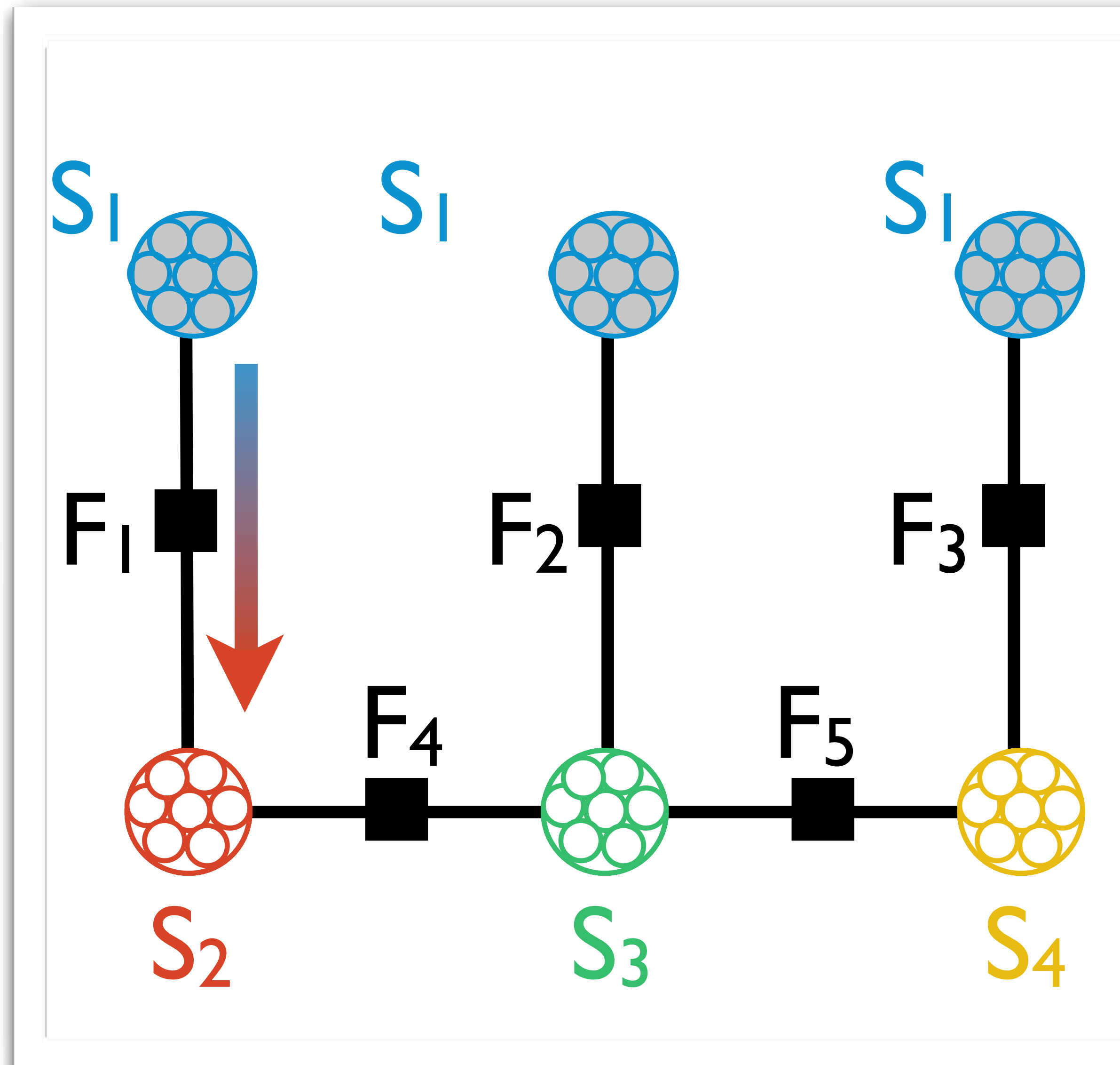


Inference. *Multiple strings*

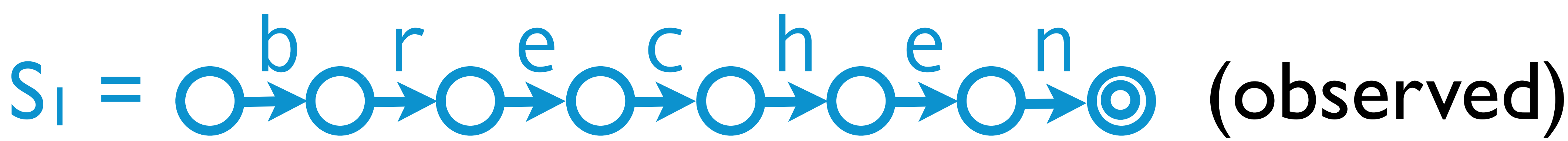
S_1 = 

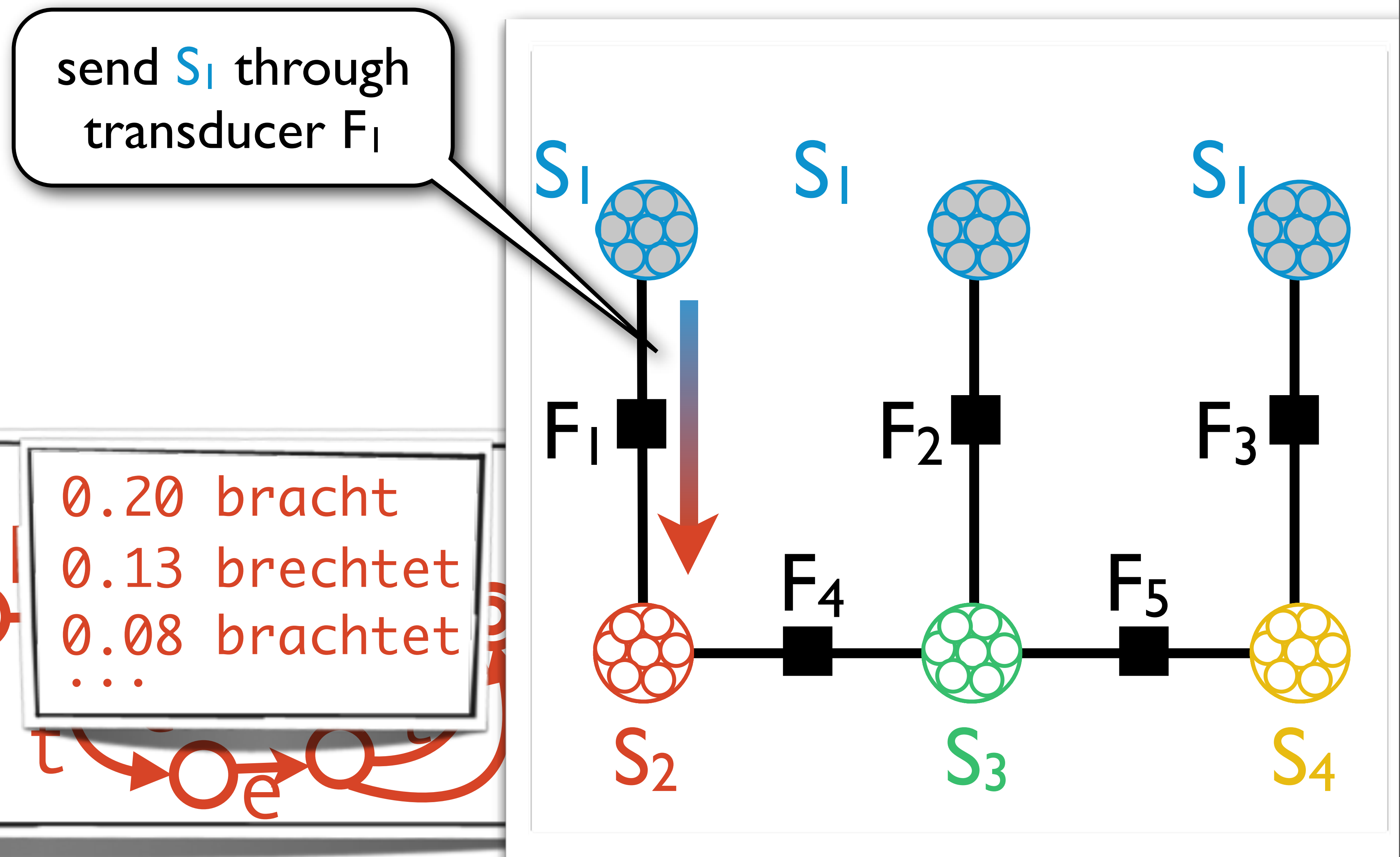
(observed)

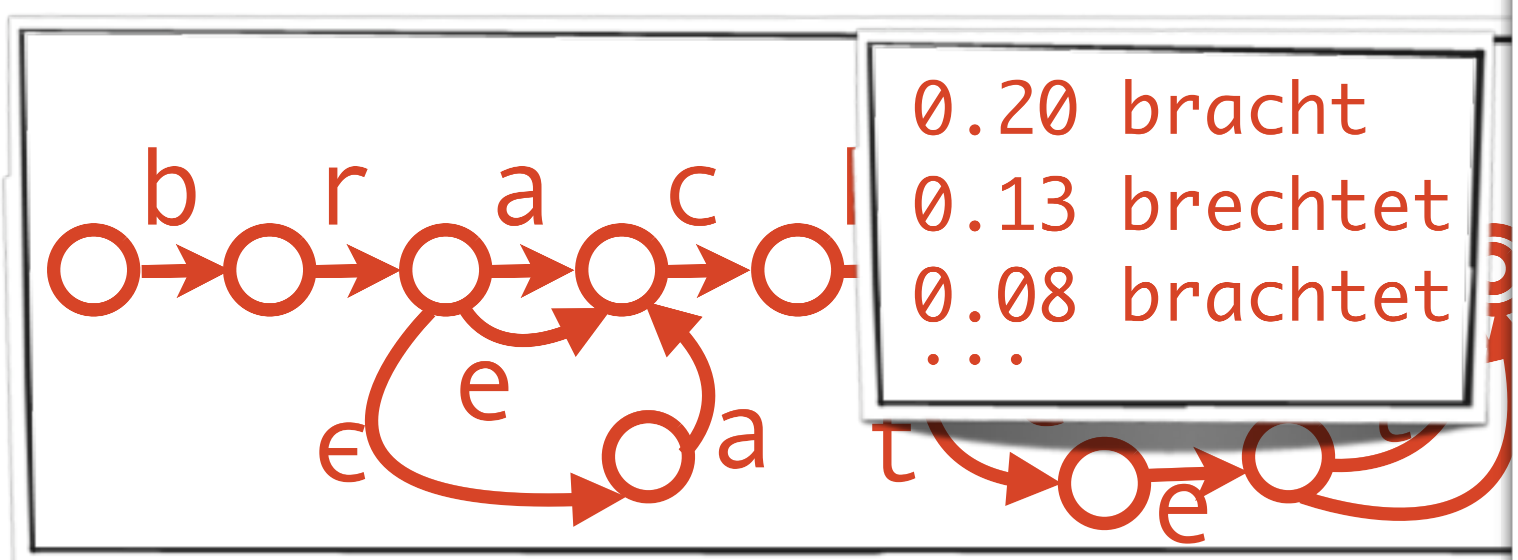
```
0.20 bracht
0.13 brechtet
0.08 brachtet
...
```



Inference. *Multiple strings*

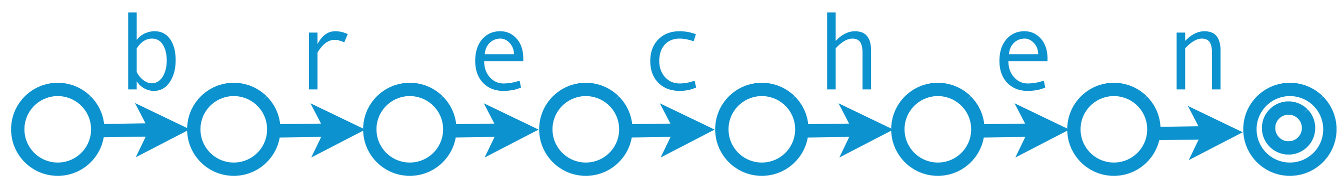






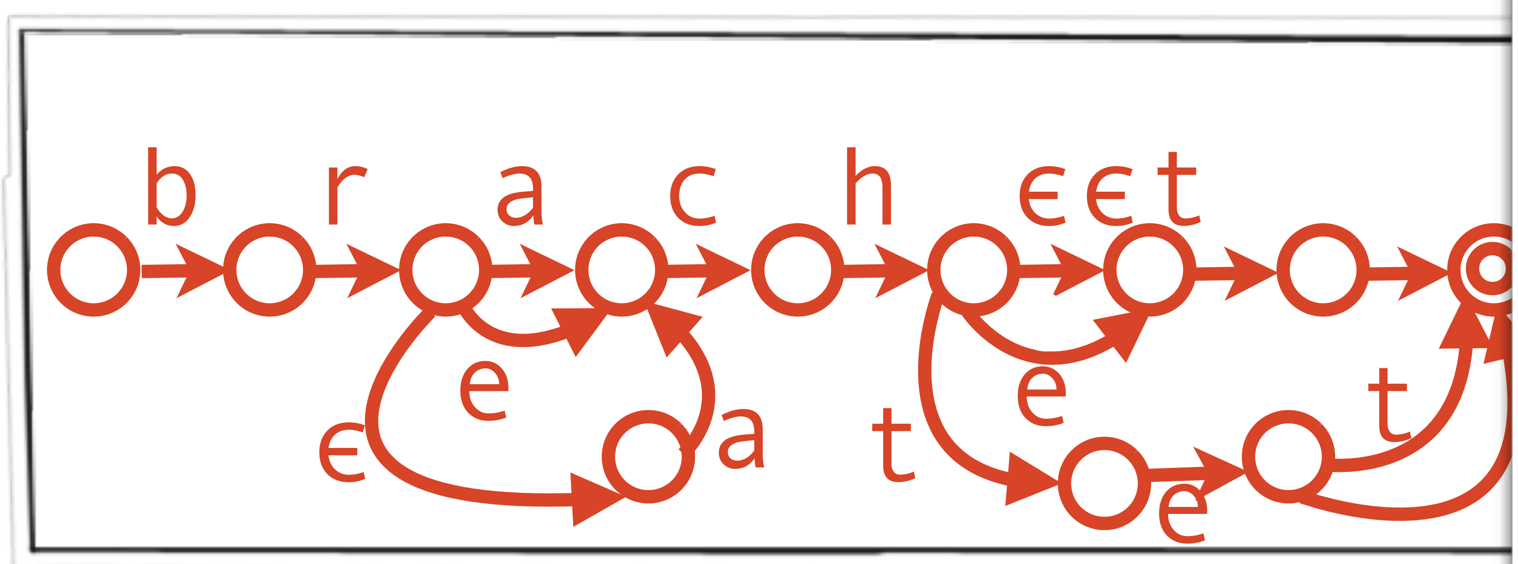
weighted finite-state acceptor

Inference. *Multiple strings*

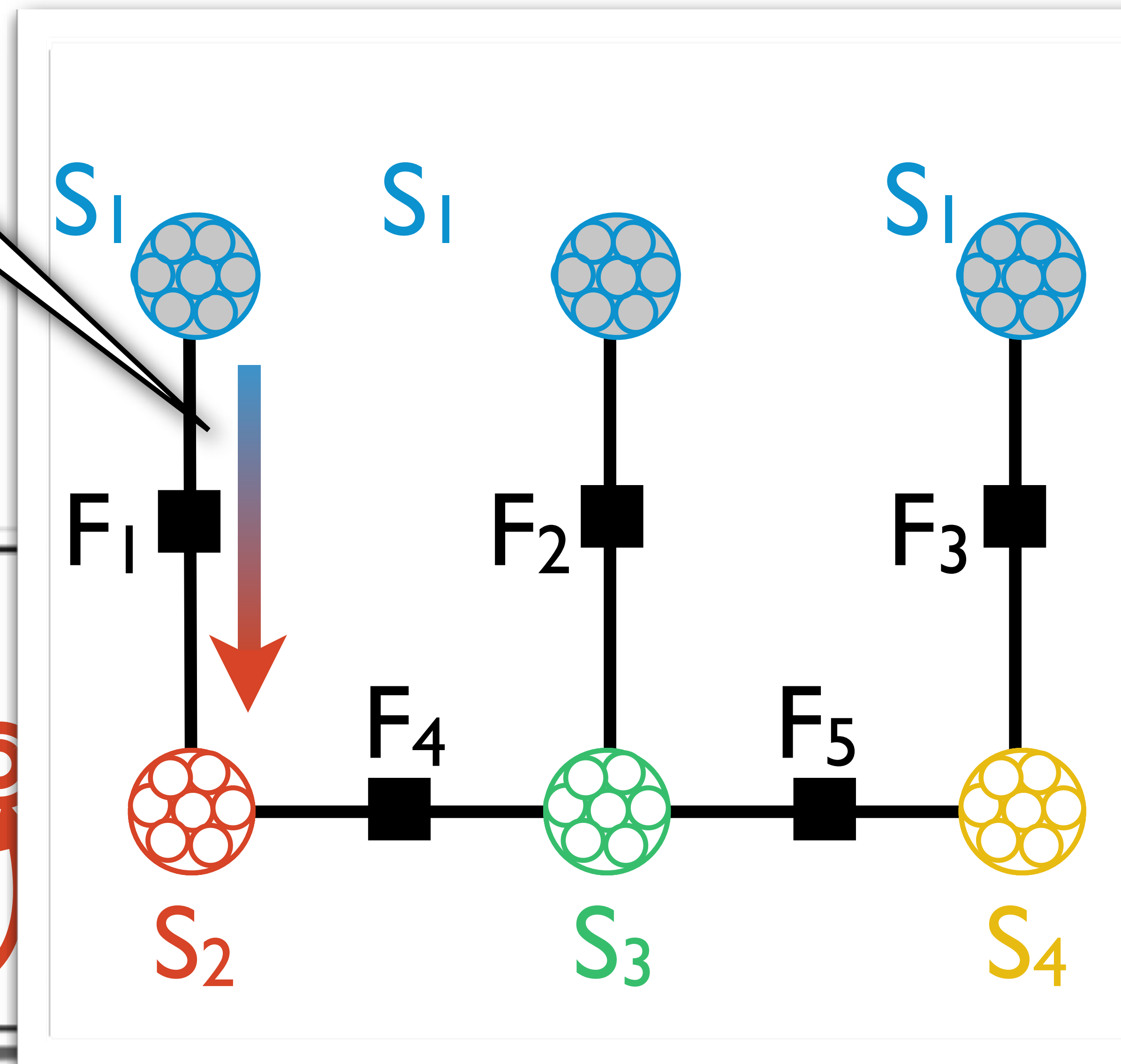
S_1 = 

(observed)

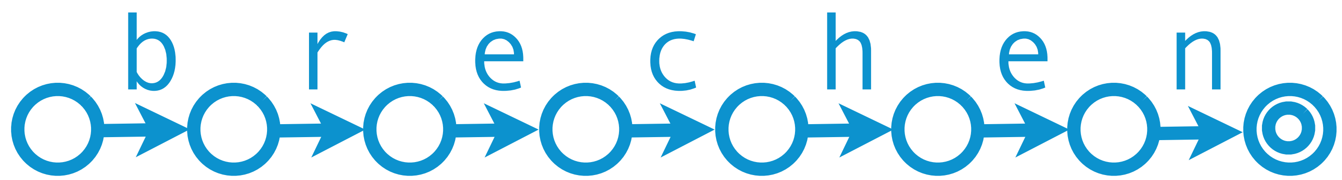
send S_1 through transducer F_1



weighted finite-state acceptor



Inference. *Multiple strings*

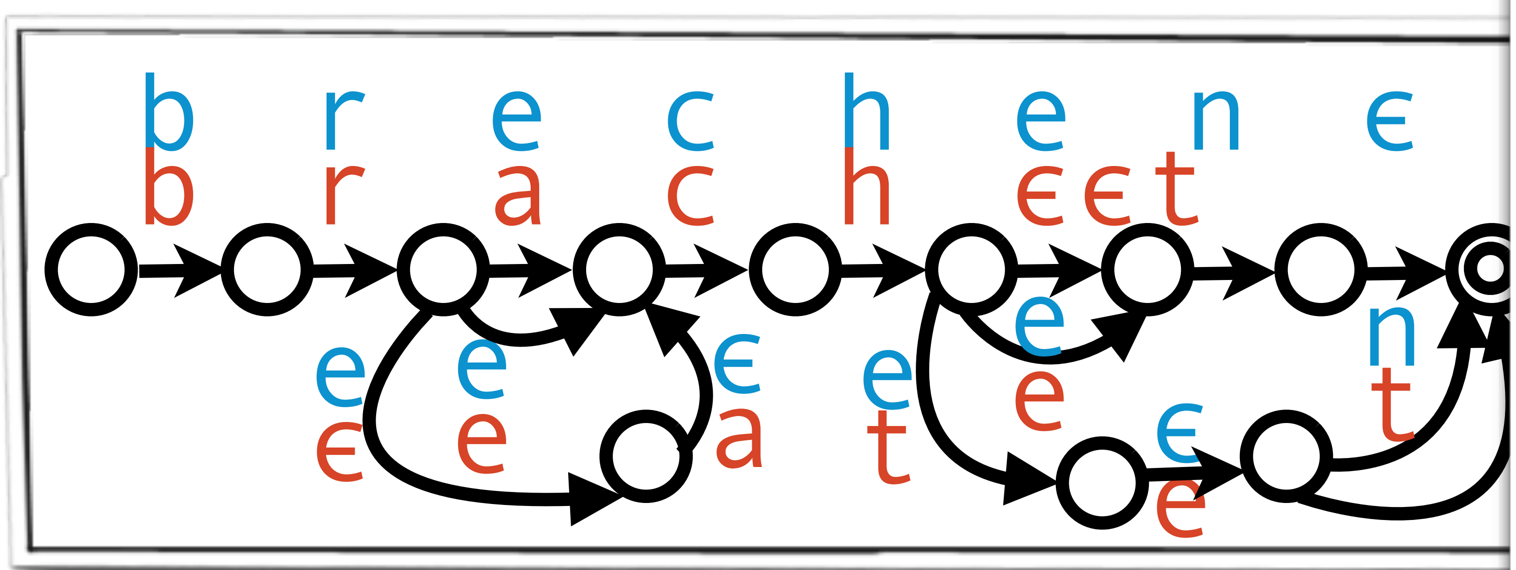
S_1 = 

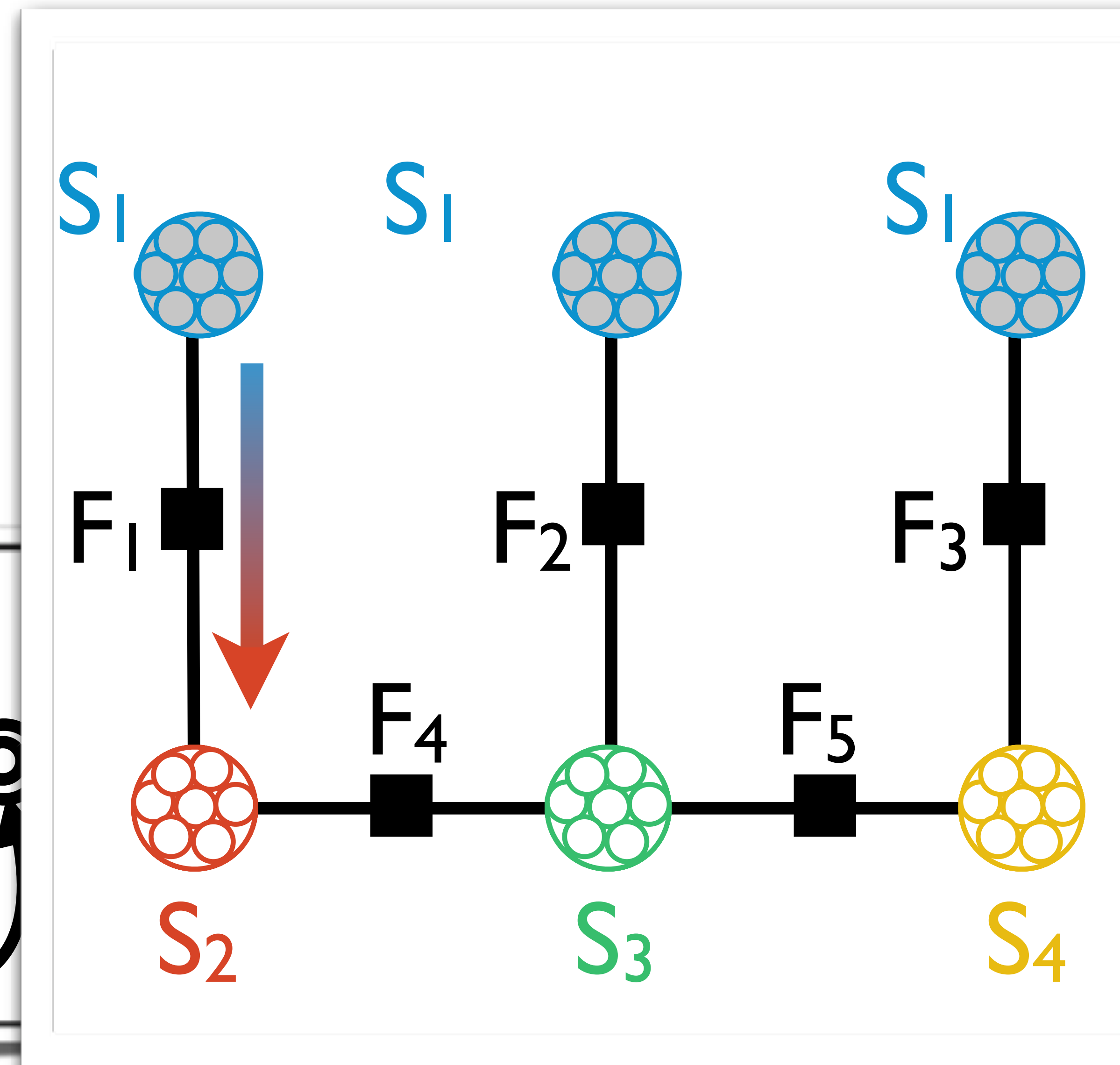
(observed)

Step 1:

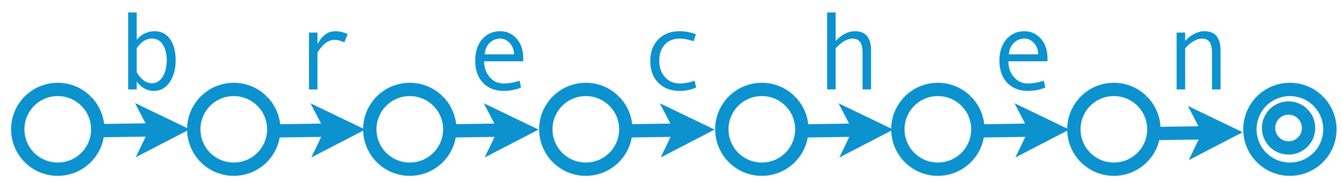
circle means **composition**

$S_1 \circ F_1$





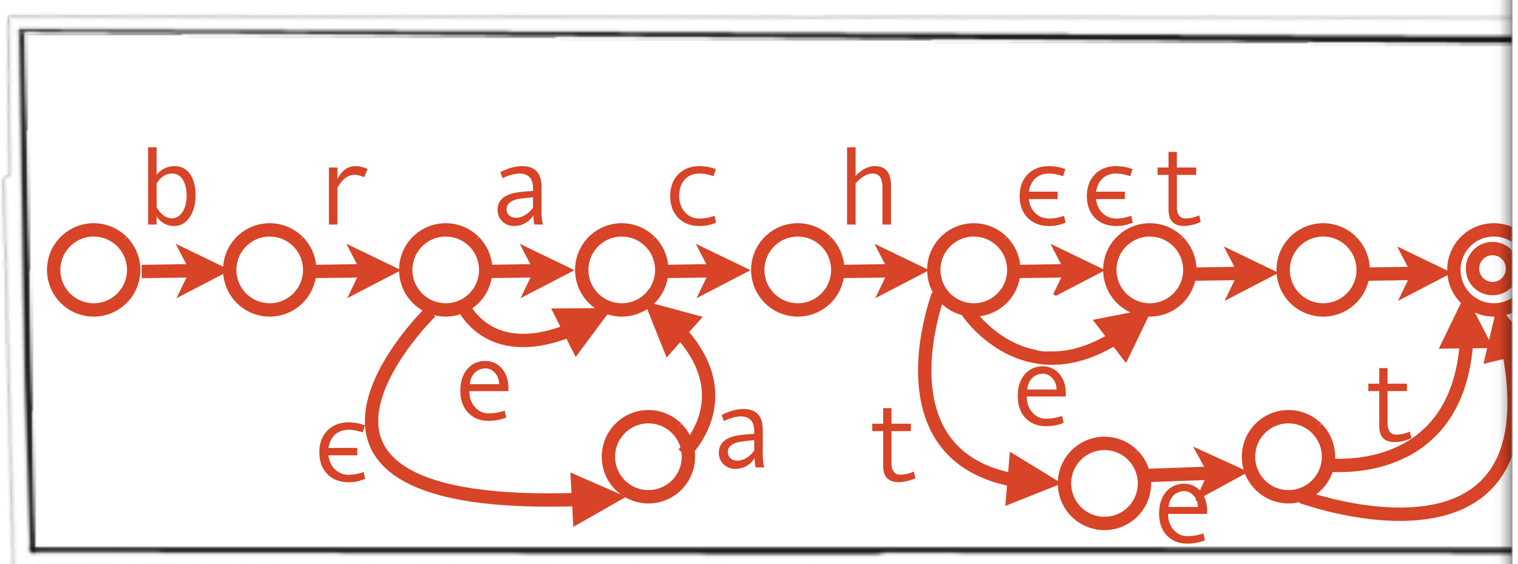
Inference. *Multiple strings*

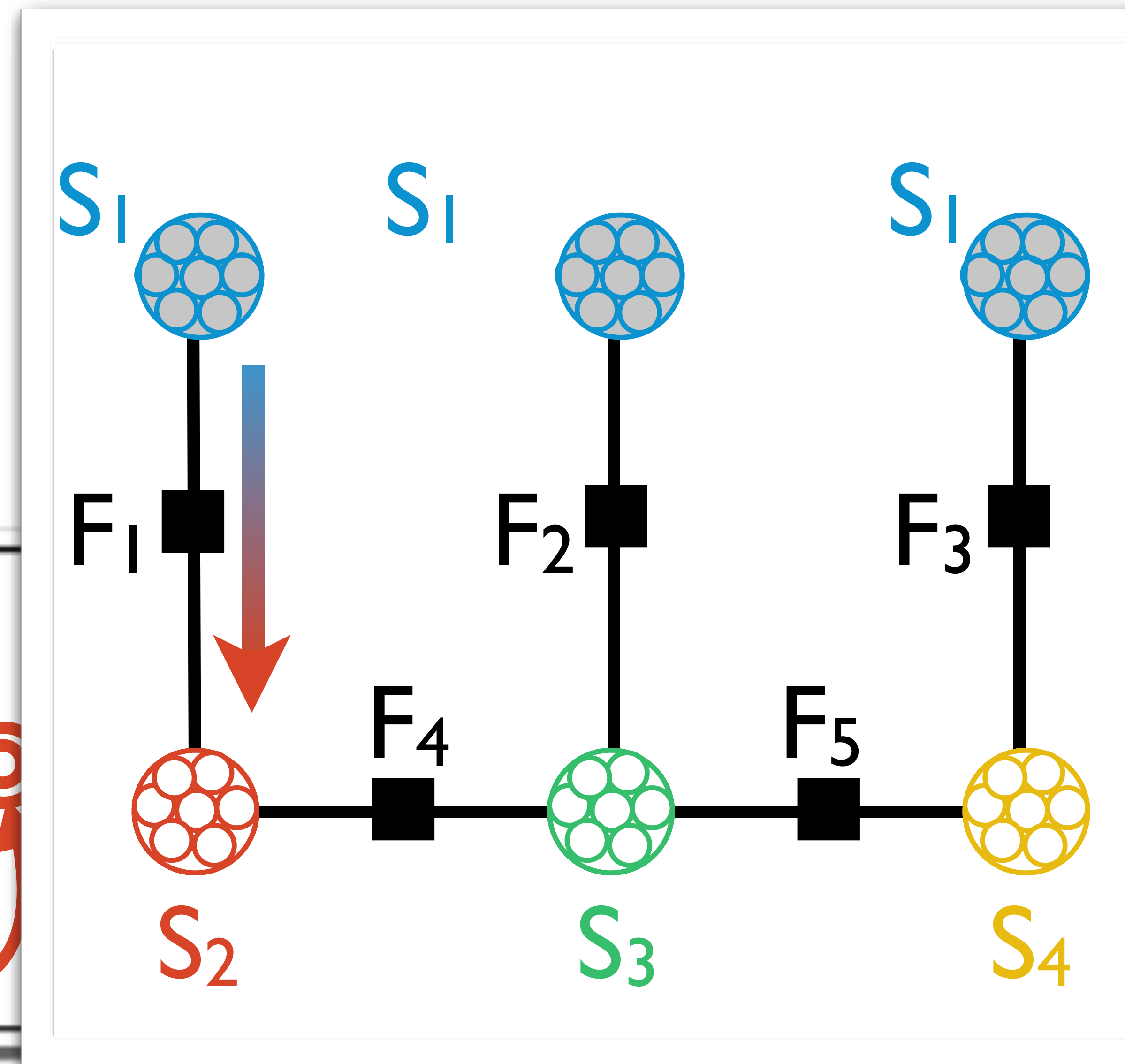
S_1 = 

(observed)

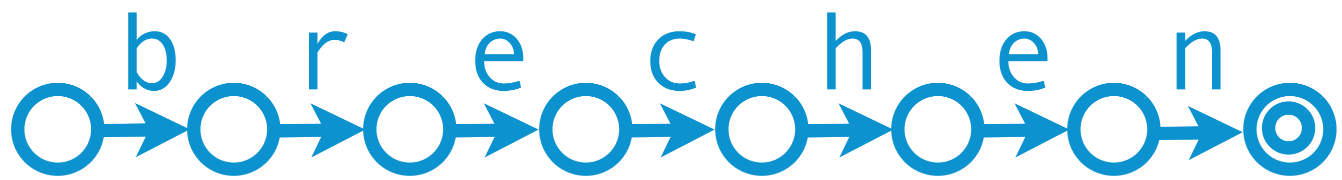
Step 2:

range($S_1 \circ F_1$)





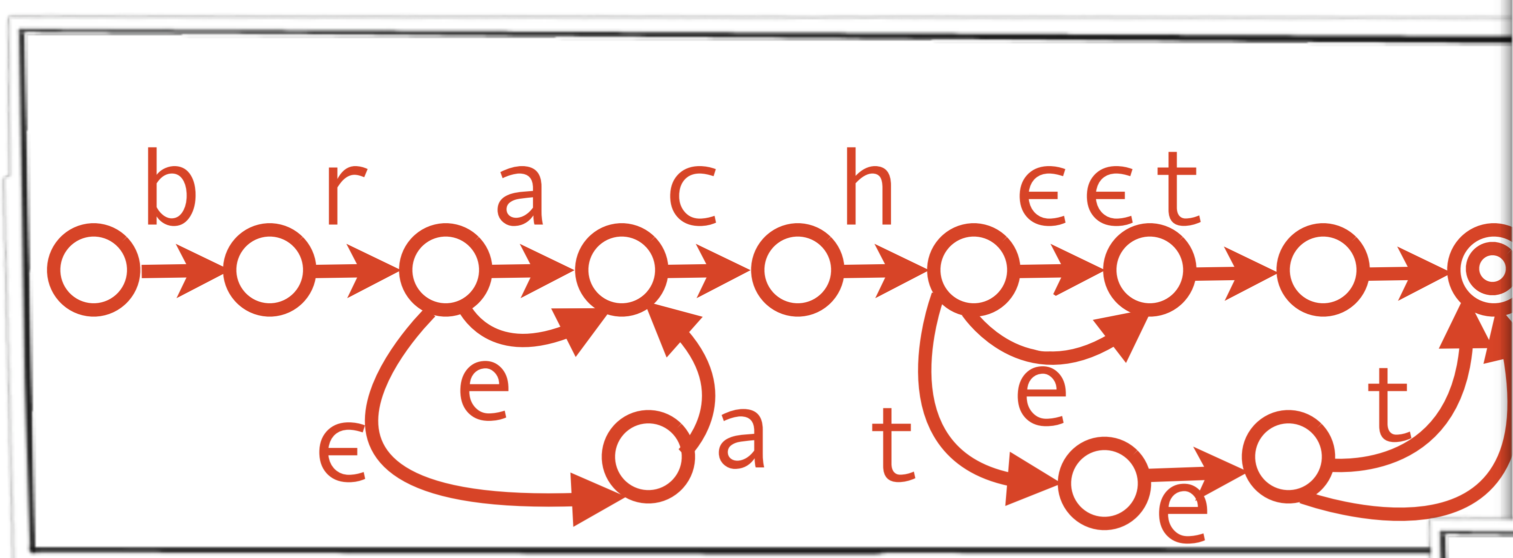
Inference. *Multiple strings*

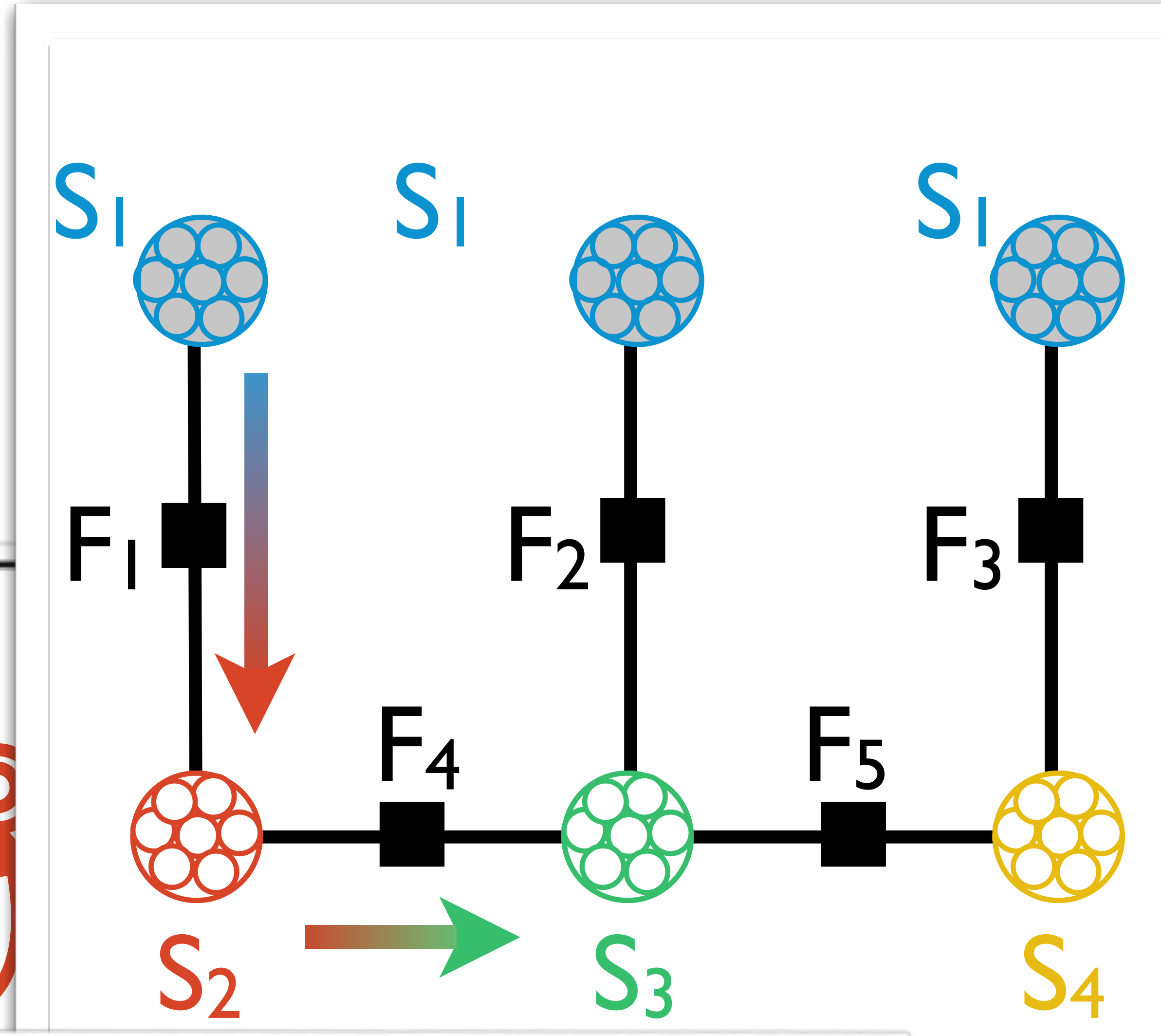
$S_1 =$ 

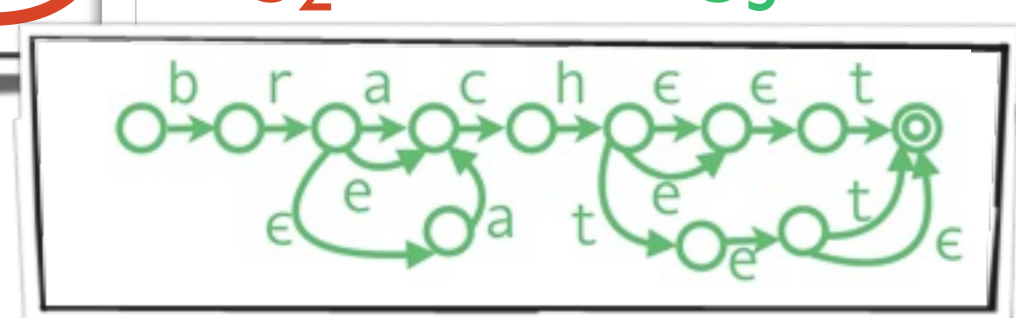
(observed)

Step 2:

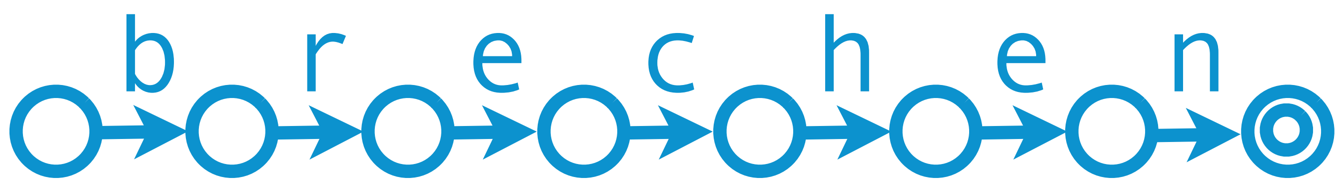
range($S_1 \circ F_1$)



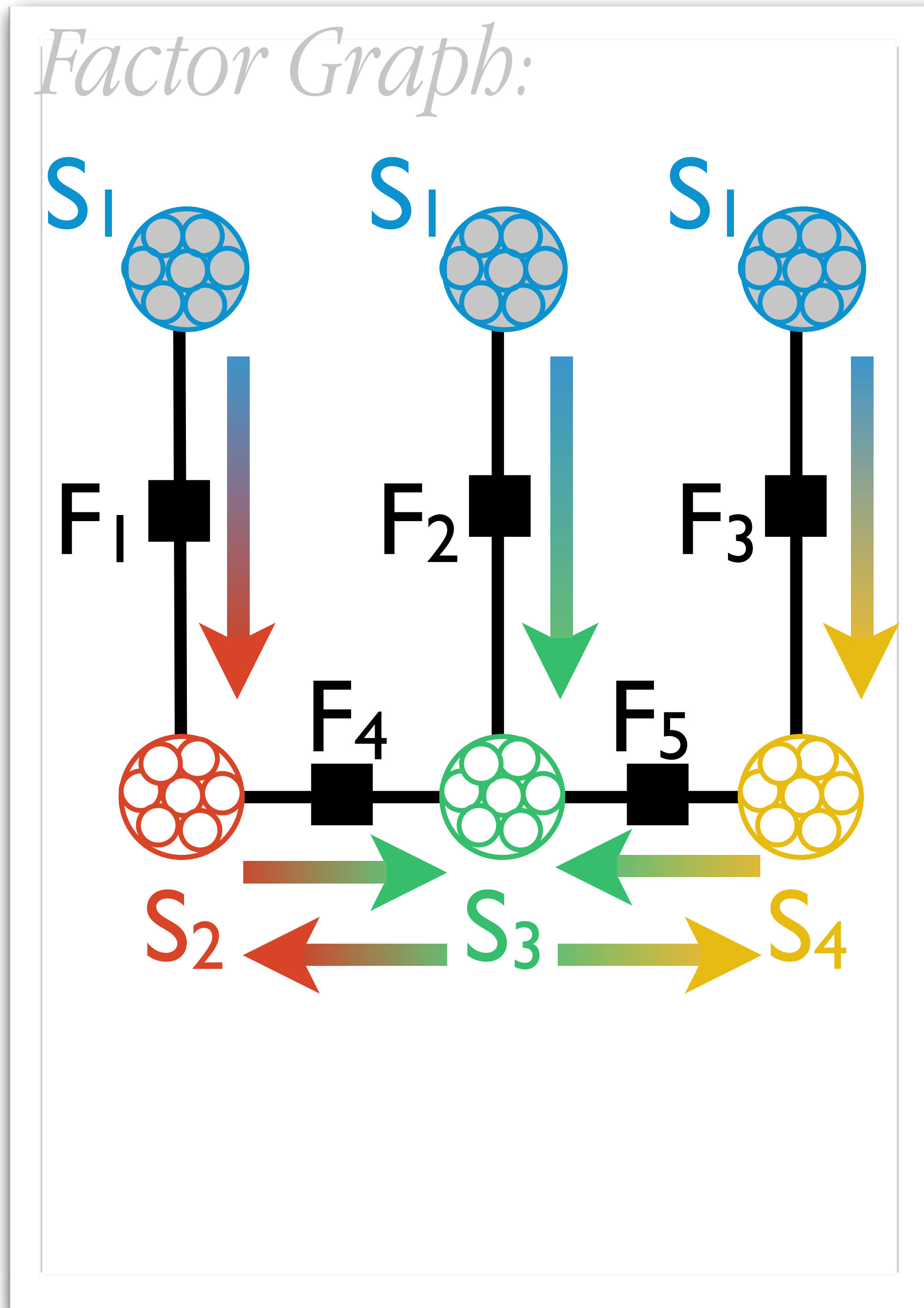




Inference. *Multiple strings*

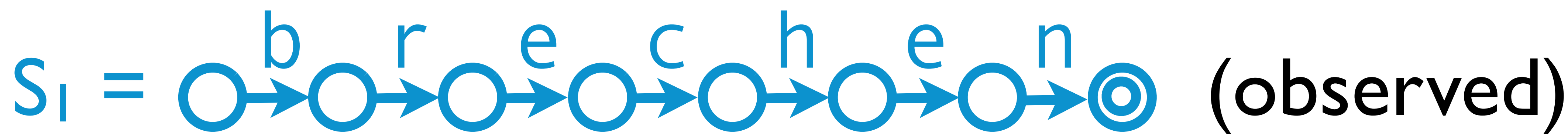
S_1 = 

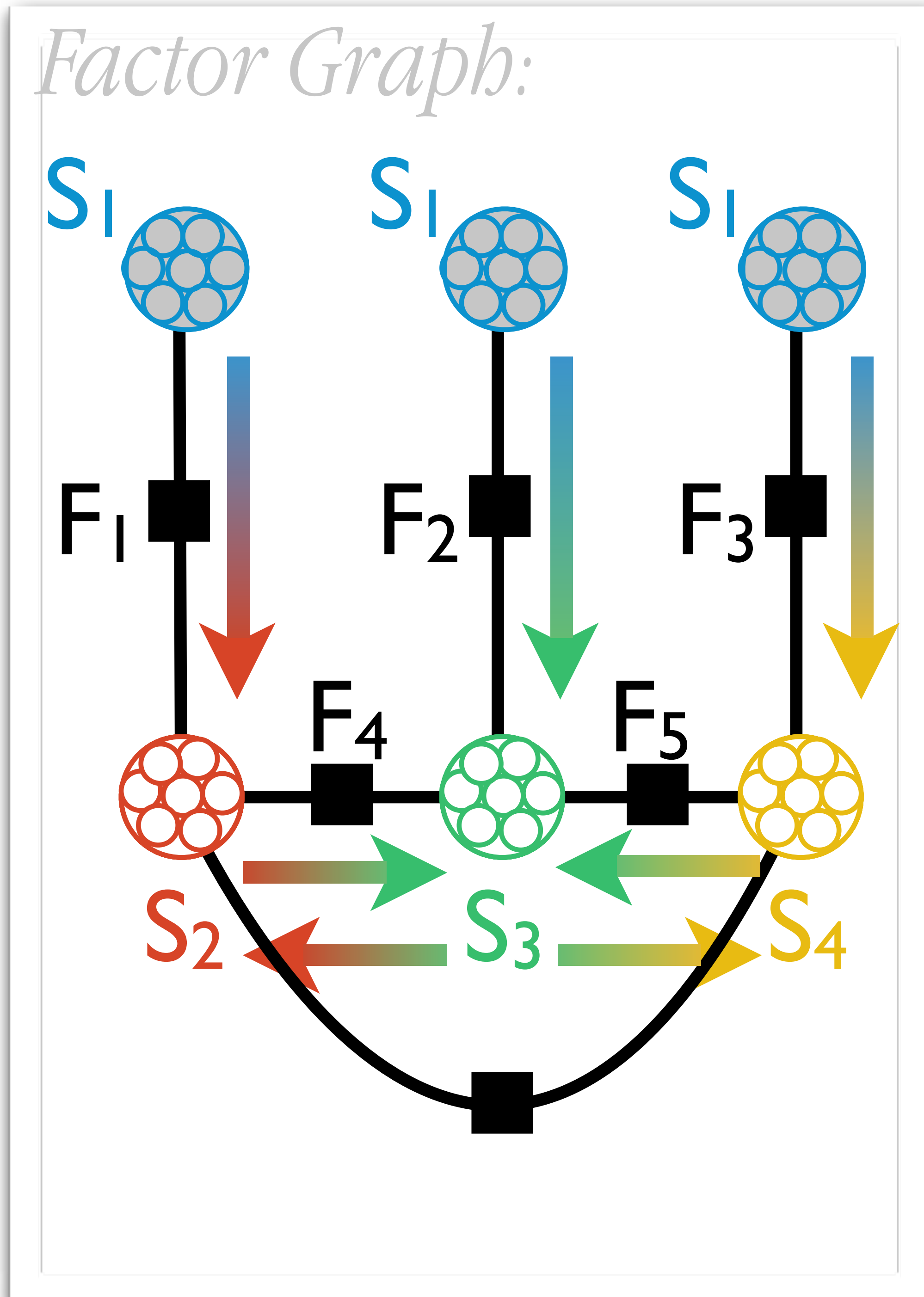
(observed)



- What happens if the factor graph has **loops**?
  - Messages **not independent** of each other anymore
  - Send anyway, **iterate**!
  - Obtained beliefs are only **approximate**

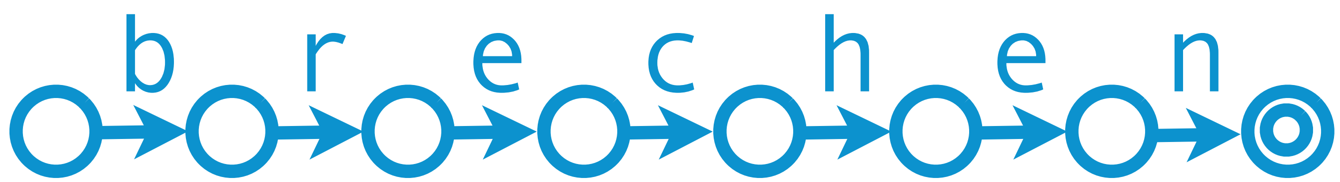
Inference. *Multiple strings*



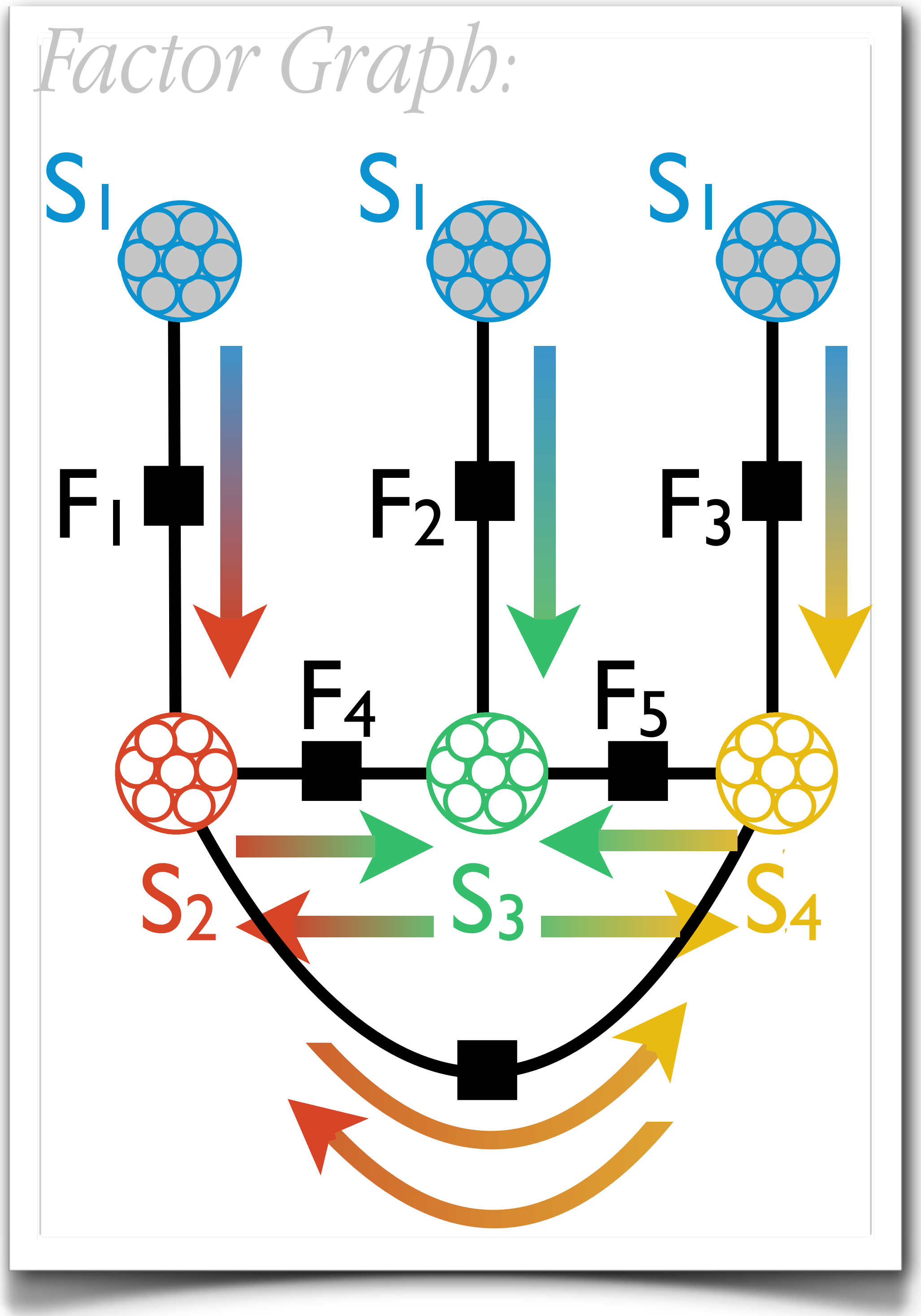


- What happens if the factor graph has **loops**?
  - Messages **not independent** of each other anymore
  - Send anyway, **iterate**!
  - Obtained beliefs are only **approximate**

Inference. *Multiple strings*

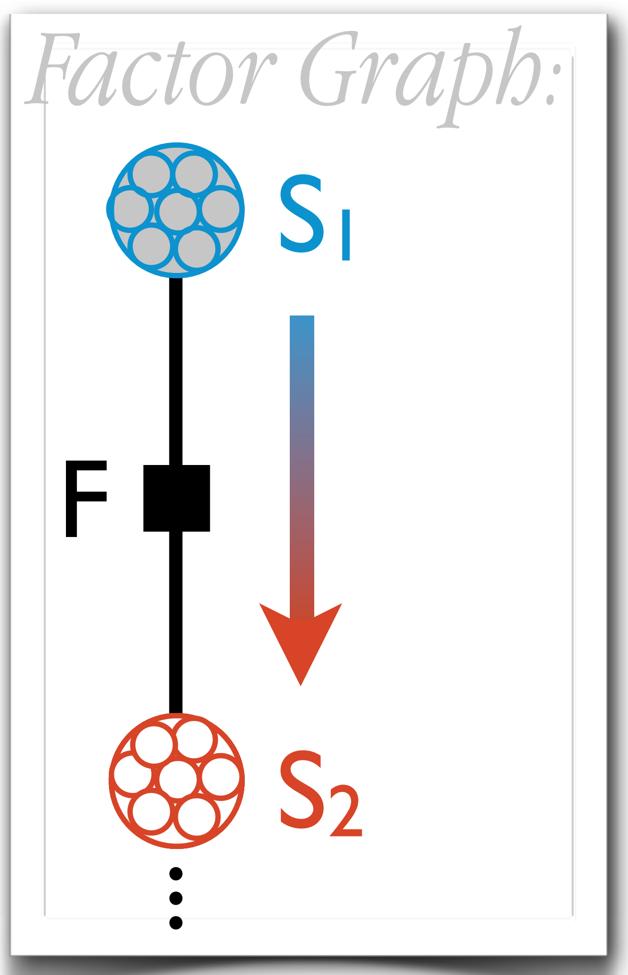
S_1 = 

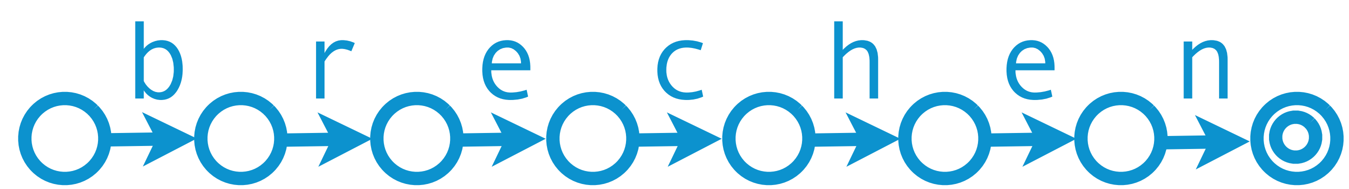
(observed)



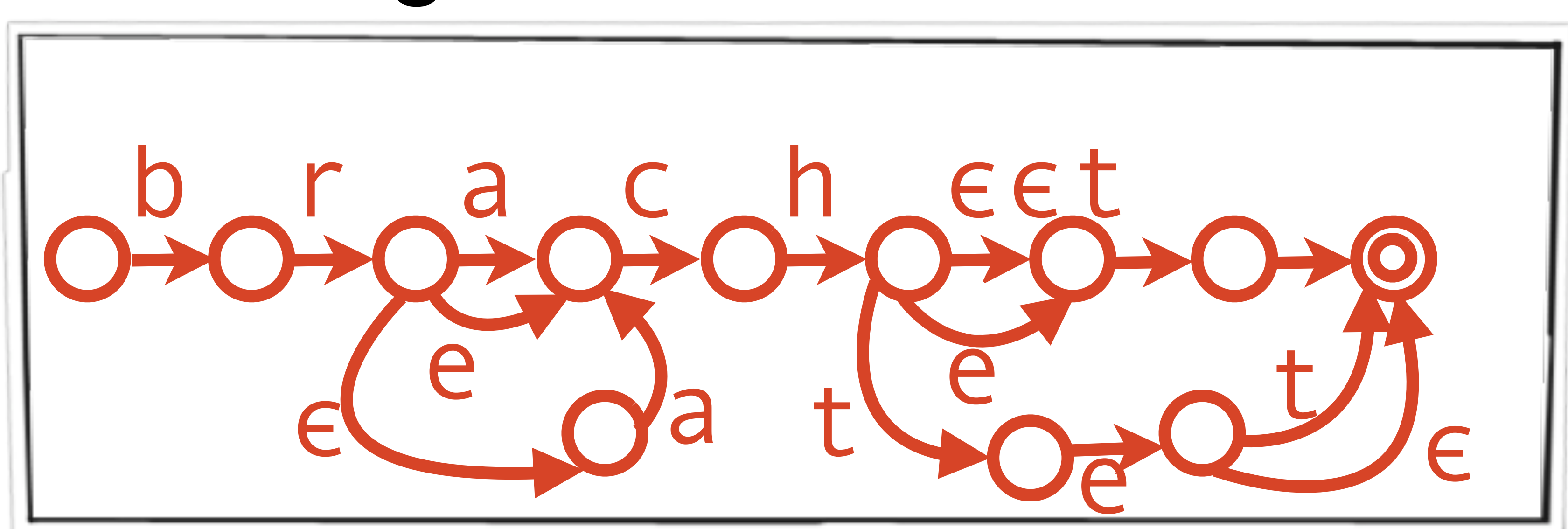
- What happens if the factor graph has **loops**?
  - Messages **not independent** of each other anymore
  - Send anyway, **iterate**!
  - Obtained beliefs are only **approximate**

Inference. *Approximations*



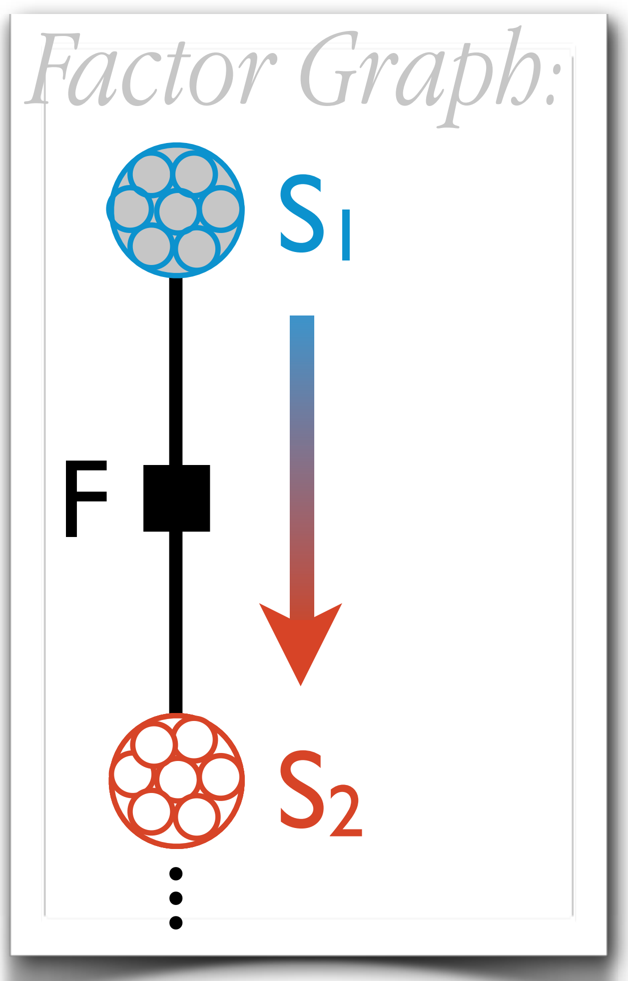
S_1 = 

send S_1 through transducer F:

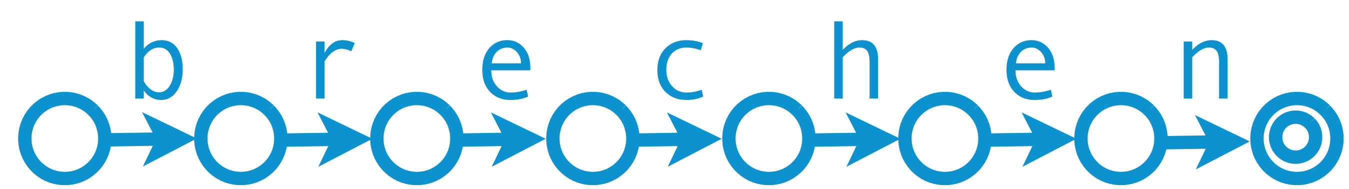


Inference. *Approximations*

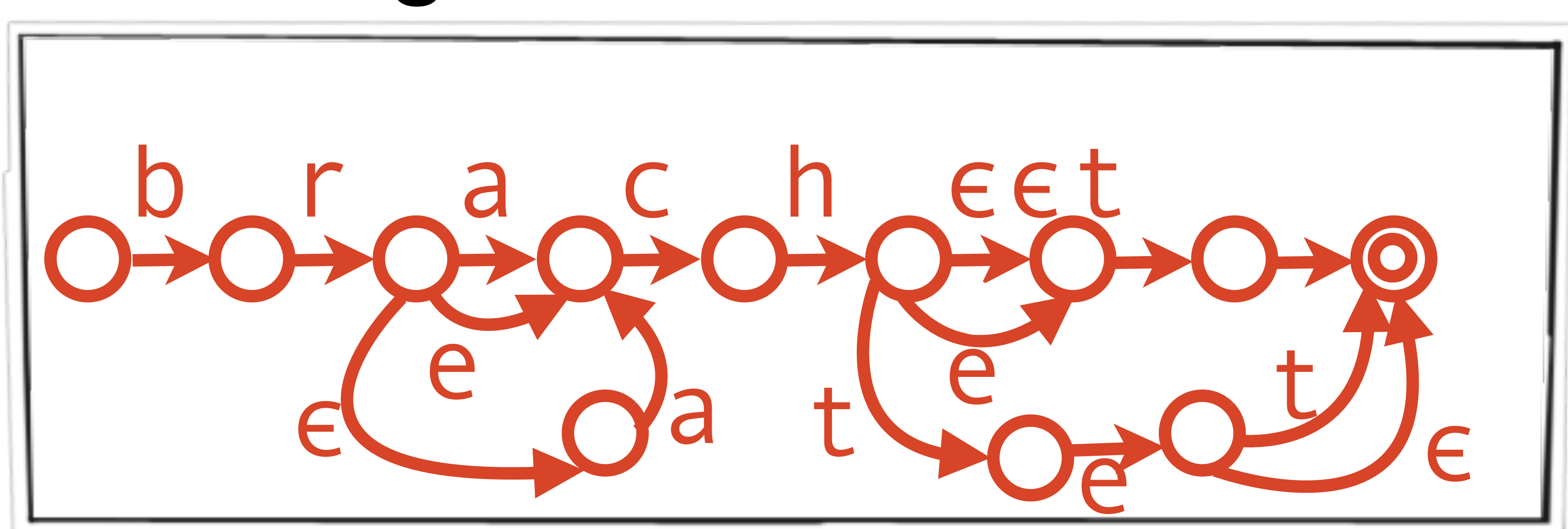
- A message becomes bigger when sent through transducer!



S_1 =

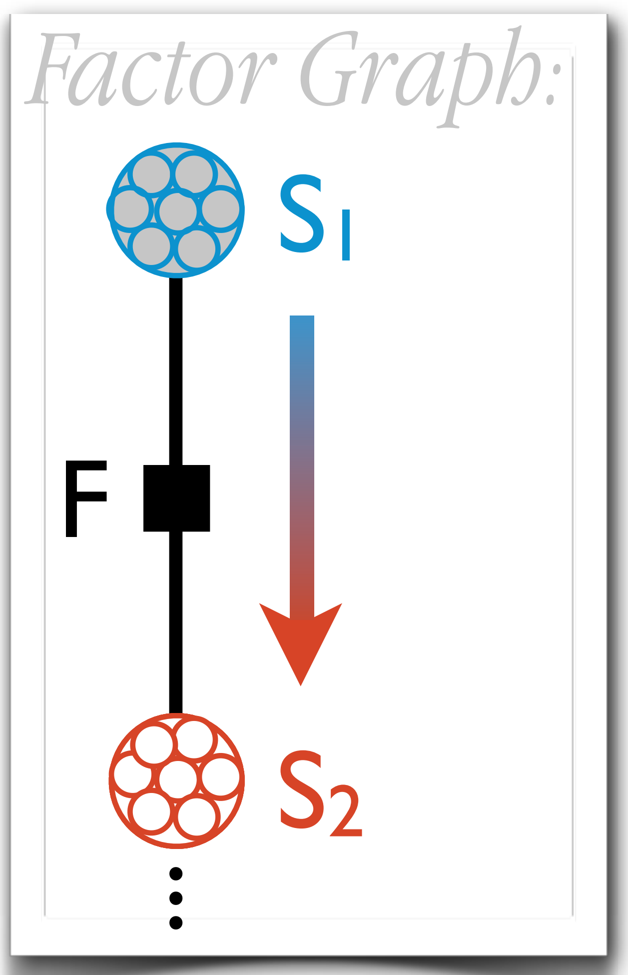


send S_1 through transducer F:

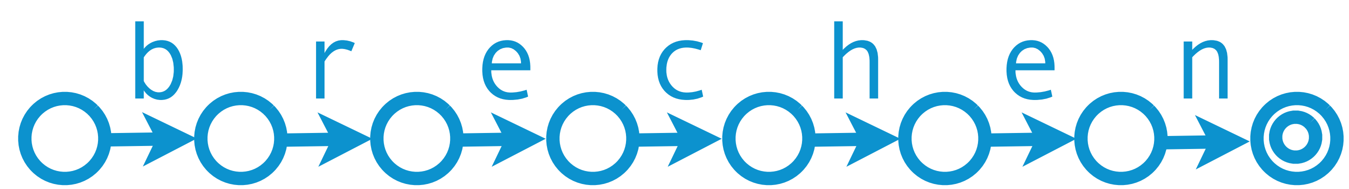


Inference. *Approximations*

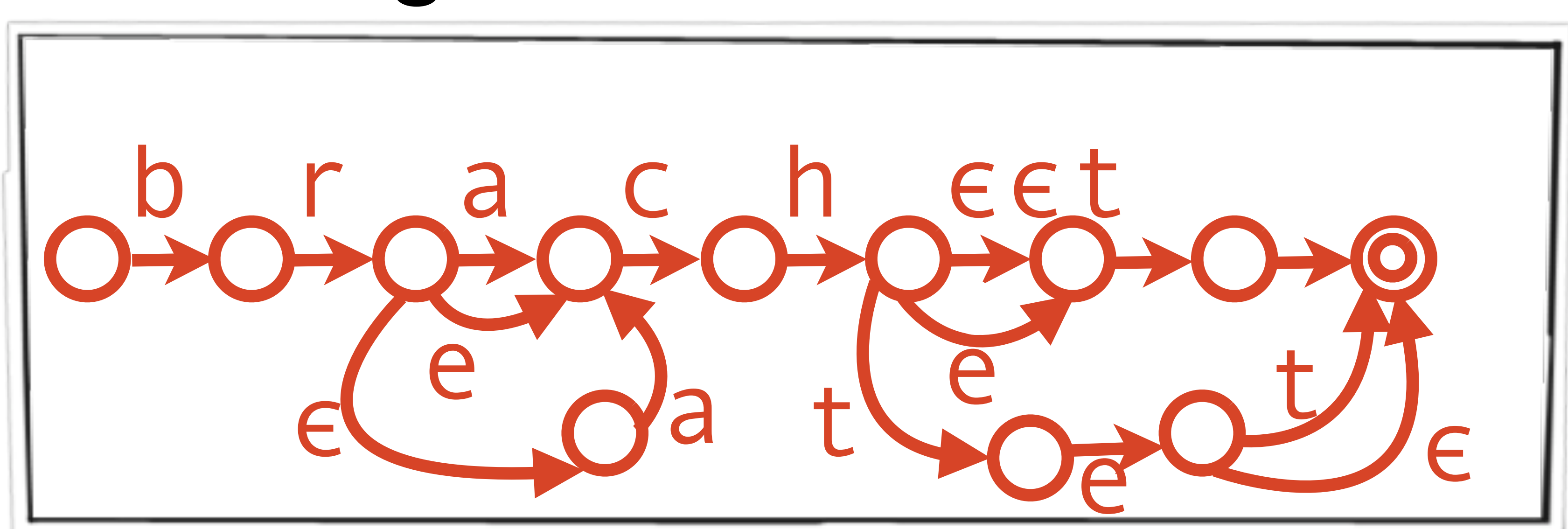
- A message becomes bigger when sent through transducer!
- And we keep sending from transducer to transducer, so messages **keep growing** in size!



S_1 =

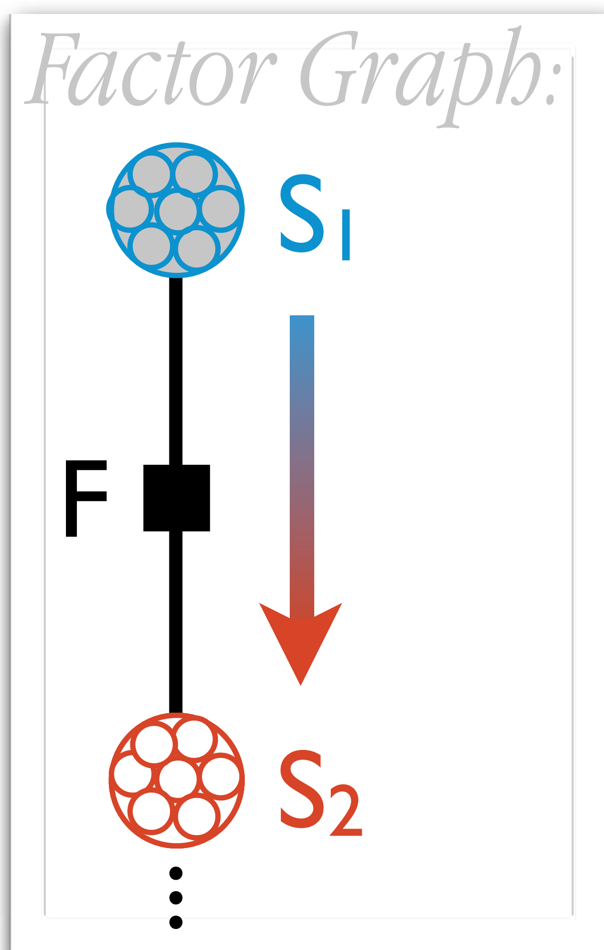


send S_1 through transducer F:

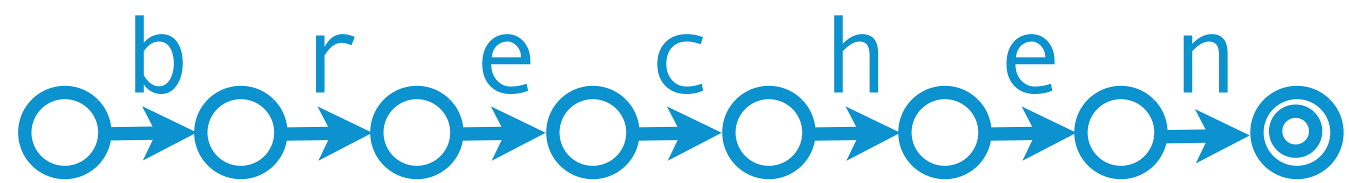


Inference. *Approximations*

- A message becomes bigger when sent through transducer!
- And we keep sending from transducer to transducer, so messages **keep growing** in size!



S_1 =



send S_1 through transducer F:

Inference. *Approximations*

- **Determinize** does not help
- Solution: **Approximate** messages!
  - n-gram approximation
  - k-best-paths approximation
  - mixture model: use both!

Li, Eisner & Khudanpur, 2009

In our experiments:
k=1000, n=0

Overview

- Motivation
- Model
- Inference & Approximations
- **Experiments**
- Conclusions

Experiments

Task: Reconstruct missing word forms in morphological paradigms (German)

infinitive	brechen			
1st	breche	brechen ?	brach	brachen ?
2nd	brichst ?	brecht ?	brachst ?	bracht
3rd	bricht	brechen	brach ?	brachen
	singular	plural	singular	plural
	present		past	

Missing: forms that occur rarely in free text *(i.e. frequency count <10 in CELEX)*

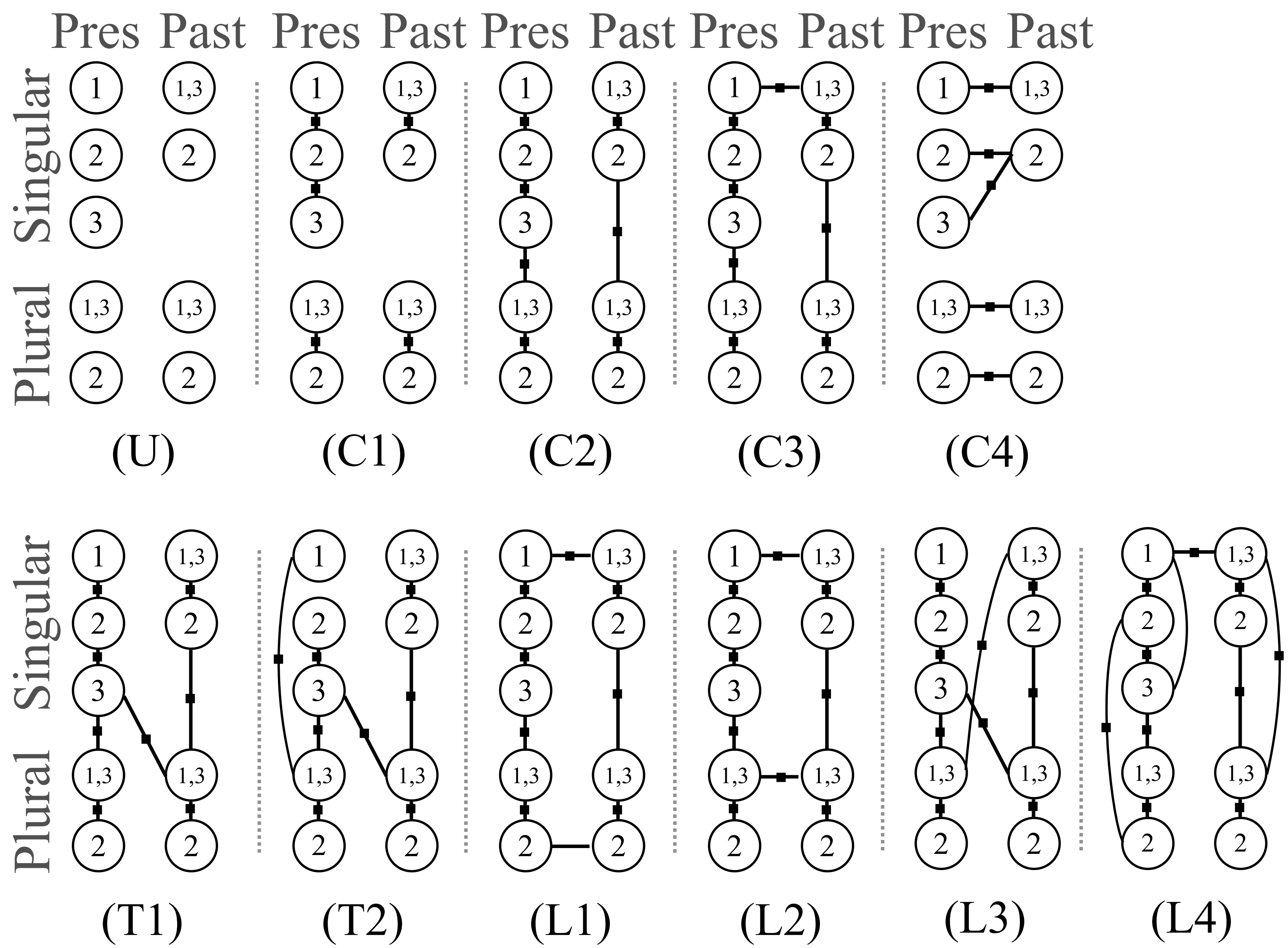
Experiments

- Train model parameters from the observed forms in the 9393 paradigms (piecewise training)
- Task: Exactly reconstruct all missing forms!
- Use 100 paradigms as dev set for model selection, evaluate on remaining 9293 paradigms

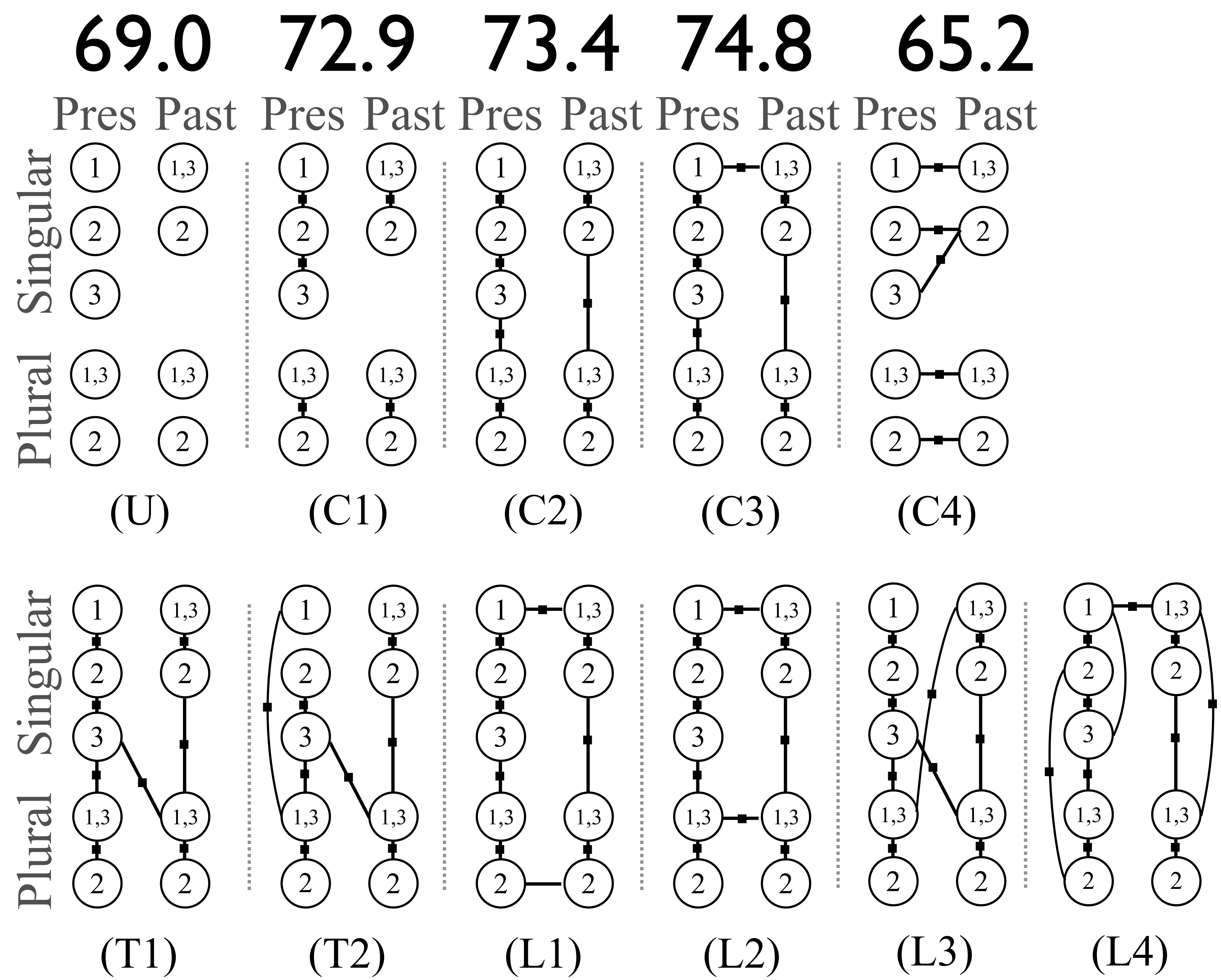
infinitive	brechen

	present		past	
	singular	plural	singular	plural
1st	breche	bre?en	brach	bra?en
2nd	bri?st	bre?ht	bra?nst	bracht
3rd	bricht	brechen	bra?h	brachen

Experiments

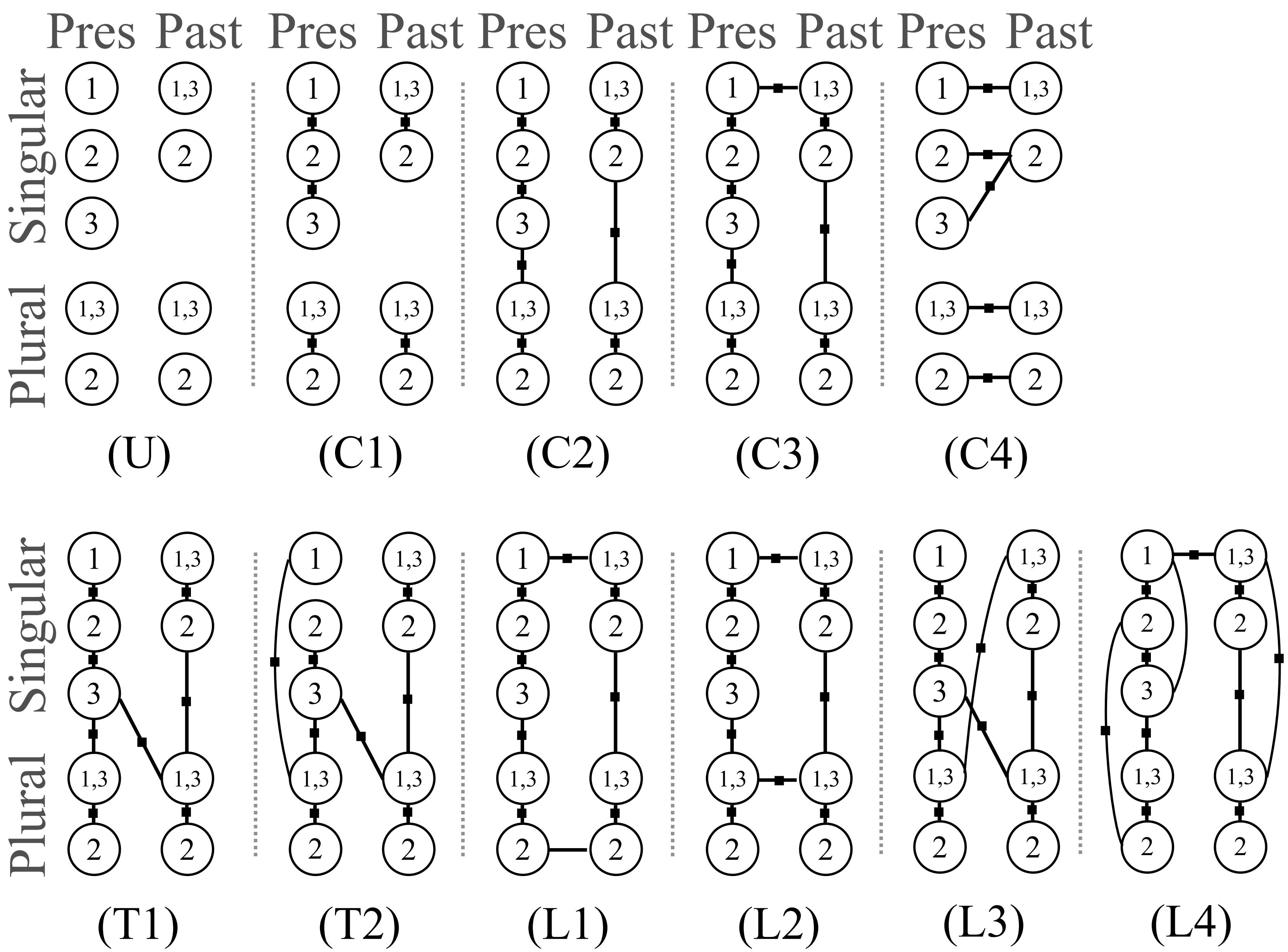


Experiments



Experiments

69.0 72.9 73.4 74.8 65.2



78.1 78.7 62.3 79.6 78.9 **82.1**

Experiments

		Moses baselines		Un-connected	Loopy
Form	# obs.	M,3	M,15	U	L4
All	6633	59.2	67.3	68.0	**81.2**

Experiments

Form	# obs.	M,3	M,15	U	L4
2.Sg.Pa.	4	0.0	0.2	0.8	**69.7**
2.Pl.Pa.	9	0.9	1.1	1.4	**45.6**
2.Sg.Pr.	166	49.4	62.6	74.7	**90.5**
1.Sg.Pr.	285	**99.6**	98.8	99.3	97.2
1,3.Pl.Pa.	673	46.5	**78.3**	75.0	75.6
1,3.Sg.Pa.	1124	65.0	**88.8**	84.0	74.8
2.Pl.Pr.	1274	98.3	**99.2**	99.0	96.4
3.Sg.Pr.	1410	91.0	**95.9**	95.2	88.2
1,3.Pl.Pr.	1688	**99.8**	98.9	**99.8**	98.0
All	6633	59.2	67.3	68.0	**81.2**

Moses baselines: M,3 and M,15; Un-connected: U; Loopy: L4

more obser-vations

Experiments

		Moses baselines		Un-connected	Loopy
Form	# obs.	M,3	M,15	U	L4
2.Sg.Pa.	4	0.0	0.2	0.8	**69.7**
2.Pl.Pa.	9	0.9	1.1	1.4	**45.6**
2.Sg.Pr.	166	49.4	62.6	74.7	**90.5**
1.Sg.Pr.	285	**99.6**	98.8	99.3	97.2
1,3.Pl.Pa.	673	46.5	**78.3**	75.0	75.6
1,3.Sg.Pa.	1124	65.0	**88.8**	84.0	74.8
2.Pl.Pr.	1274	98.3	**99.2**	99.0	96.4
3.Sg.Pr.	1410	91.0	**95.9**	95.2	88.2
1,3.Pl.Pr.	1688	**99.8**	98.9	**99.8**	98.0
All	6633	59.2	67.3	68.0	**81.2**

more obser-vations

Experiments

Moses baselines | Un-connected | Loopy

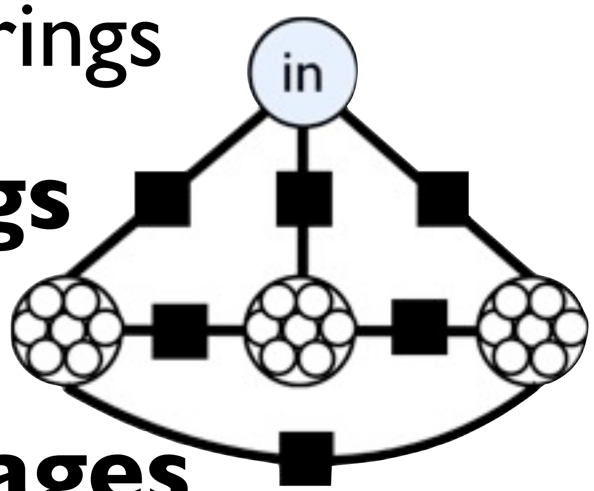
Form	# obs.	M,3	M,15	U	L4
2.Sg.Pa.	4	0.0	0.2	0.8	**69.7**
2.Pl.Pa.	9	0.9	1.1	1.4	**45.6**
2.Sg.Pr.	166	49.4	62.6	74.7	**90.5**
1.Sg.Pr.	285	**99.6**	98.8	99.3	97.2
1,3.Pl.Pa.	673	46.5	**78.3**	75.0	75.6
1,3.Sg.Pa.	1124	65.0	**88.8**	84.0	74.8
2.Pl.Pr.	1274	98.3	**99.2**	99.0	96.4
3.Sg.Pr.	1410	91.0	**95.9**	95.2	88.2
1,3.Pl.Pr.	1688	**99.8**	98.9	**99.8**	98.0
All	6633	59.2	67.3	68.0	**81.2**

more obser-vations

Overview

- Motivation
- Model
- Inference & Approximations
- Experiments
- **Conclusions**

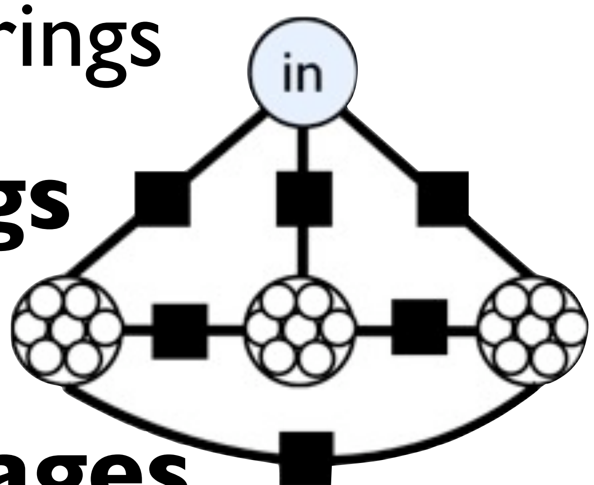
Conclusions



- **Jointly predict** multiple interdependent strings
- Undirected **graphical model over strings** (variables: *strings*, factors: *finite-state machines*)
- Belief propagation with **finite-state messages**
- Approximations:
  - Loopy BP
  - Approximate messages to prevent blowup
- Showed results in **morphology**, potentially useful for many other string tasks (transliteration, cognate modeling, ...)

Conclusions

- **Jointly predict** multiple interdependent strings
- Undirected **graphical model over strings** (variables: *strings*, factors: *finite-state machines*)



- Belief propagation with **finite-state messages**
- Approximations:
  - Loopy BP
  - Approximate messages to prevent blowup
- Showed results in **morphology**, potentially useful for many other string tasks (transliteration, cognate modeling, ...)

General idea: Coordinate NLP models (FSTs, PCFGs, ...) by using them as factors in graphical models!