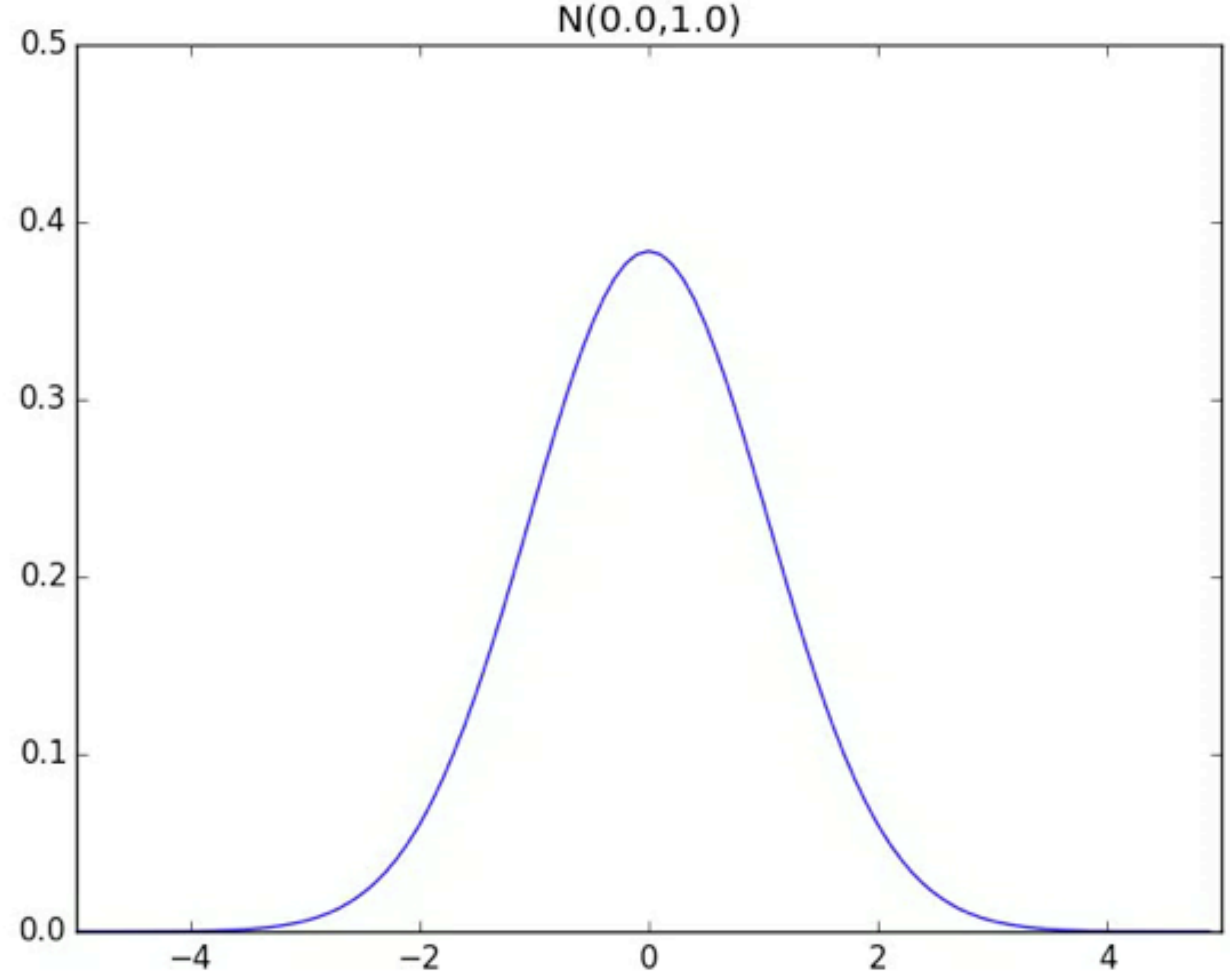
EM

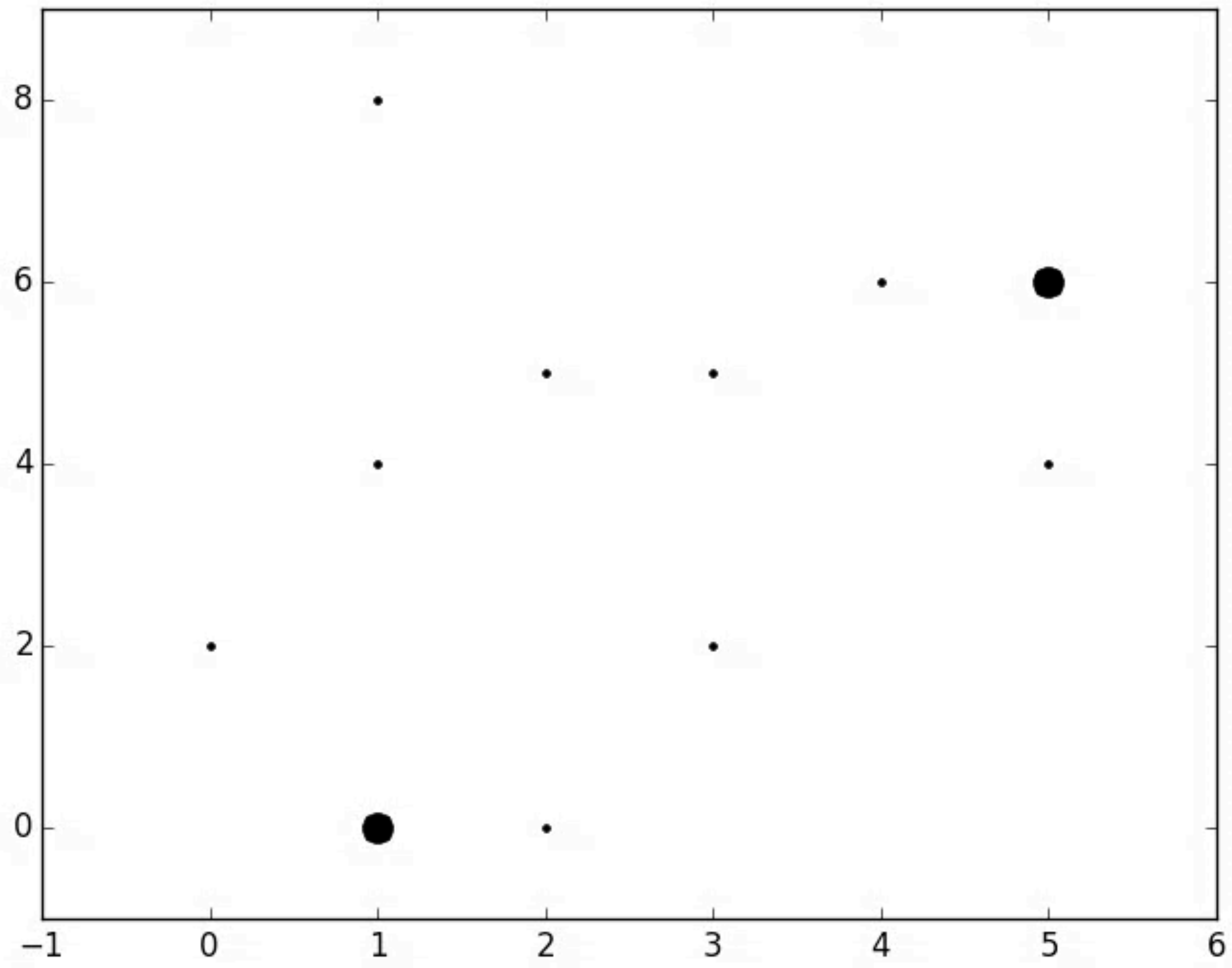
Monday, February 16, 2015

Plan for Today:

- K-Means
- EM algorithm

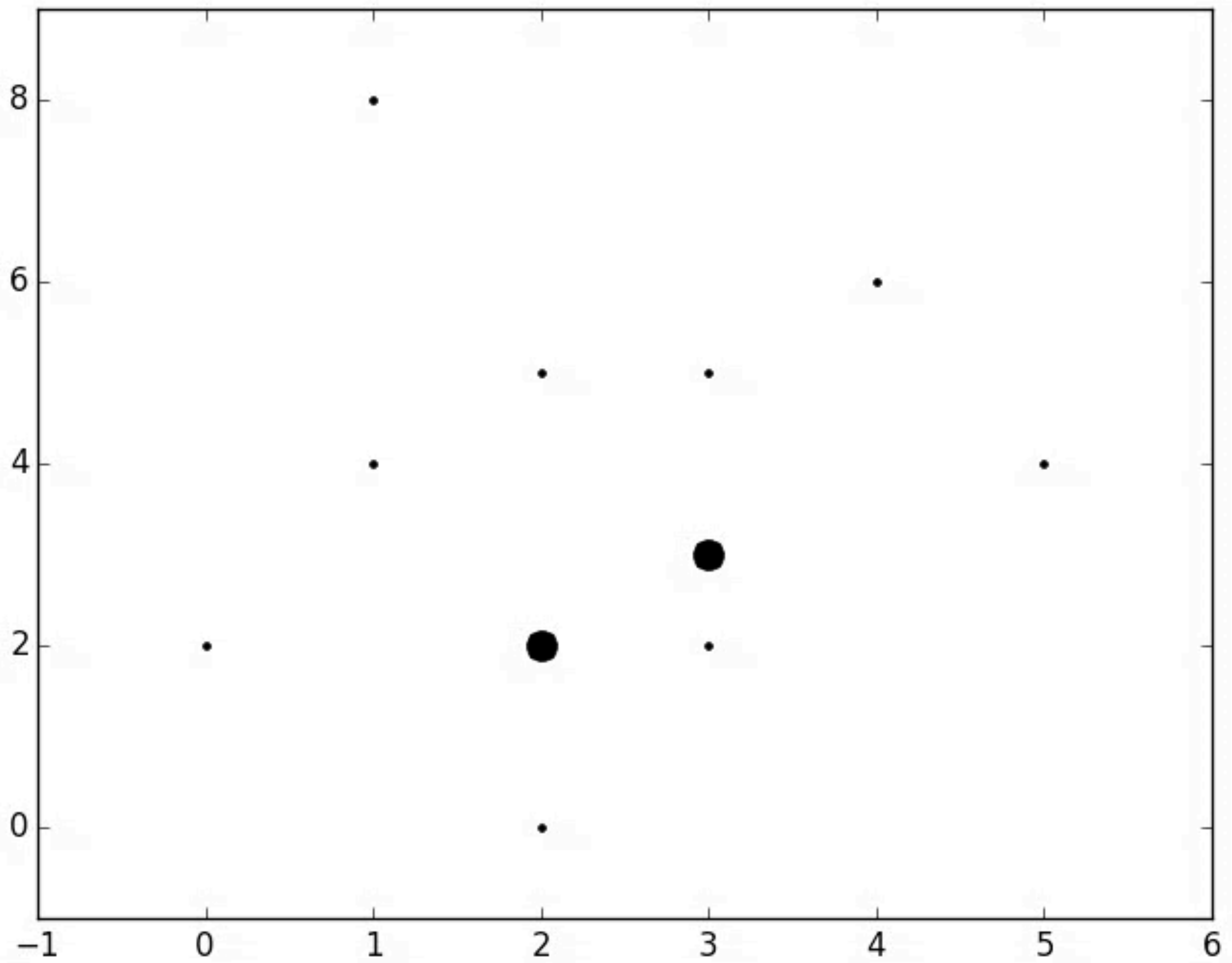


K-Means



K-Means

With different initialization…



Does K-Means always converge so quickly?

(Demo)

EM

Expectation-Maximization

Family of algorithms for iteratively finding:

- The parameters of a model
- The best explanation for data

EM

Expectation Step:

- Given the current parameters, what's the most likely explanation for the data?

Maximization Step:

- Given the current explanation for the data, what's the most likely value for each of the model parameters?

For k-Means

- E: Given two centers, which cluster does each point most likely come from?
- M: Given cluster assignments, what's the MLE of the cluster means?

Model description

We still have to define the model

EM will find the best parameterization for the model we pick, regardless of how well that model reflects the underlying data.

What assumptions does k-means make about the data?

Another possibility: Gaussian Mixture Models

Gaussian Mixture Models (GMMs)

Key idea: Data is being generated by some number of gaussian (normally-distributed) variables

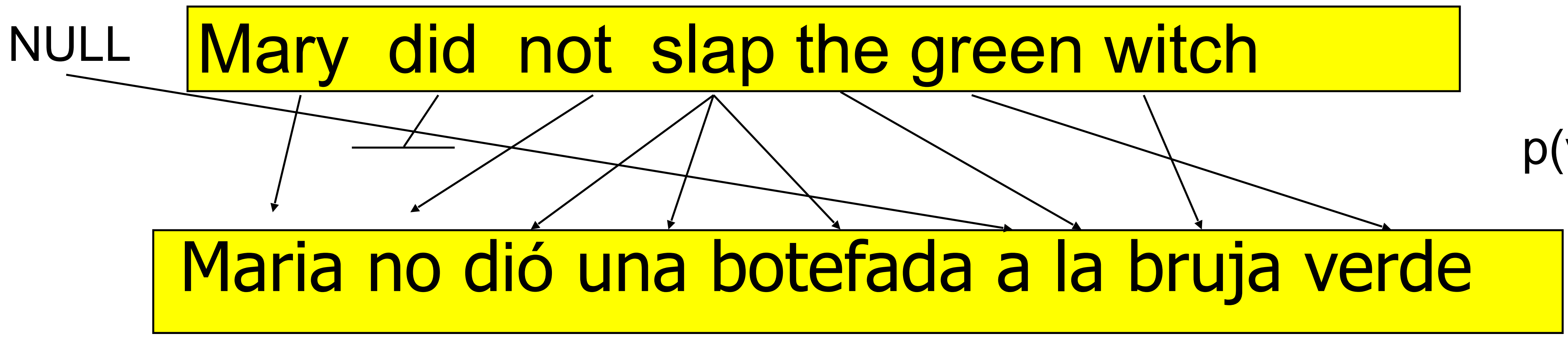
E-Step: Decide which gaussian was most likely to have generated each point

M-Step: Re-estimate the mean, variance, and prior for each gaussian

GMM Demo

(Demo)

Word Models: IBM Model 1



p(verde | green)

Each foreign word is aligned to exactly one English word

This is the **ONLY** thing we model!

$$p(f_1 f_2 \ldots f_{|F|}, a_1 a_2 \ldots a_{|F|} \mid e_1 e_2 \ldots e_{|E|}) = \prod_{i=1}^{|F|} p(f_i \mid e_{a_i})$$

Training Word Models

The old man is happy. He has fished many times.	El viejo está feliz porque ha pescado muchos veces.
His wife talks to him.	Su mujer habla con él.
The sharks await.	Los tiburones esperan.
…	…

$$p(f_1 f_2 \ldots f_{|F|}, a_1 a_2 \ldots a_{|F|} \mid e_1 e_2 \ldots e_{|E|}) = \prod_{i=1}^{|F|} p(f_i \mid e_{a_i})$$

$p(f_i \mid e_{a_i})$: probability that *e* is translated as *f*

Thought Experiment

The old man is happy. He has fished many times.

El viejo está feliz porque ha pescado muchos veces.

His wife talks to him.

Su mujer habla con él.

The sharks await.

Los tiburones esperan.

$$p(f_i \mid e_{a_i}) = \ ?$$

Thought Experiment

The old man is happy. He has fished many times.

El viejo está feliz porque ha pescado muchos veces.

His wife talks to him.

Su mujer habla con él.

The sharks await.

Los tiburones esperan.

$$p(f_i \mid e_{a_i}) = \frac{count(f \ \ aligned\text{-}to\ e)}{count(e)}$$

p(el | the) = 0.5
p(Los | the) = 0.5

Any problems or concerns?

Thought Experiment

The old man is happy. He has fished many times.

El viejo está feliz porque ha pescado muchos veces.

His wife talks to him.

Su mujer habla con él.

The sharks await.

Los tiburones esperan.

Getting data like this is expensive!

Even if we had it, what happens when we switch to a new domain/corpus

Thought Experiment #2

The old man is happy. He has fished many times.

80 annotators

El viejo está feliz porque ha pescado muchos veces.

The old man is happy. He has fished many times.

20 annotators

El viejo está feliz porque ha pescado muchos veces.

$$p(f_i \mid e_{a_i}) = \frac{count(f \text{ } aligned\text{-}to \text{ } e)}{count(e)}$$

What do we do?

Thought Experiment #2

The old man is happy. He has fished many times.

El viejo está feliz porque ha pescado muchos veces.

80 annotators

The old man is happy. He has fished many times.

El viejo está feliz porque ha pescado muchos veces.

20 annotators

$$p(f_i \mid e_{a_i}) = \frac{count(f\ aligned\text{-}to\ e)}{count(e)}$$

Use partial counts:

- count(viejo | man) 0.8
- count(viejo | old) 0.2

Training Without Alignments

x y

a b

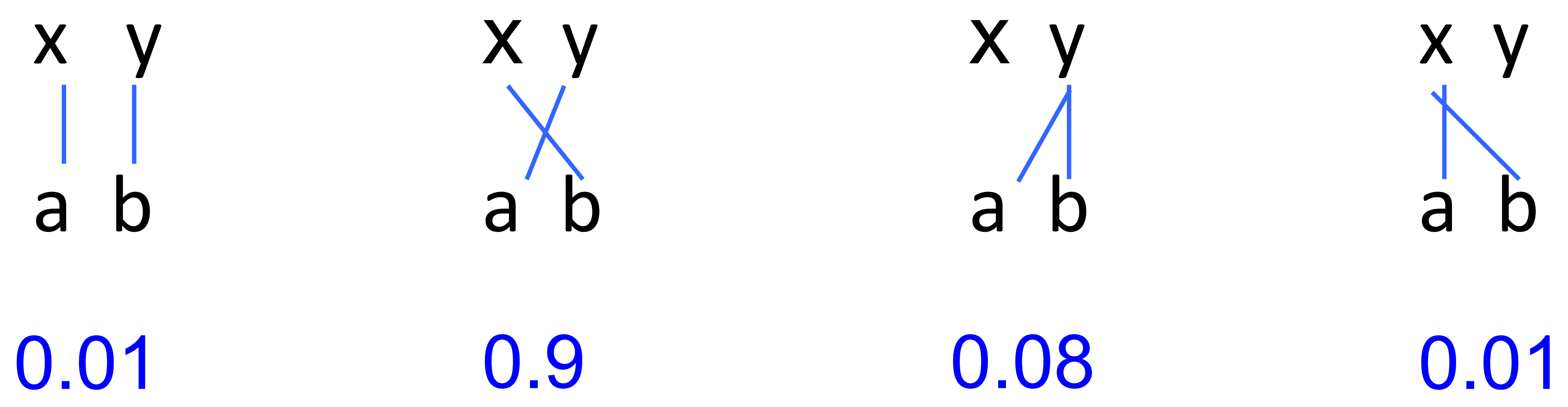
IBM model 1: Each foreign word is aligned to 1 English word (ignore NULL for now)

What are the possible alignments?

Training Without Alignments

IBM model 1: Each foreign word is aligned to 1 English word

Training Without Alignments



IBM model 1: Each foreign word is aligned to 1 English word

If I told you how likely each of these were, does that help us with calculating p(f | e)?

Training Without Alignments

0.01 0.9 0.08 0.01

IBM model 1: Each foreign word is aligned to 1 English word

$$p(f_i \mid e_{a_i}) = \frac{count(f\ aligned\text{-}to\ e)}{count(e)}$$

Use partial counts:

- count(y | a) 0.9+0.01
- count(x | a) 0.01+0.08

On the One Hand

0.01 | 0.9 | 0.08 | 0.01

If you had the likelihood of each alignment, you could calculate p(f|e)

$$p(f_i \mid e_{a_i}) = \frac{count(f \ \ aligned\text{-}to \ e)}{count(e)}$$

On the Other Hand

$$p(F, a_1 a_2 \ldots a_{|F|} \mid E) = \prod_{i=1}^{|F|} p(f_i \mid e_{a_i})$$

If you had p(f|e) could you calculate the probability of the alignments?

$$p(f_i \mid e_{a_i})$$

On the Other Hand

$\boxed{p(x \mid a) * p(y \mid b)}$ $\boxed{p(x \mid b) * p(y \mid a)}$ $\boxed{p(x \mid b) * p(y \mid b)}$ $\boxed{p(x \mid a) * p(y \mid a)}$

$$p(F, a_1 a_2 \ldots a_{|F|} \mid E) = \prod_{i=1}^{|F|} p(f_i \mid e_{a_i})$$

$$p(f_i \mid e_{a_i})$$

How Does this Relate to EM?

Training Without Alignments

Initially assume all p(f|e) are equally probable

Repeat:

- Enumerate all possible alignments
- Calculate how probable the alignments are under the current model (i.e. p(f|e))
  - Recalculate p(f|e) using counts from **all** alignments, **weighted** by how probable they are

EM Alignment

E-step

- Enumerate all possible alignments
- Calculate how probable the alignments are under the current model (i.e. $p(f|e)$)

M-step

- Recalculate $p(f|e)$ using counts from **all** alignments, **weighted** by how probable they are green house

casa verde

the house

la casa

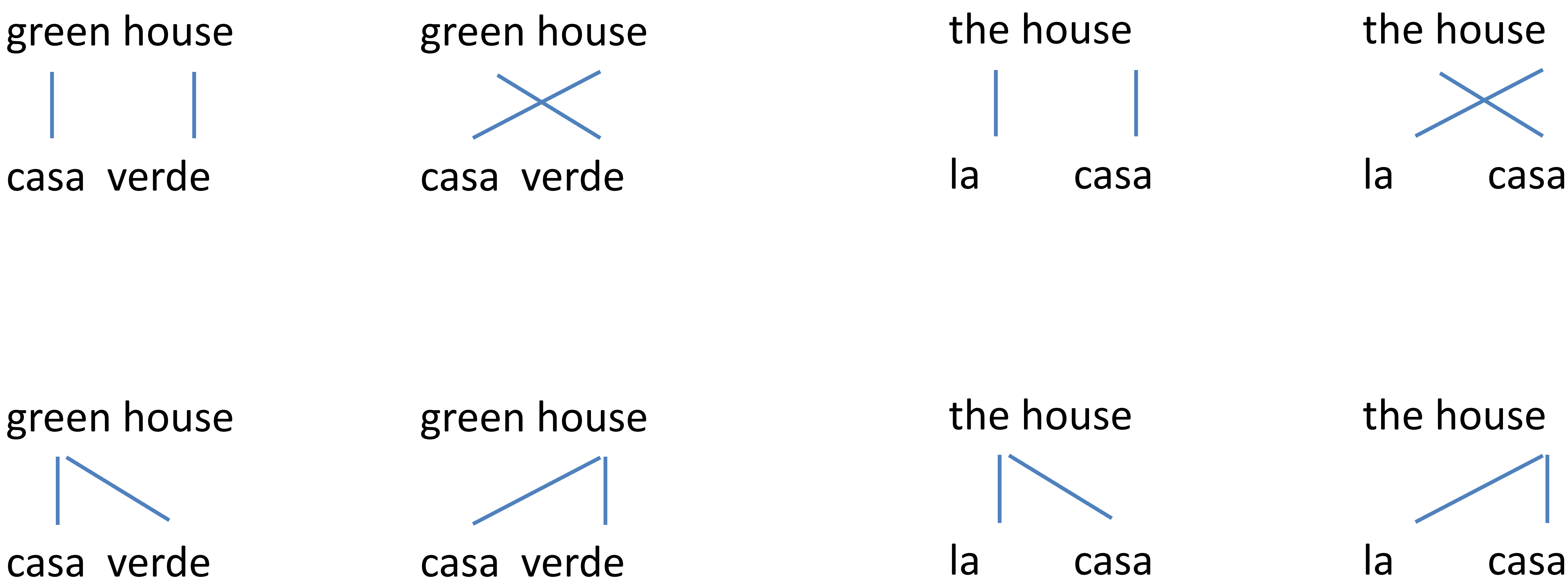
What are the different p(f|e) that make up my model?

p(casa | green)	
p(verde | green)	
p(la | green)	

p(casa | house)	
p(verde | house)	
p(la | house)	

p(casa | the)	
p(verde | the)	
p(la | the)	

Technically, all combinations of foreign and English words

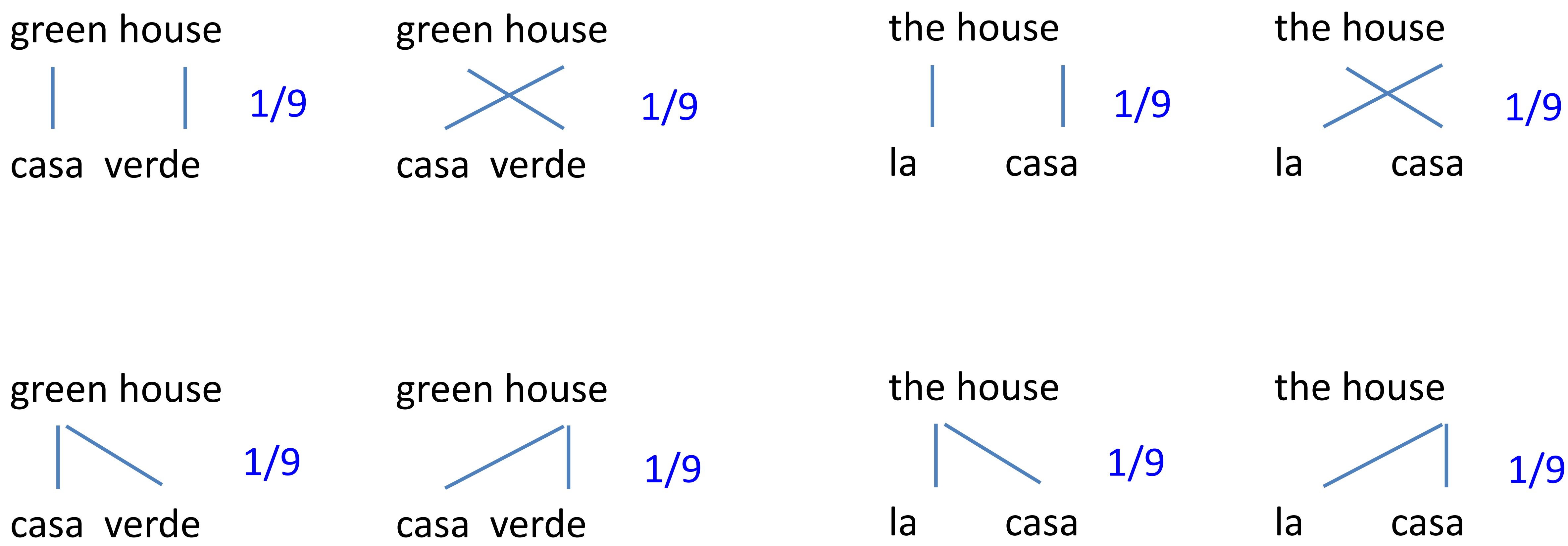


p(casa | green)	***1/3***
p(verde | green)	***1/3***
p(la | green)	***1/3***

p(casa | house)	***1/3***
p(verde | house)	***1/3***
p(la | house)	***1/3***

p(casa | the)	***1/3***
p(verde | the)	***1/3***
p(la | the)	***1/3***

Start with all p(f|e) equally probable



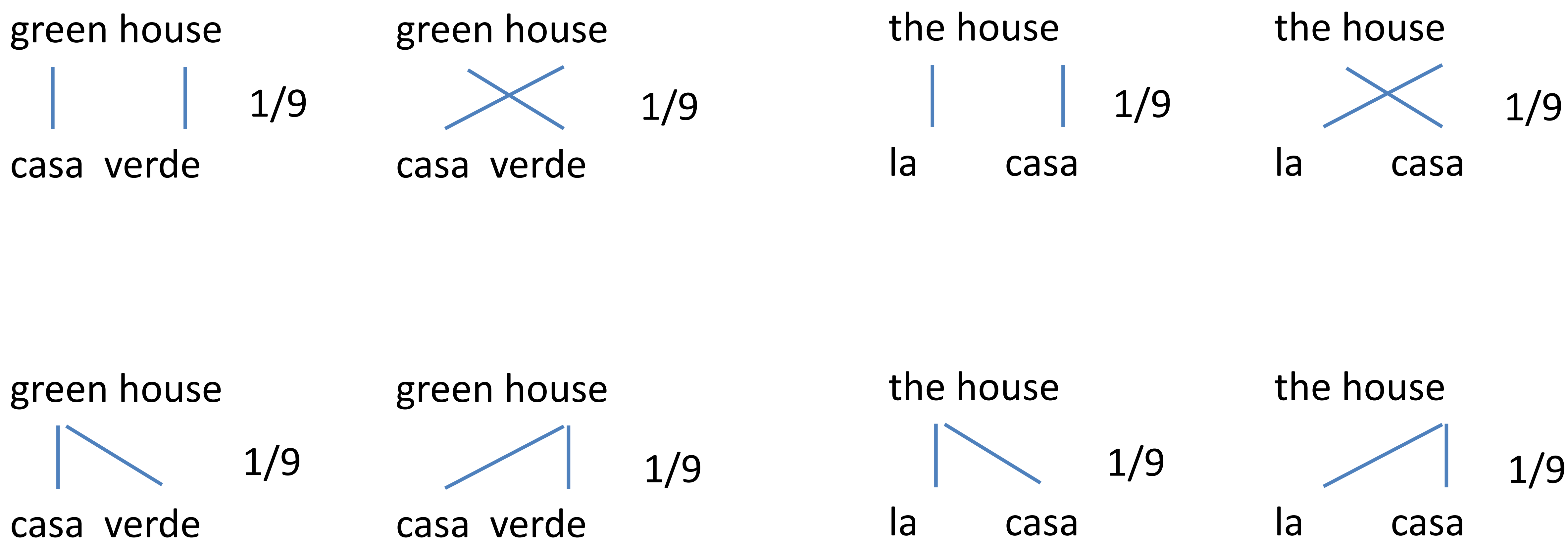
p(casa | green)	*1/3*
p(verde | green)	*1/3*
p(la | green)	*1/3*

p(casa | house)	*1/3*
p(verde | house)	*1/3*
p(la | house)	*1/3*

p(casa | the)	*1/3*
p(verde | the)	*1/3*
p(la | the)	*1/3*

E-step: What are the probabilities of the alignments?

$$p(f_1 f_2 \ldots f_{|F|}, a_1 a_2 \ldots a_{|F|} \mid e_1 e_2 \ldots e_{|E|}) = \prod_{i=1}^{|F|} p(f_i \mid e_{a_i})$$



M-step: What are the p(f|e) given the alignments?

p(casa | green)	***1/3***
p(verde | green)	***1/3***
p(la | green)	***1/3***

p(casa | house)	***1/3***
p(verde | house)	***1/3***
p(la | house)	***1/3***

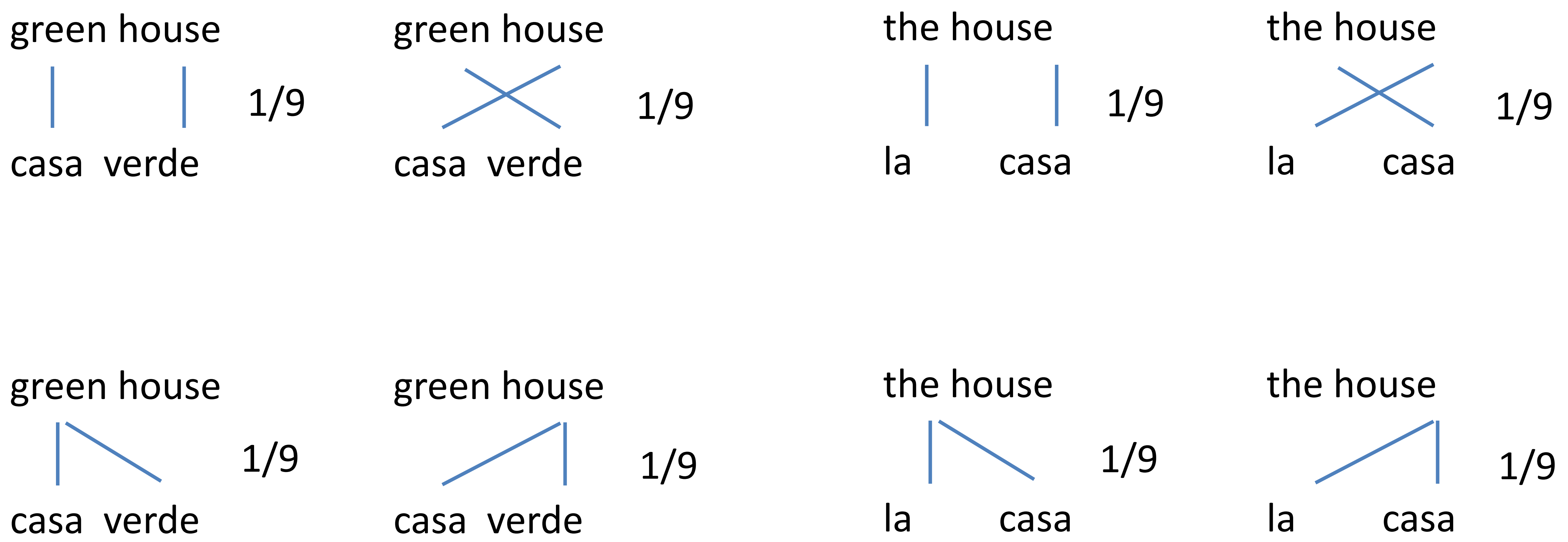
p(casa | the)	***1/3***
p(verde | the)	***1/3***
p(la | the)	***1/3***

c(casa,green) = ?
c(verde,green) = ?
c(la, green) = ?

c(casa,house) ?
c(verde,house) = ?
c(la,house) = ?

c(casa,the) = ?
c(verde,the) = ?
c(la,the) = ?

First, calculate the partial counts



M-step: What are the p(f|e) given the alignments?

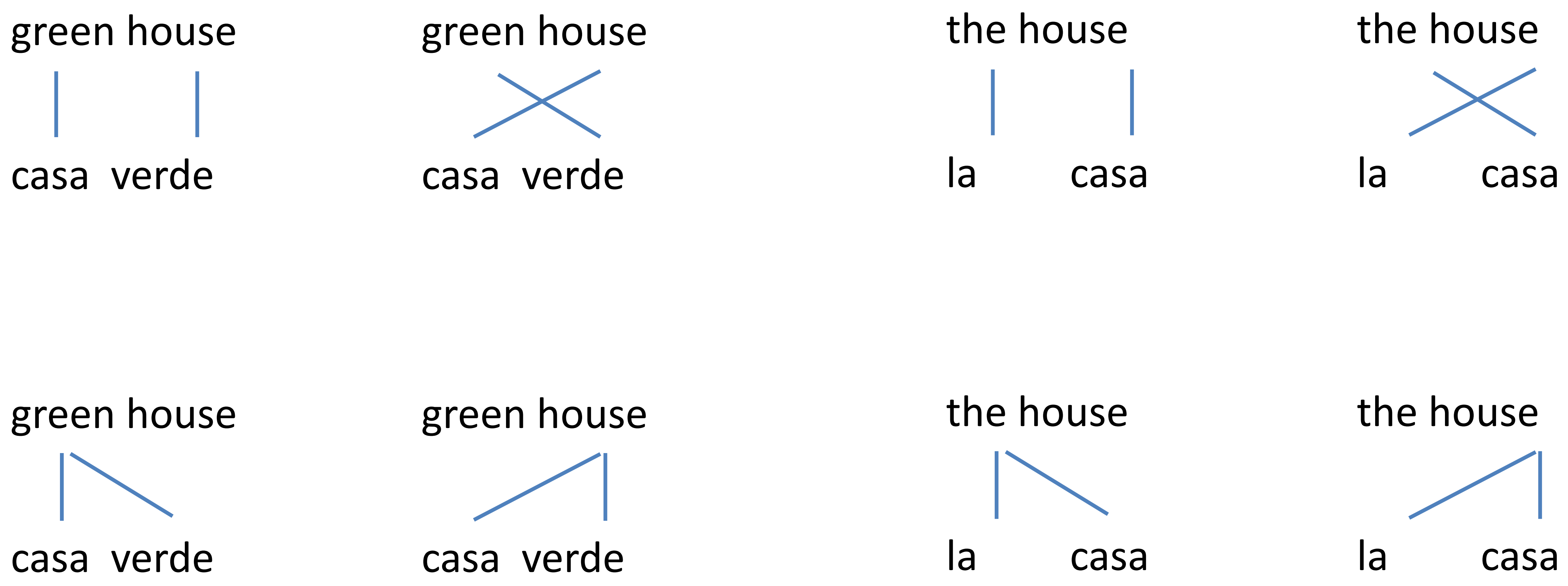
p(casa | green)	***?***	***p(casa | house)***	***?***	***p(casa | the)***	***?***
p(verde | green)	***?***	***p(verde | house)***	***?***	***p(verde | the)***	***?***
p(la | green)	***?***	***p(la | house)***	***?***	***p(la | the)***	***?***

c(casa,green) = 1/9+1/9 = 2/9
c(verde,green) = 1/9+1/9 = 2/9
c(la, green) = 0

c(casa,house) = 1/9+1/9+ 1/9+1/9 = 4/9
c(verde,house) = 1/9+1/9 = 2/9
c(la,house) = 1/9+1/9 = 2/9

c(casa,the) = 1/9+1/9 = 2/9
c(verde,the) = 0
c(la,the) = 1/9+1/9 = 2/9

Then, calculate the probabilities by normalizing the counts



p(casa | green)	***1/2***
p(verde | green)	***1/2***
p(la | green)	***0***

p(casa | house)	***1/2***
p(verde | house)	***1/4***
p(la | house)	***1/4***

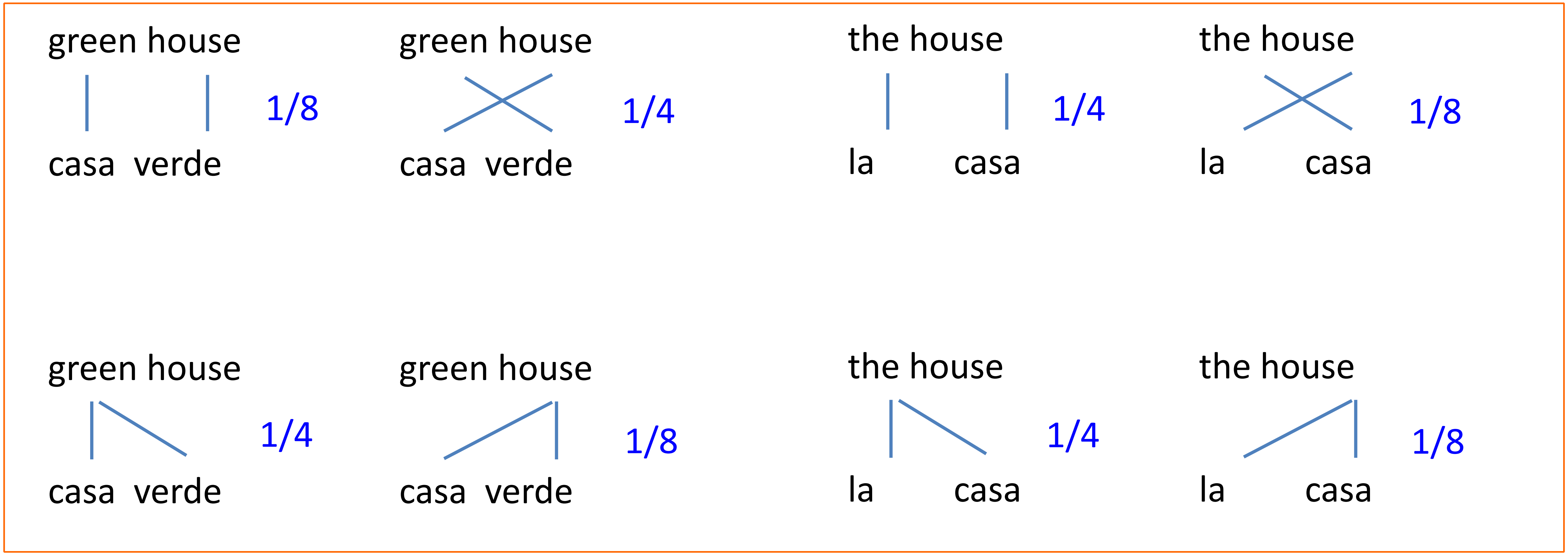
p(casa | the)	***1/2***
p(verde | the)	***0***
p(la | the)	***1/2***

c(casa,green) = 1/9+1/9 = 2/9
c(verde,green) = 1/9+1/9 = 2/9
c(la, green) = 0

c(casa,house) = 1/9+1/9+
1/9+1/9 = 4/9
c(verde,house) = 1/9+1/9 = 2/9
c(la,house) = 1/9+1/9 = 2/9

c(casa,the) = 1/9+1/9 = 2/9
c(verde,the) = 0
c(la,the) = 1/9+1/9 = 2/9

E-step: What are the probabilities of the alignments?



p(casa | green)	***1/2***
p(verde | green)	***1/2***
p(la | green)	***0***

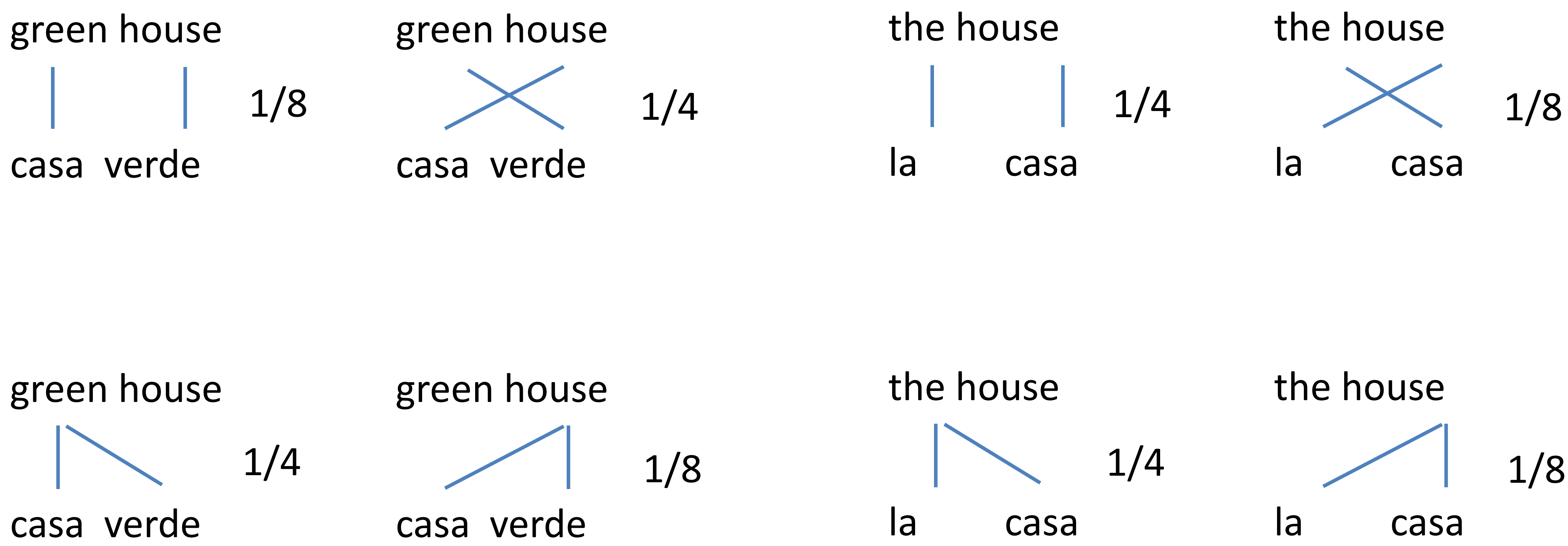
c(casa,green) = 1/9+1/9 = 1/3
c(verde,green) = 1/9+1/9 = 1/3
c(la, green) = 0

p(casa | house)	***1/2***
p(verde | house)	***1/4***
p(la | house)	***1/4***

c(casa,house) = 1/9+1/9+
1/9+1/9 = 2/3
c(verde,house) = 1/9+1/9 = 1/3
c(la,house) = 1/9+1/9 = 1/3

p(casa | the)	***1/2***
p(verde | the)	***0***
p(la | the)	***1/2***

c(casa,the) = 1/9+1/9 = 1/3
c(verde,the) = 0
c(la,the) = 1/9+1/9 = 1/3



M-step: What are the p(f|e) given the alignments?

p(casa | green)	***1/2***
p(verde | green)	***1/2***
p(la | green)	***0***

p(casa | house)	***1/2***
p(verde | house)	***1/4***
p(la | house)	***1/4***

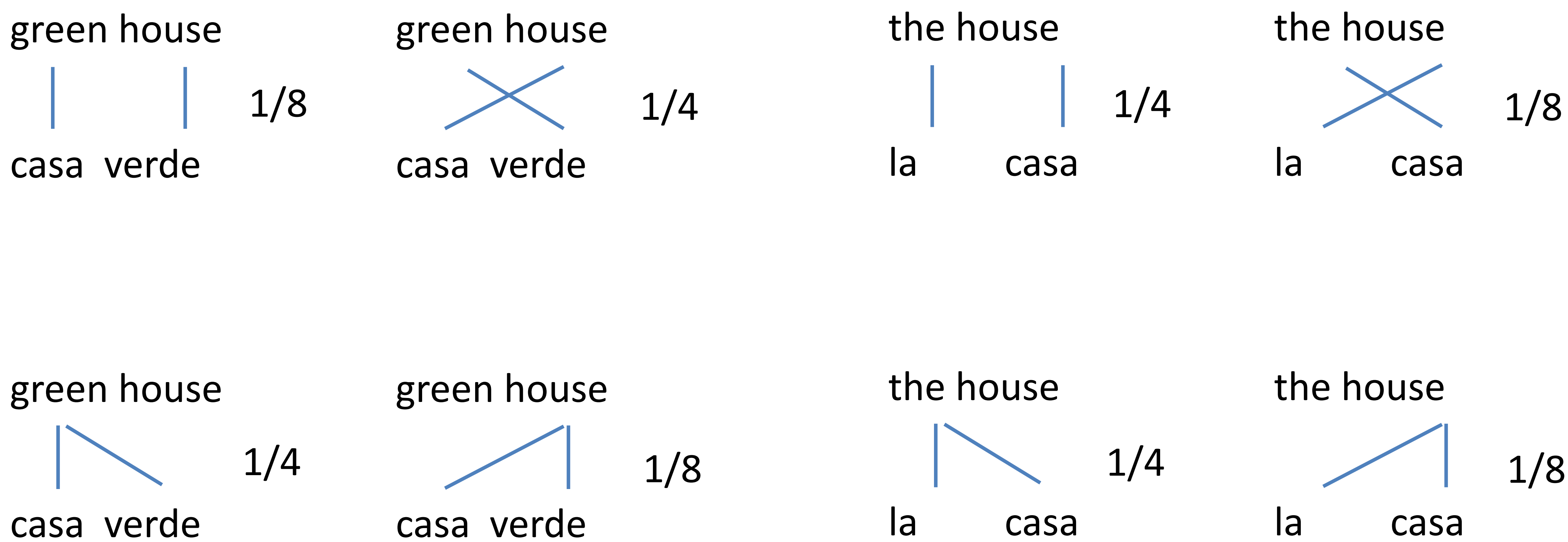
p(casa | the)	***1/2***
p(verde | the)	***0***
p(la | the)	***1/2***

c(casa,green) = ?
c(verde,green) = ?
c(la, green) = ?

c(casa,house) = ?
c(verde,house) = ?
c(la,house) = ?

c(casa,the) = ?
c(verde,the) = ?
c(la,the) = ?

First, calculate the partial counts



p(casa | green)	***1/2***
p(verde | green)	***1/2***
p(la | green)	***0***

p(casa | house)	***1/2***
p(verde | house)	***1/4***
(la | house)	***1/4***

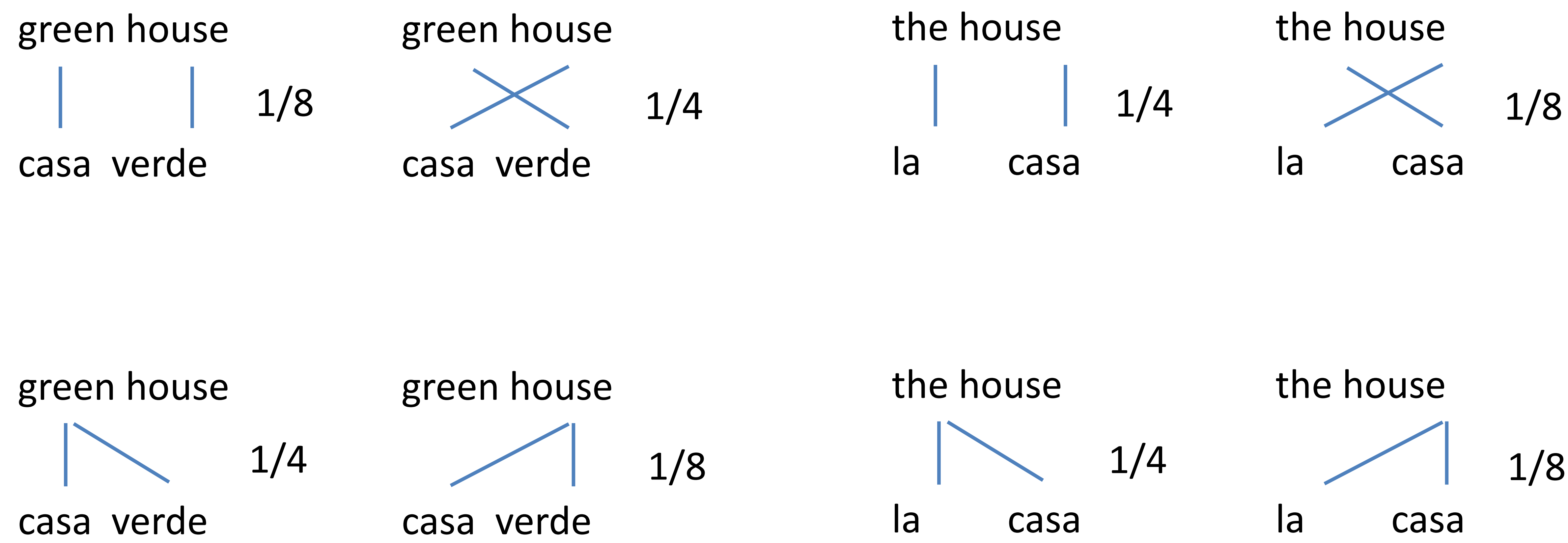
p(casa | the)	***1/2***
p(verde | the)	***0***
p(la | the)	***1/2***

c(casa,green) = 1/8+1/4 = 3/8
c(verde,green) = 1/4+1/4 = 1/2
c(la, green) = 0

c(casa,house) = 1/4+1/8+
1/4+1/8 = 3/4
c(verde,house) = 1/8+1/8 = 1/4
c(la,house) = 1/8+1/8 = 1/4

c(casa,the) = 1/8+1/4 = 3/8
c(verde,the) = 0
c(la,the) = 1/4+1/4 = 1/2

Then, calculate the probabilities by normalizing the counts



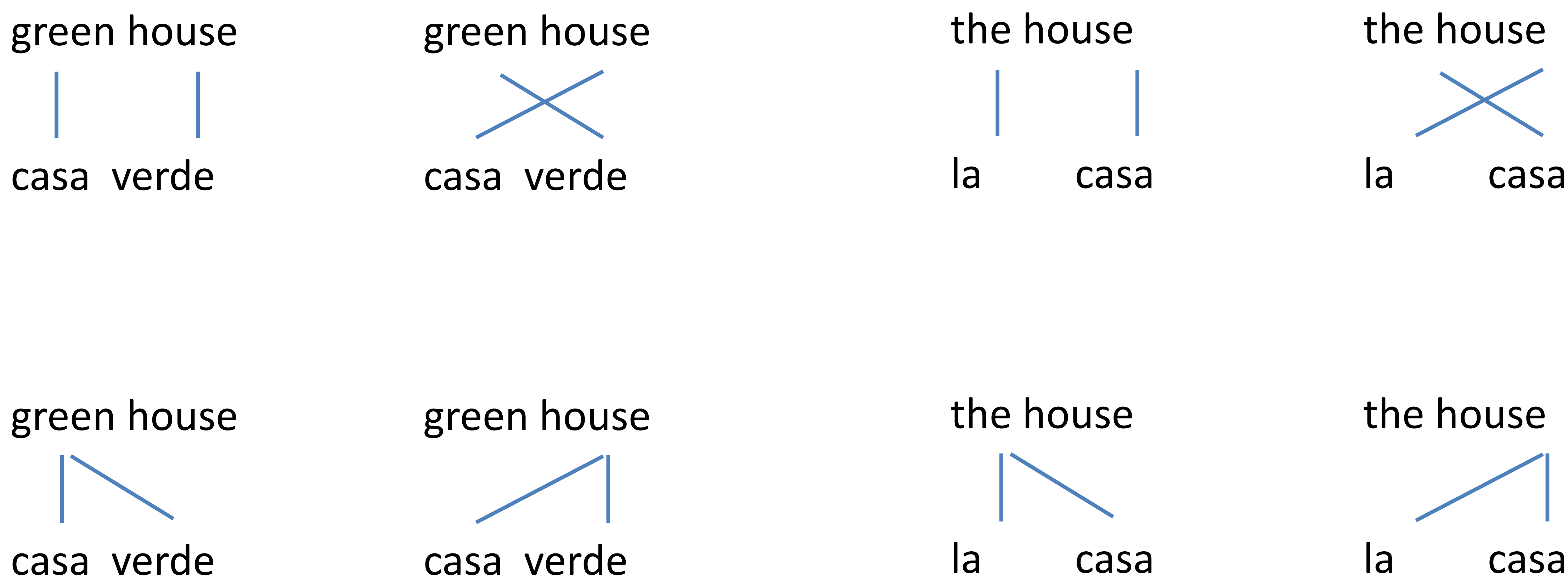
M-step: What are the p(f|e) given the alignments?

p(casa | green)	***?***	***p(casa | house)***	***?***	***p(casa | the)***	***?***
p(verde | green)	***?***	***p(verde | house)***	***?***	***p(verde | the)***	***?***
p(la | green)	***?***	***p(la | house)***	***?***	***p(la | the)***	***?***

c(casa,green) = 1/8+1/4 = 3/8
c(verde,green) = 1/4+1/4 = 1/2
c(la, green) = 0

c(casa,house) = 1/4+1/8+ 1/4+1/8 = 3/4
c(verde,house) = 1/8+1/8 = 1/4
c(la,house) = 1/8+1/8 = 1/4

c(casa,the) = 1/8+1/4 = 3/8
c(verde,the) = 0
c(la,the) = 1/4+1/4 = 1/2



p(casa | green)	***3/7***
p(verde | green)	***4/7***
p(la | green)	***0***

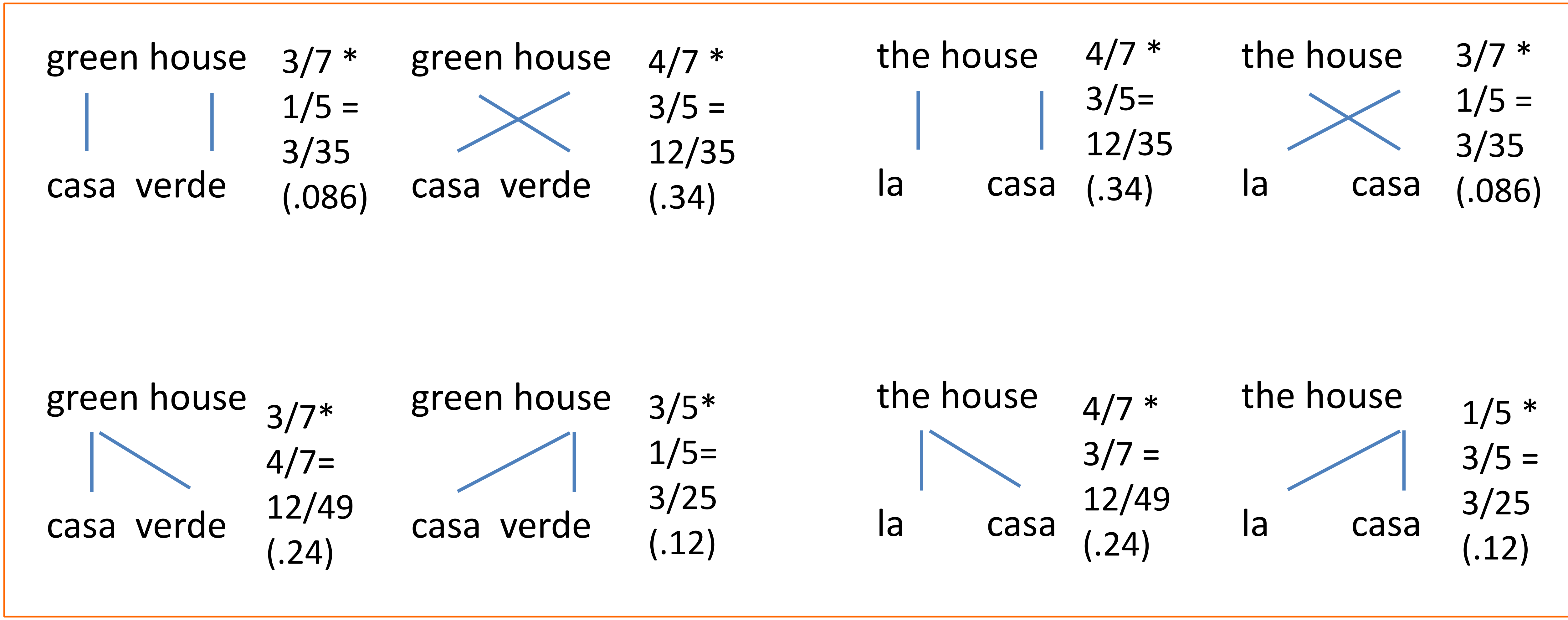
p(casa | house)	***3/5***
p(verde | house)	***1/5***
p(la | house)	***1/5***

p(casa | the)	***3/7***
p(verde | the)	***0***
p(la | the)	***4/7***

c(casa,green) = 1/8+1/4 = 3/8
c(verde,green) = 1/4+1/4 = 1/2
c(la, green) = 0

c(casa,house) = 1/4+1/8+
1/4+1/8 = 3/4
c(verde,house) = 1/8+1/8 = 1/4
c(la,house) = 1/8+1/8 = 1/4

c(casa,the) = 1/8+1/4 = 3/8
c(verde,the) = 0
c(la,the) = 1/4+1/4 = 1/2



p(casa | green)	***3/7***
p(verde | green)	***4/7***
p(la | green)	***0***

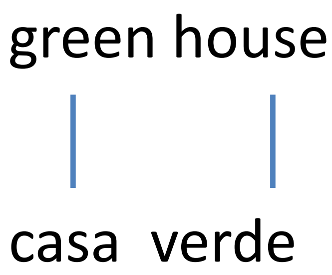
c(casa,green) = 1/8+1/4 = 3/8
c(verde,green) = 1/4+1/4 = 1/2
c(la, green) = 0

p(casa | house)	***3/5***
p(verde | house)	***1/5***
p(la | house)	***1/5***

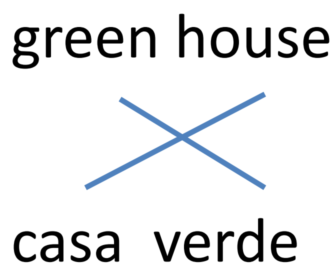
c(casa,house) = 1/4+1/8+ 1/4+1/8 = 3/4
c(verde,house) = 1/8+1/8 = 1/4
c(la,house) = 1/8+1/8 = 1/4

p(casa | the)	***3/7***
p(verde | the)	***0***
p(la | the)	***4/7***

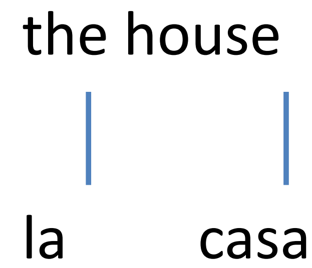
c(casa,the) = 1/8+1/4 = 3/8
c(verde,the) = 0
c(la,the) = 1/4+1/4 = 1/2



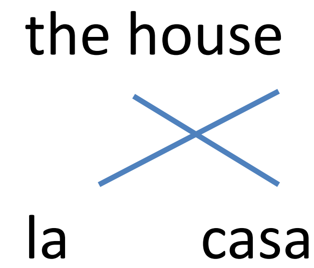
3/7 *
1/5 =
3/35
(.086)



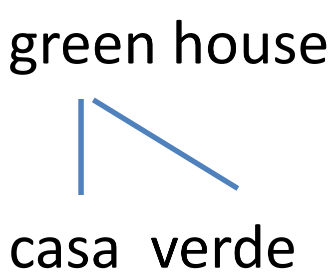
4/7 *
3/5 =
12/35
(.343)



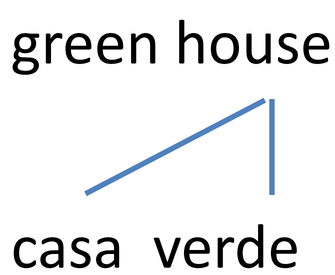
4/7 *
3/5=
12/35
(.343)



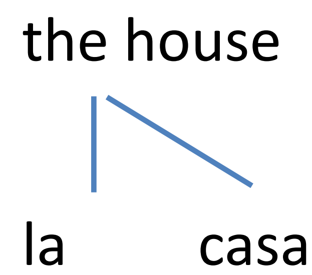
3/7 *
1/5 =
3/35
(.086)



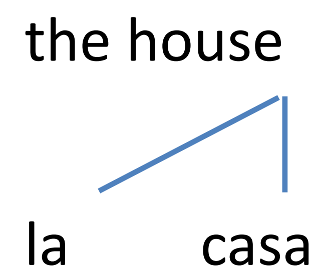
3/7*
4/7=
12/49
(.245)



3/5*
1/5=
3/25
(.12)



4/7 *
3/7 =
12/49
(.245)



1/5 *
3/5 =
3/25
(.12)

p(casa | green)	***3/7***
p(verde | green)	***4/7***
p(la | green)	***0***

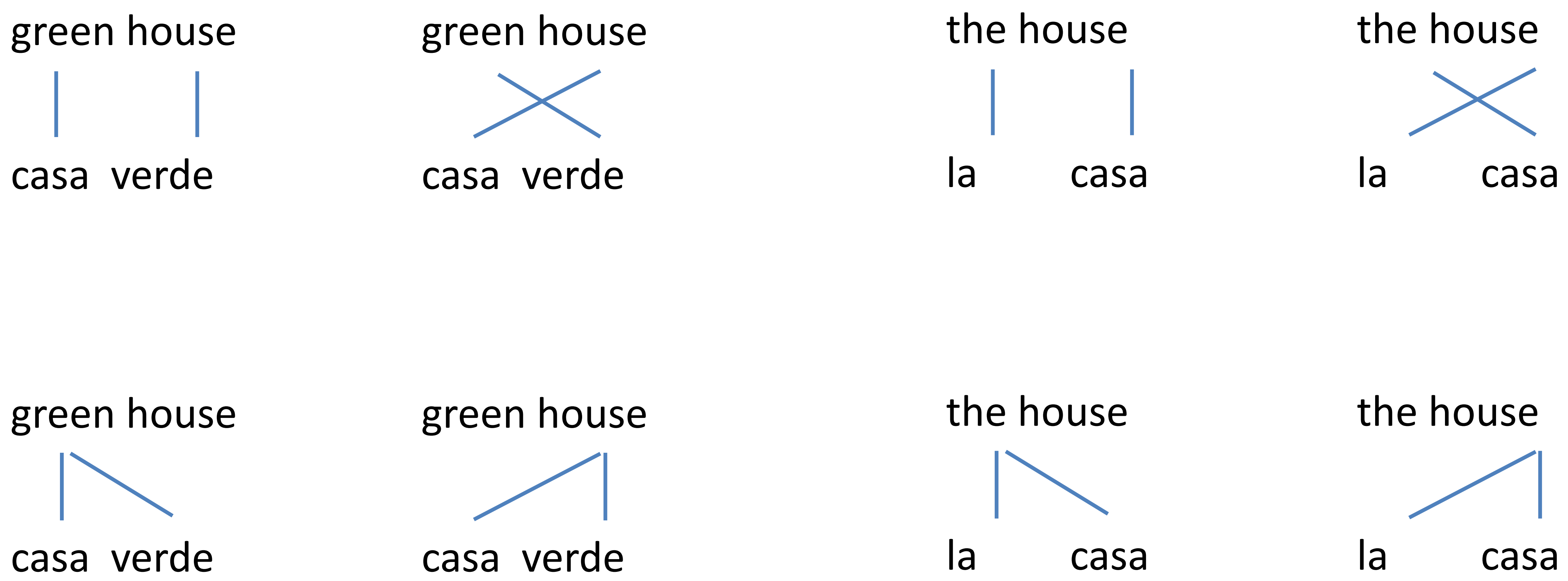
p(casa | house)	***3/5***
p(verde | house)	***1/5***
p(la | house)	***1/5***

p(casa | the)	***3/7***
p(verde | the)	***0***
p(la | the)	***4/7***

c(casa,green) = .086+.245=0.331
c(verde,green) = .343+0.245 = 0.588
c(la, green) = 0

c(casa,house) = .343+.12+
.343+.12=0.926
c(verde,house) = .086+.12=0.206
c(la,house) = .086+.12=0.206

c(casa,the) = .086+.245=0.331
c(verde,the) = 0
c(la,the) = .343+.245=0.588



p(casa | green)	***0.36***
p(verde | green)	***0.64***
p(la | green)	***0***

p(casa | house)	***0.69***
p(verde | house)	***0.15***
p(la | house)	***0.15***

p(casa | the)	***0.36***
p(verde | the)	***0***
p(la | the)	***0.64***

c(casa,green) = .086+.245=0.331
c(verde,green) = .343+0.245 = 0.588
c(la, green) = 0

c(casa,house) = .343+.12+
.343+.12=0.926
c(verde,house) = .086+.12=0.206
c(la,house) = .086+.12=0.206

c(casa,the) = .086+.245=0.331
c(verde,the) = 0
c(la,the) = .343+.245=0.588

Iterate...

5 iterations

p(casa | green)	*0.24*
p(verde | green)	*0.76*
p(la | green)	*0*

p(casa | house)	*0.84*
p(verde | house)	*0.08*
p(la | house)	*0.08*

p(casa | the)	*0.24*
p(verde | the)	*0*
p(la | the)	*0.76*

10 iterations

p(casa | green)	*0.1*
p(verde | green)	*0.9*
p(la | green)	*0*

p(casa | house)	*0.98*
p(verde | house)	*0.01*
p(la | house)	*0.01*

p(casa | the)	*0.1*
p(verde | the)	*0*
p(la | the)	*0.9*

100 iterations

p(casa | green)	*0.005*
p(verde | green)	*0.995*
p(la | green)	*0*

p(casa | house)	*~1.0*
p(verde | house)	*~0.0*
p(la | house)	*~0.0*

p(casa | the)	*0.005*
p(verde | the)	*0*
p(la | the)	*0.995*