Alignment by Agreement

Percy Liang, Ben Taskar, Dan Klein

UC Berkeley

Computer Science Division

Unsupervised word alignment

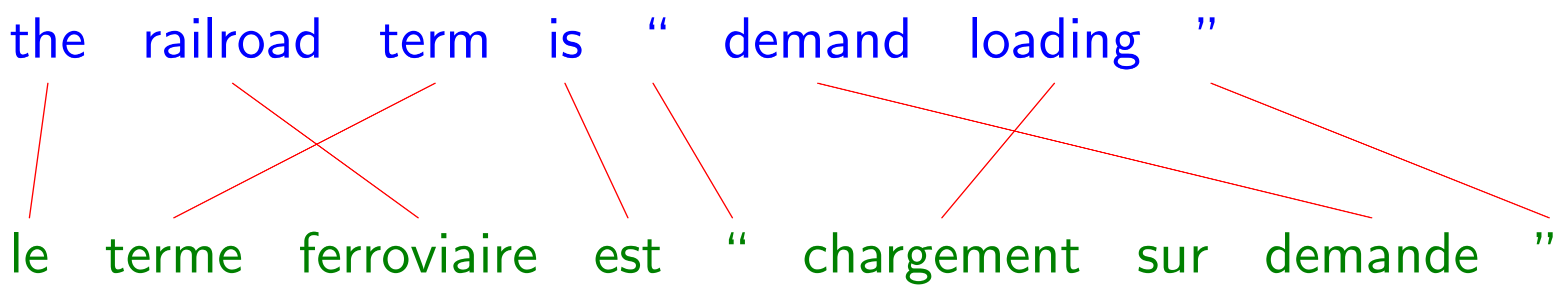
Goal: learn to map sentence pairs to alignments

the railroad term is “ demand loading ”

le terme ferroviaire est “ chargement sur demande ”

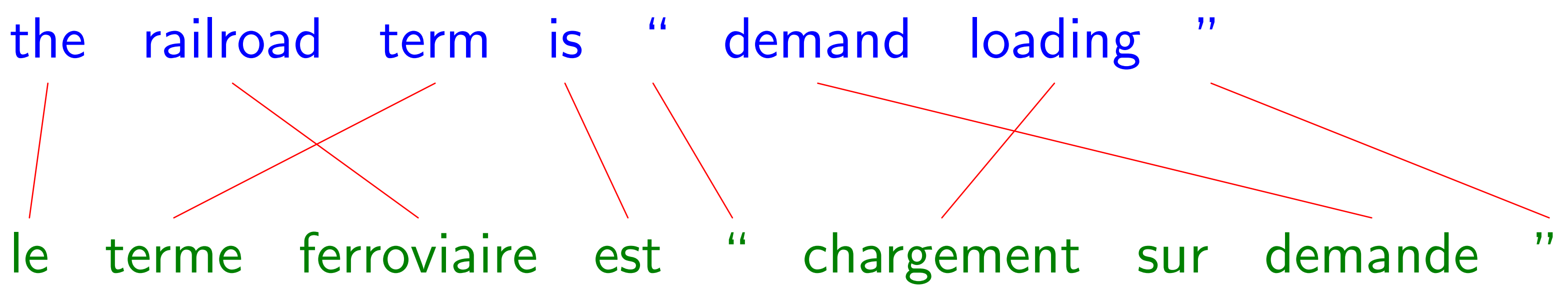
Unsupervised word alignment

Goal: learn to map sentence pairs to alignments



Unsupervised word alignment

Goal: learn to map sentence pairs to alignments



Approach:

jointly train two models to encourage *agreement*

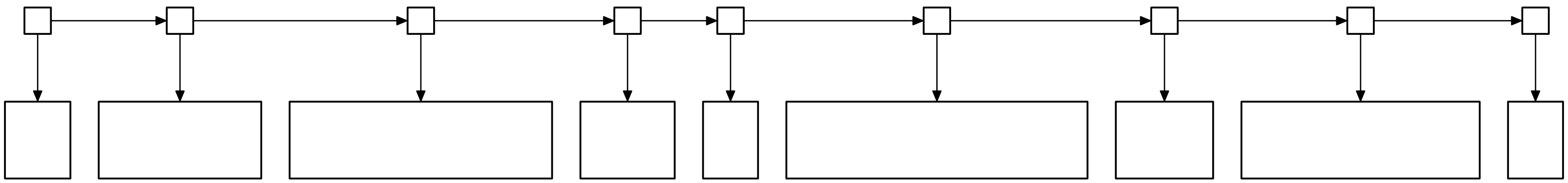
HMM model [Ney, Vogel '96]

Generative model: $p(\mathbf{a}, \mathbf{e}, \mathbf{f}; \theta)$

HMM model [Ney, Vogel '96]

Generative model: $p(\mathbf{a}, \mathbf{e}, \mathbf{f}; \theta)$

$p(\mathbf{e})$ the railroad term is " demand loading "

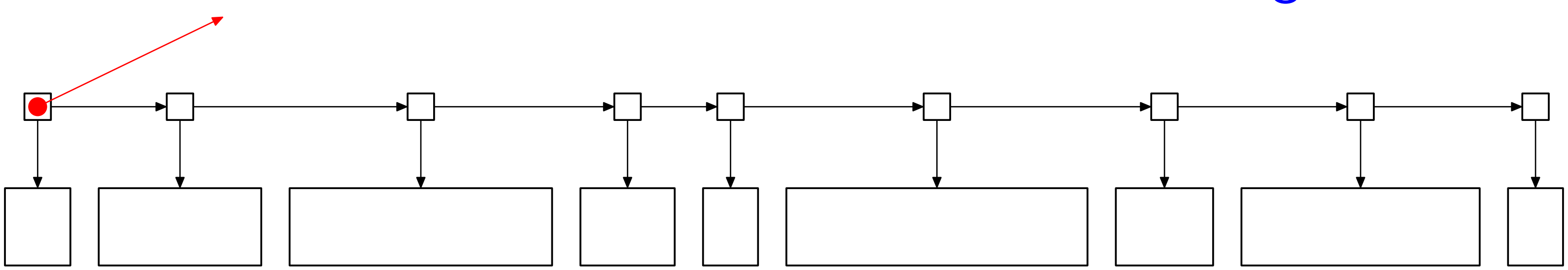


HMM model [Ney, Vogel '96]

Generative model: $p(\mathbf{a}, \mathbf{e}, \mathbf{f}; \theta)$

$p(\mathbf{e})$

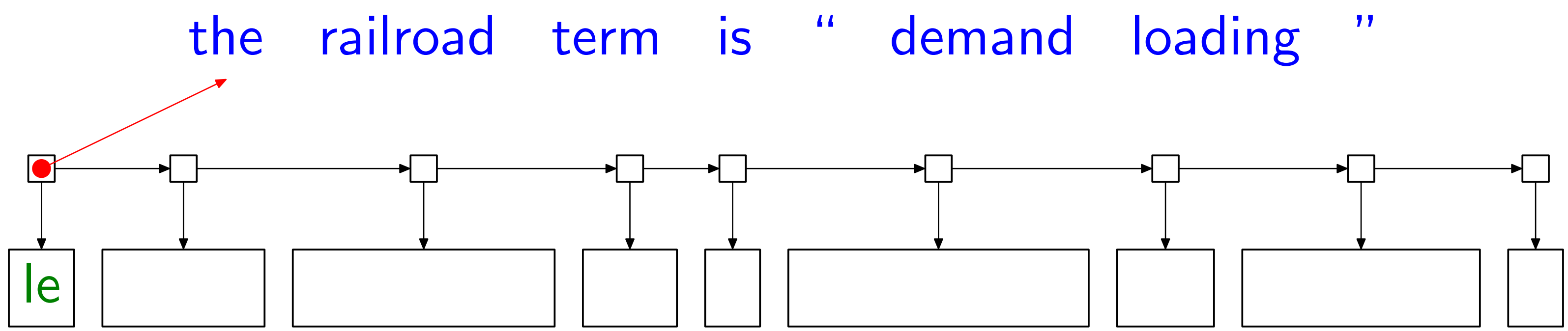
the railroad term is " demand loading "



HMM model [Ney, Vogel '96]

Generative model: $p(\mathbf{a}, \mathbf{e}, \mathbf{f}; \theta)$

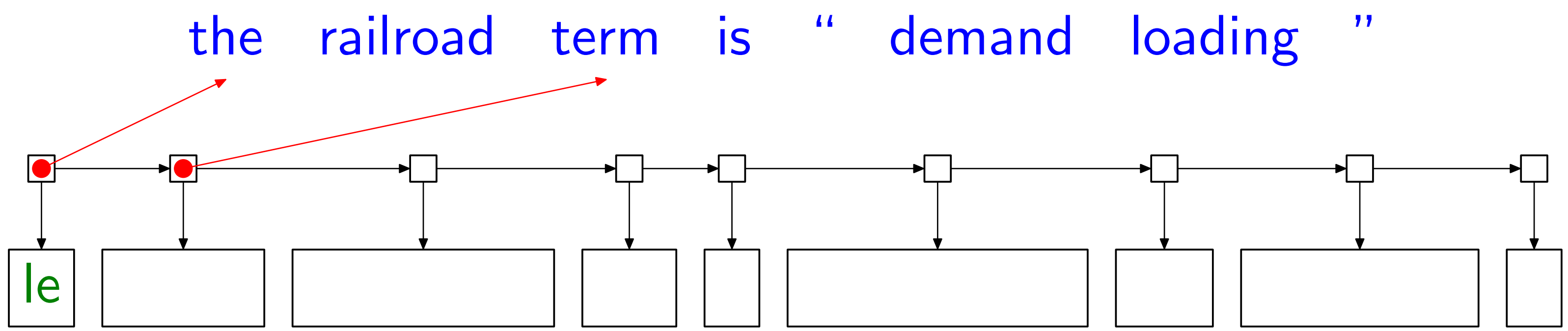
$p(\mathbf{e})$



HMM model [Ney, Vogel '96]

Generative model: $p(\mathbf{a}, \mathbf{e}, \mathbf{f}; \theta)$

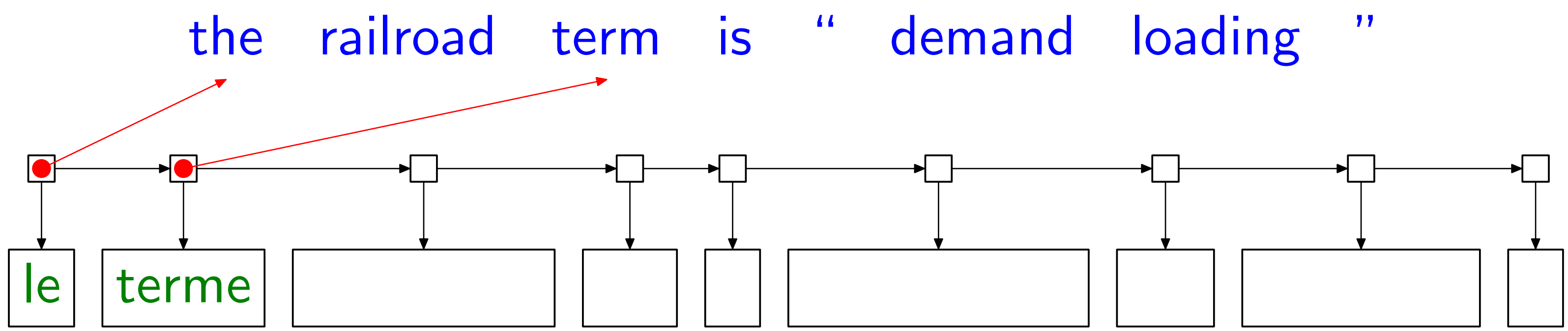
$p(\mathbf{e})$



HMM model [Ney, Vogel '96]

Generative model: $p(\mathbf{a}, \mathbf{e}, \mathbf{f}; \theta)$

$p(\mathbf{e})$

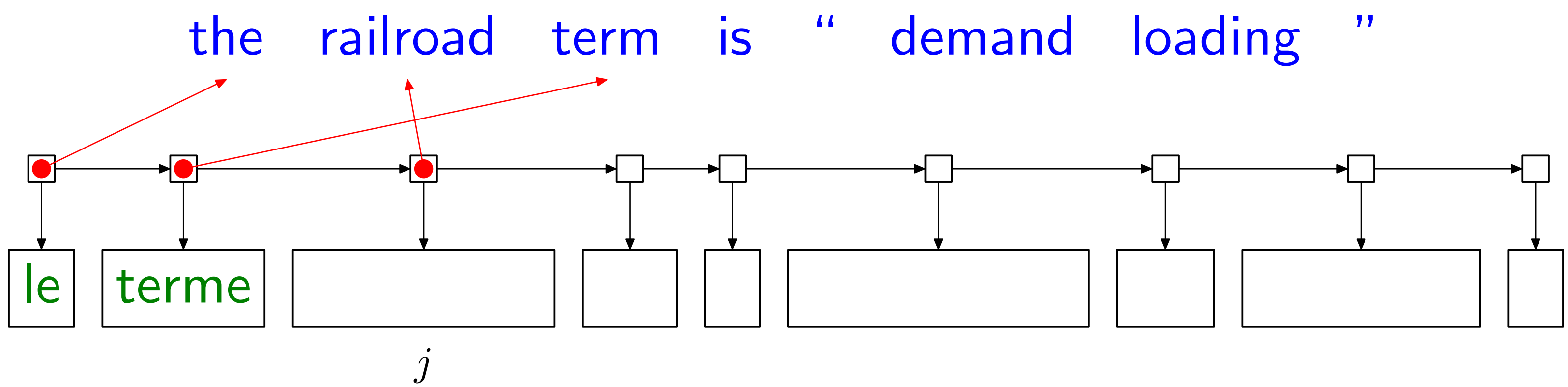


HMM model [Ney, Vogel '96]

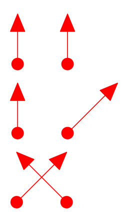
Generative model: $p(\mathbf{a}, \mathbf{e}, \mathbf{f}; \theta)$

$p(\mathbf{e})$

$p(\mathsf{a}_j \mid \mathsf{a}_{j-}; \theta_{\mathsf{d}})$



Distortion θ_{d}



$p(\quad) = 0.6$

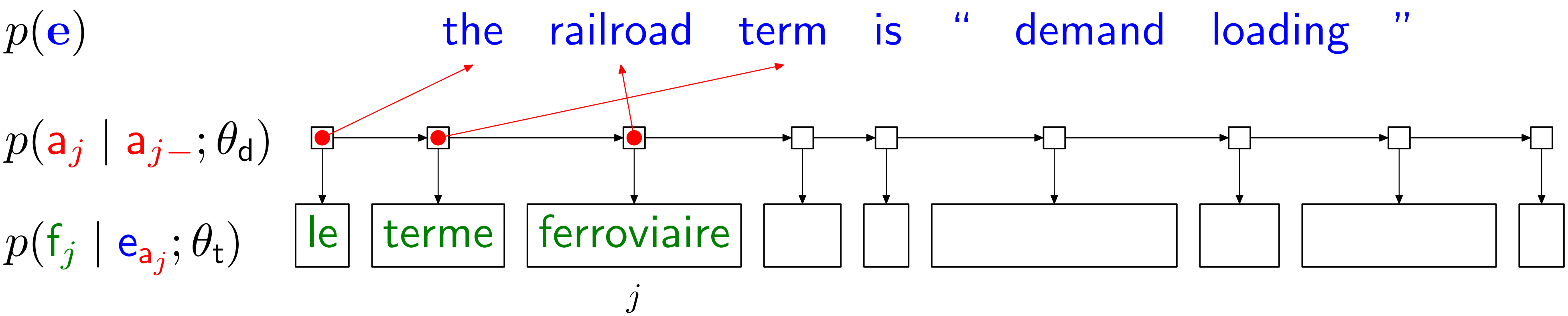
$p(\quad) = 0.2$

$p(\quad) = \mathbf{0.1}$

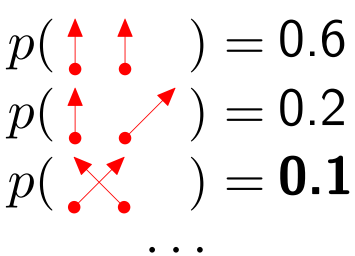
$\ldots$

HMM model [Ney, Vogel '96]

Generative model: $p(\mathbf{a}, \mathbf{e}, \mathbf{f}; \theta)$



Distortion θ_{d}



Translation θ_{t}

$p($ the $\rightarrow$ le $) = 0.53$

$p($ the $\rightarrow$ la $) = 0.24$

$p($ **railroad** $\rightarrow$ **ferroviaire** $) = \mathbf{0.19}$

$p($ NULL $\rightarrow$ le $) = 0.12$

...

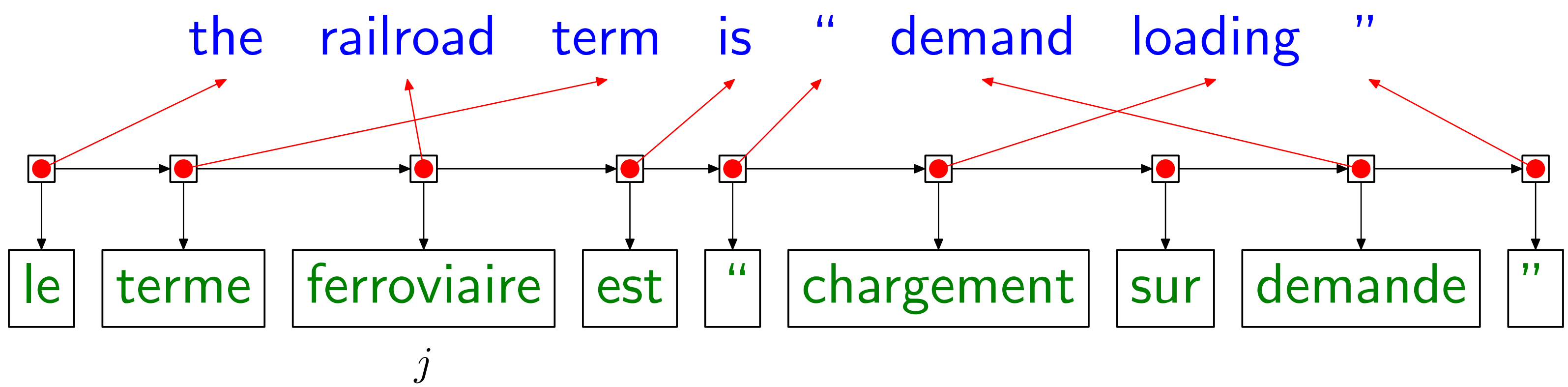
HMM model [Ney, Vogel '96]

Generative model: $p(\mathbf{a}, \mathbf{e}, \mathbf{f}; \theta)$

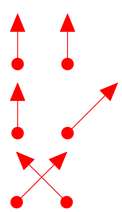
$p(\mathbf{e})$

$p(\mathsf{a}_j \mid \mathsf{a}_{j-}; \theta_\mathsf{d})$

$p(\mathsf{f}_j \mid \mathsf{e}_{\mathsf{a}_j}; \theta_\mathsf{t})$



Distortion θ_d



$p(\ \ ) = 0.6$

$p(\ \ ) = 0.2$

$p(\ \ ) = \mathbf{0.1}$

$\ldots$

Translation θ_t

$p($ the $\rightarrow$ le $) = 0.53$

$p($ the $\rightarrow$ la $) = 0.24$

$p($ **railroad** $\rightarrow$ **ferroviaire** $) = \mathbf{0.19}$

$p($ NULL $\rightarrow$ le $) = 0.12$

$\ldots$

EM training

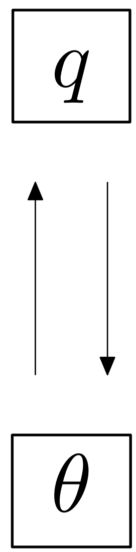
Maximize $p(\mathbf{e}, \mathbf{f}; \theta)$

EM training

Maximize $p(\mathbf{e}, \mathbf{f}; \theta)$

Parameters: θ

Expectation over alignments: q



E-step:

$q(\mathbf{a} \mid \mathbf{e}, \mathbf{f}) := p(\mathbf{a} \mid \mathbf{e}, \mathbf{f}; \theta)$

(forward-backward)

M-step:

$\theta := \mathrm{argmax}_\theta \, \mathbb{E}_q \log p(\mathbf{a}, \mathbf{e}, \mathbf{f} \mid \theta)$

(normalizing counts)

EM training

Maximize $p(\mathbf{e}, \mathbf{f}; \theta)$

Parameters: θ

Expectation over alignments: q

q

E-step:

$q(\mathbf{a} \mid \mathbf{e}, \mathbf{f}) := p(\mathbf{a} \mid \mathbf{e}, \mathbf{f}; \theta)$

(forward-backward)

M-step:

$\theta := \text{argmax}_\theta \, \mathbb{E}_q \log p(\mathbf{a}, \mathbf{e}, \mathbf{f} \mid \theta)$

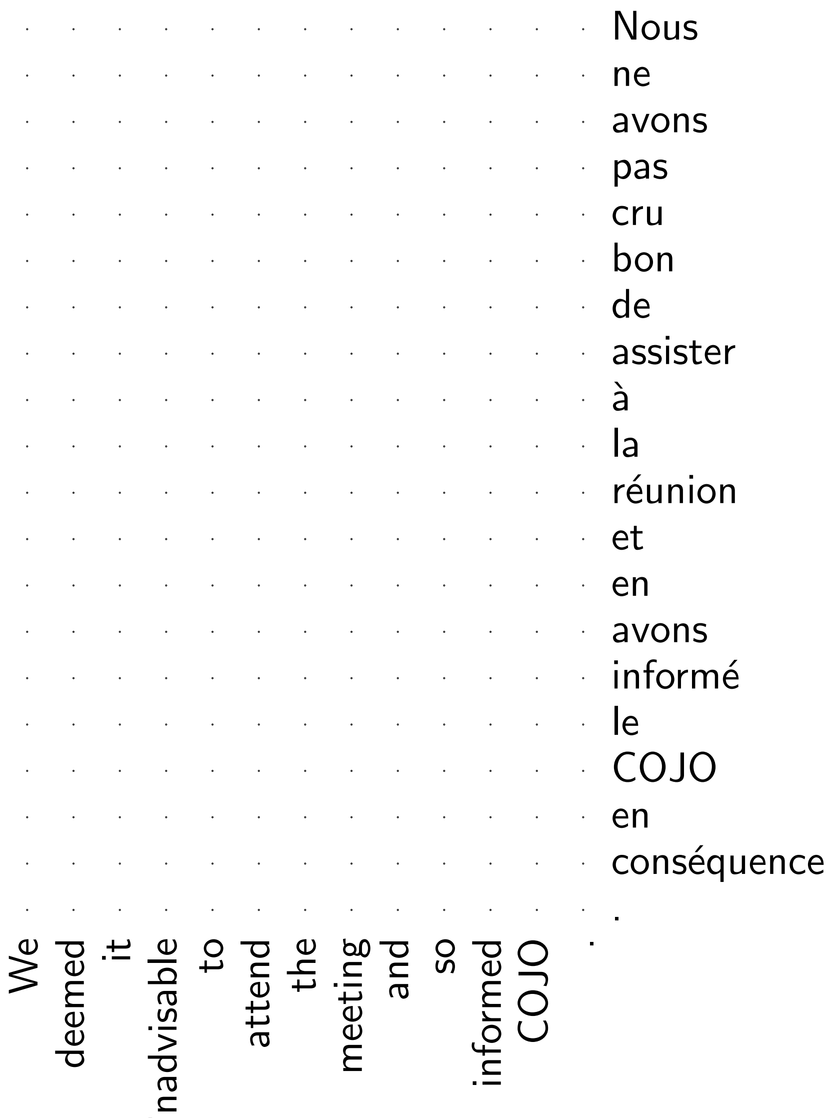
(normalizing counts)

θ

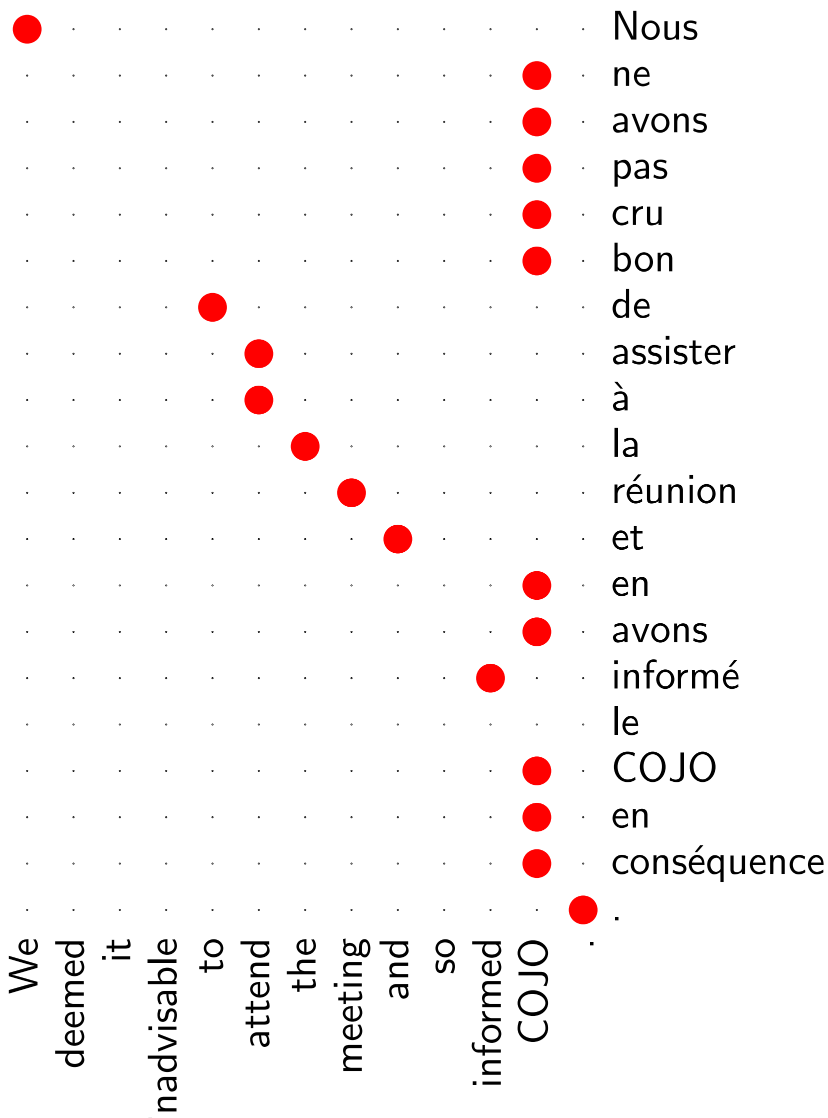
$\leftarrow \mathbf{e}, \mathbf{f}$

$\mathbf{a}$

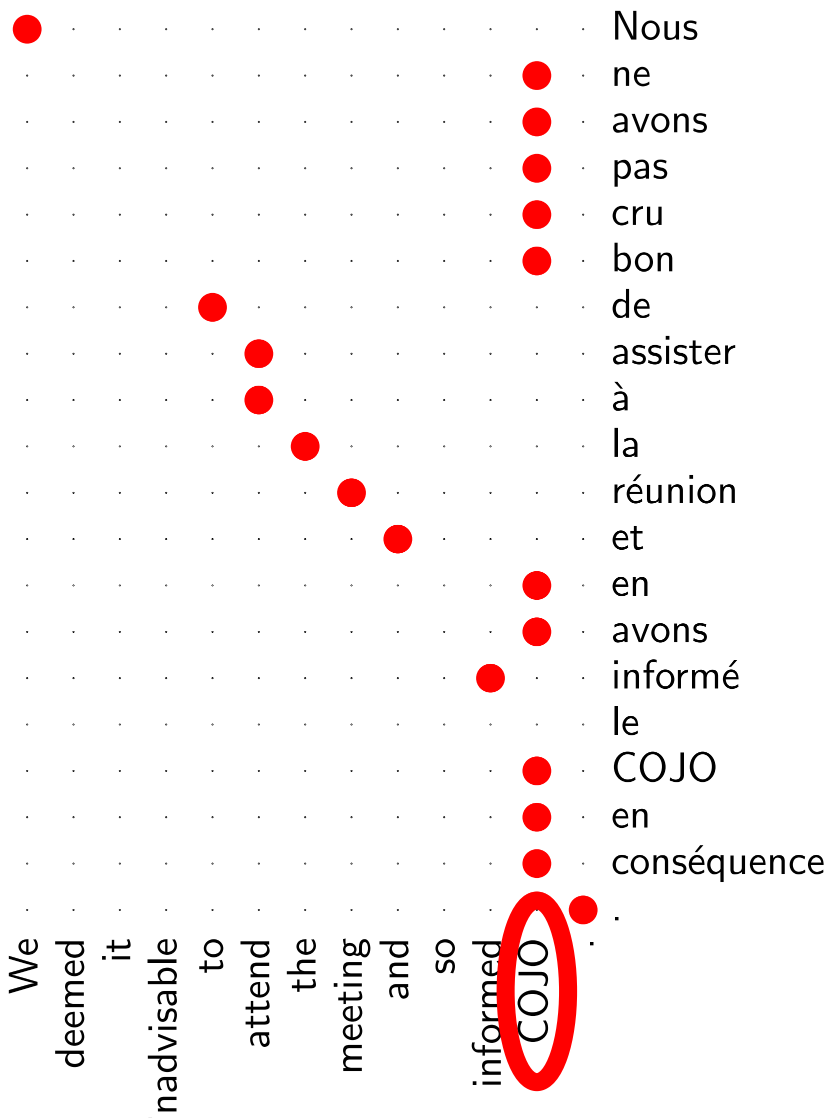
Output of one HMM model



Output of one HMM model

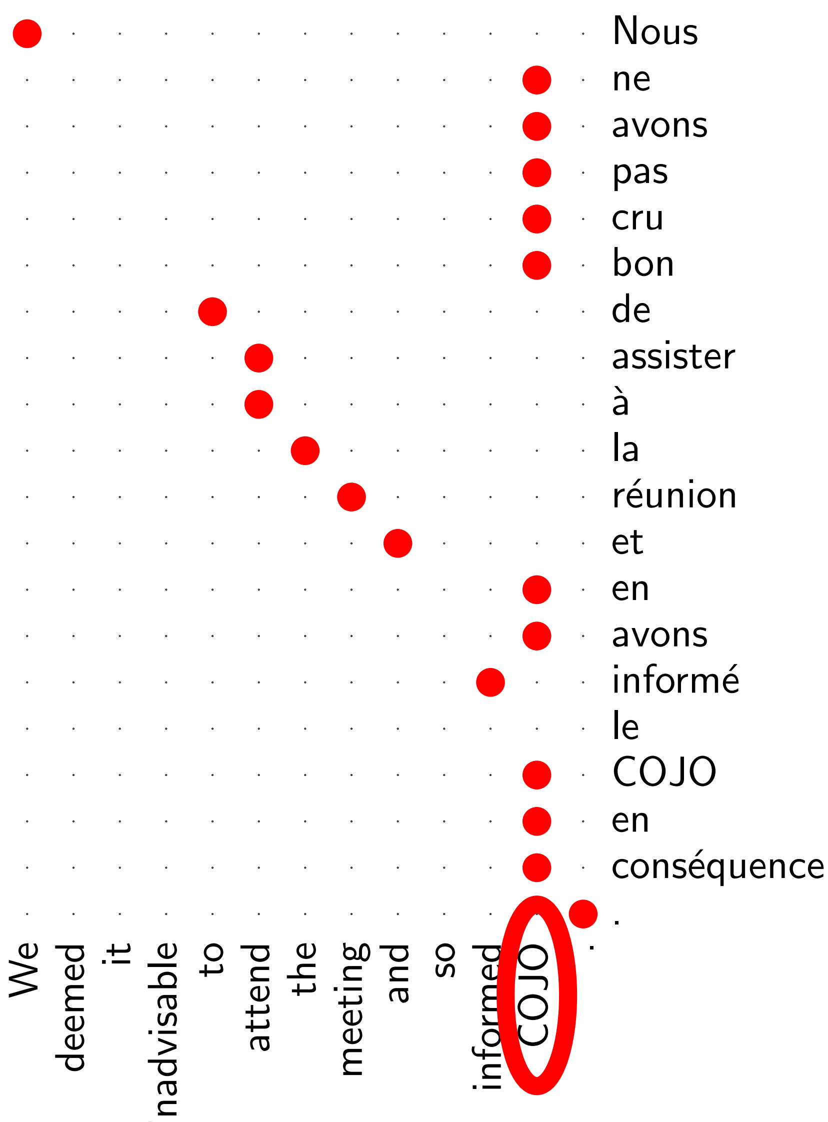


Output of one HMM model



- A problem:
  - Rare words garbage-collect alignments [Moore '05]

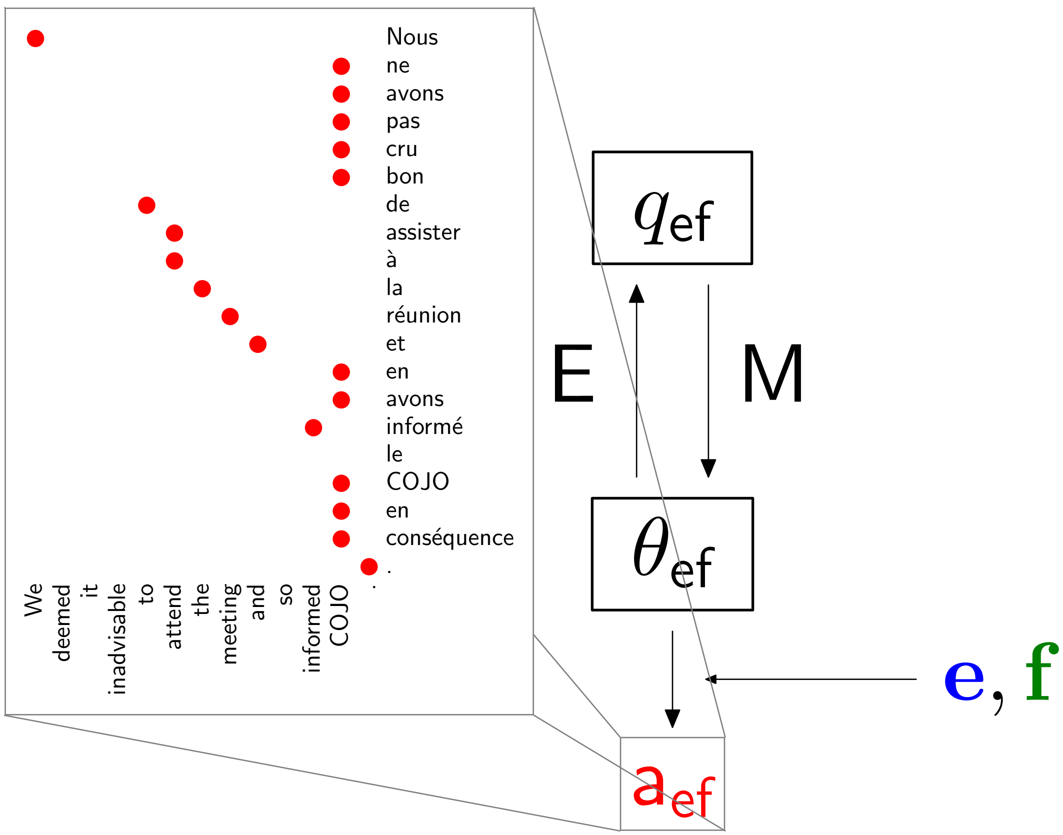
Output of one HMM model



- A problem:
  - Rare words garbage-collect alignments [Moore '05]
- One solution:
  - More complex models

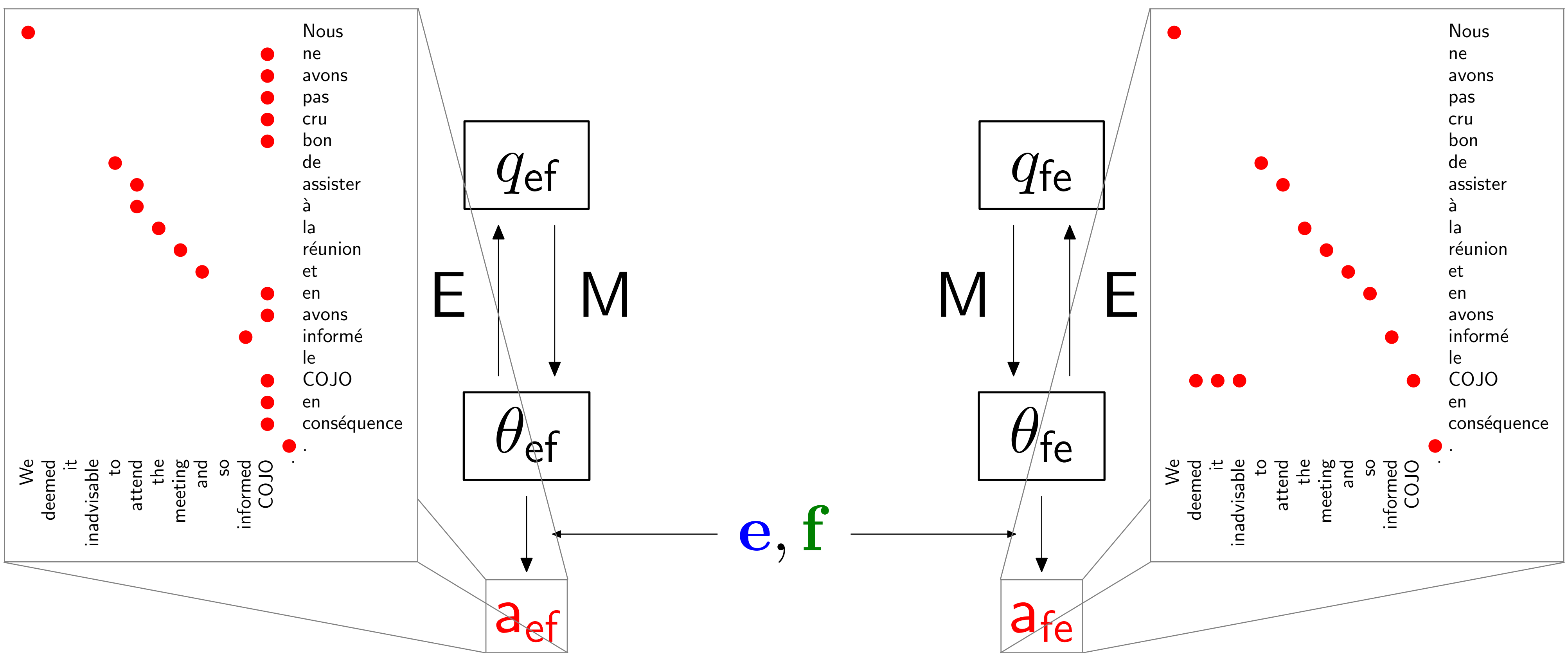
Two complementary models

One model is broken . . .

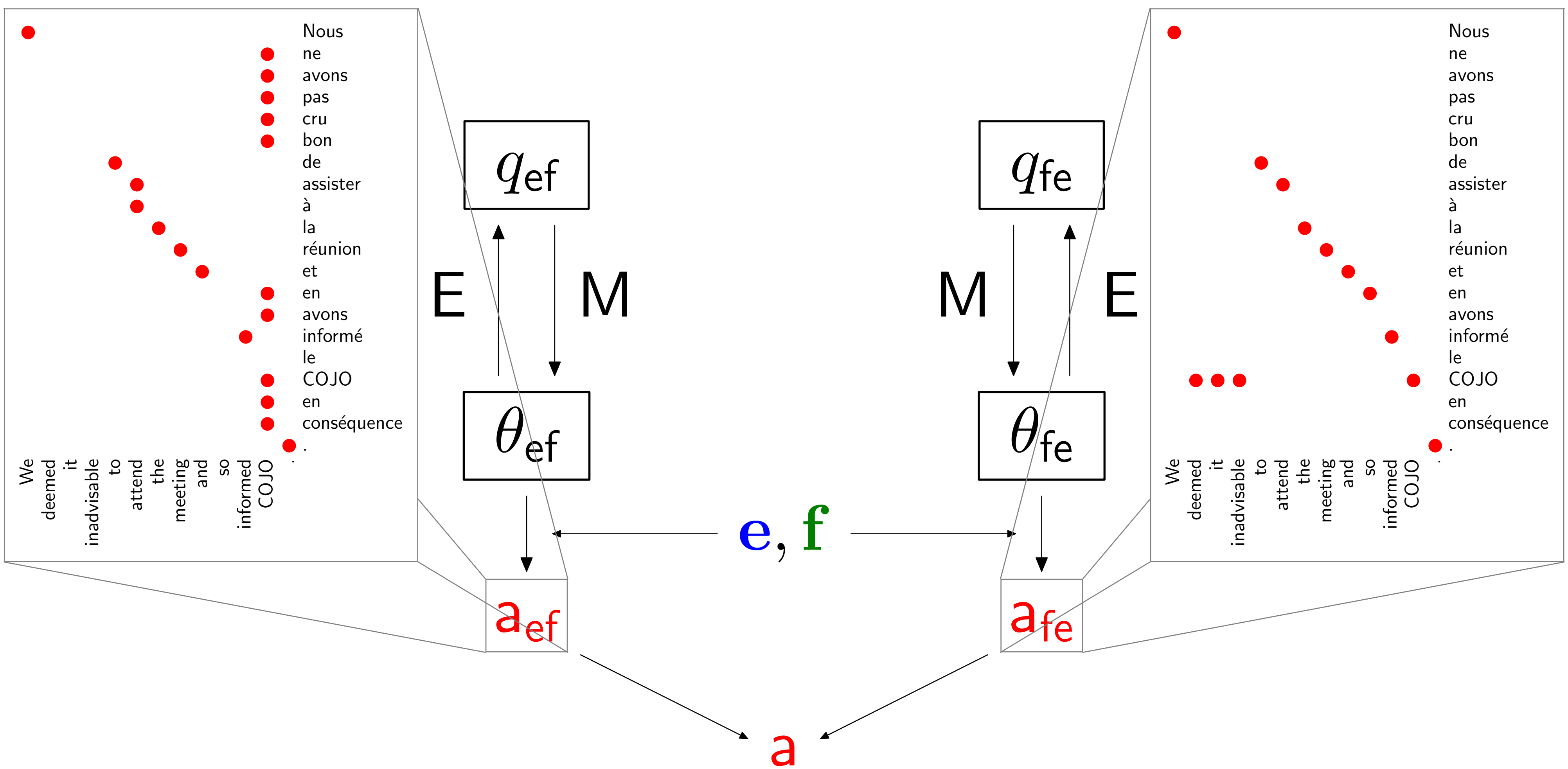


Two complementary models

But second model is not broken in the same way.

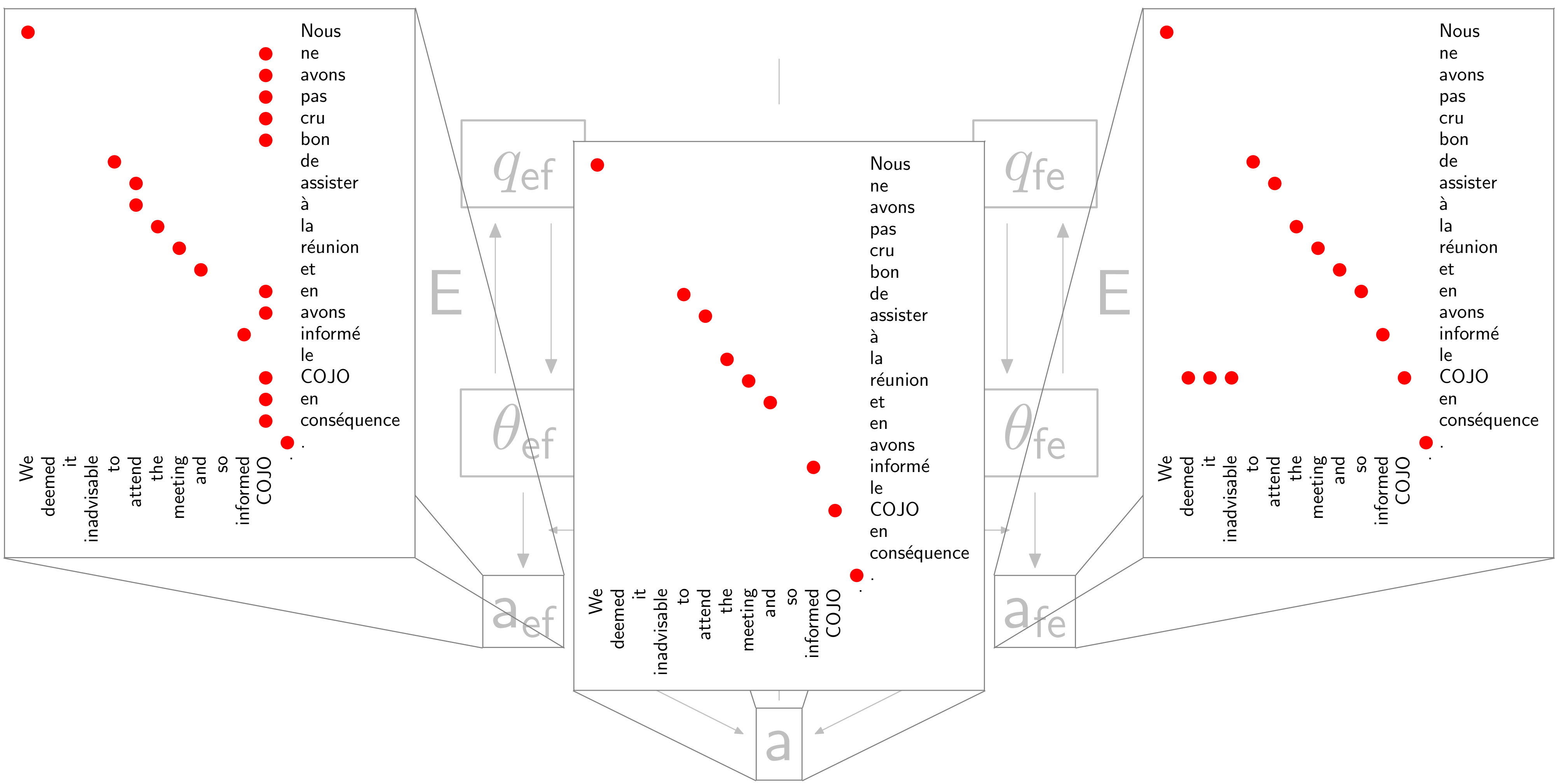


Two complementary models



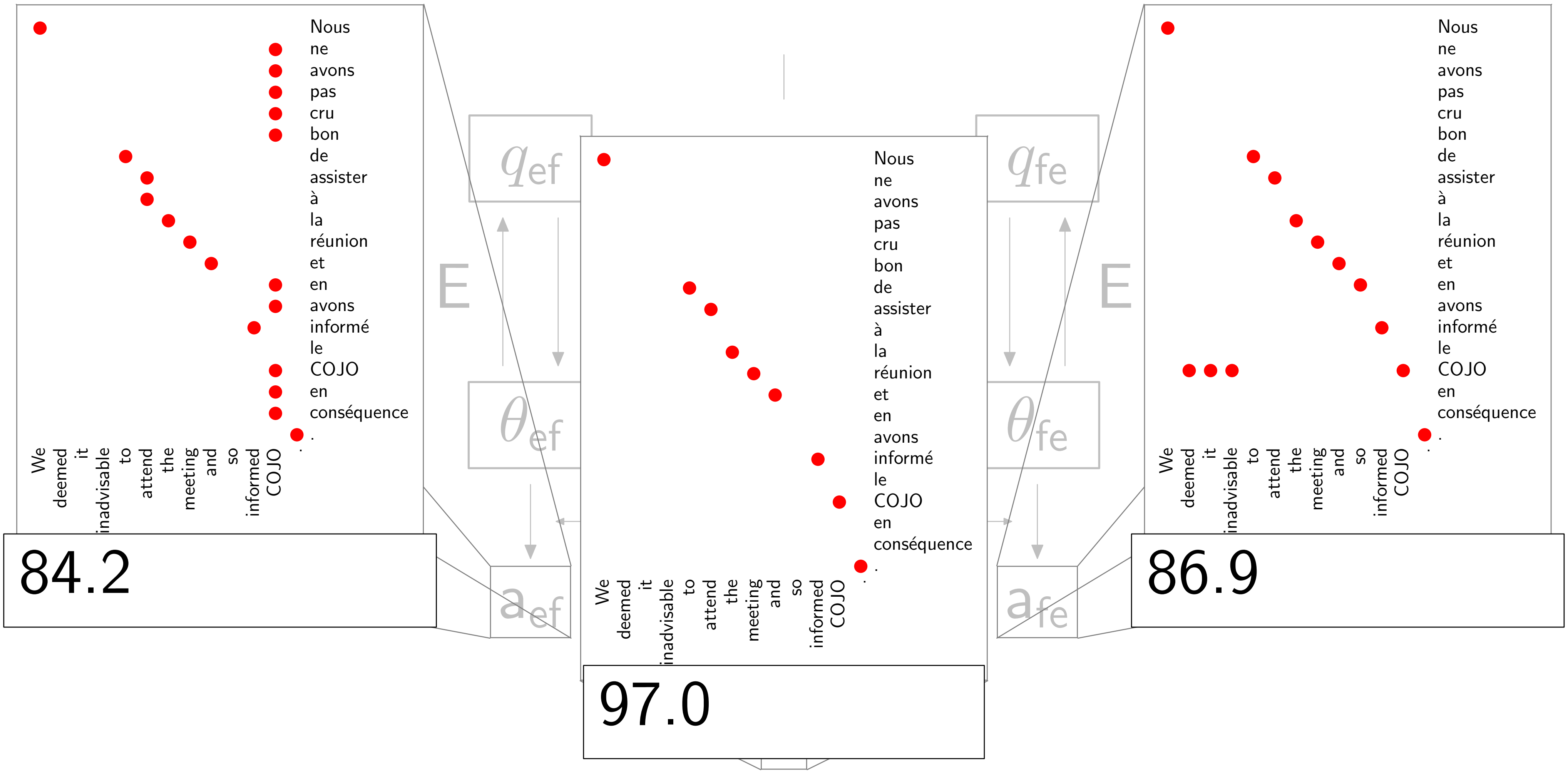
Two complementary models

Intersection kills many bad alignment edges.



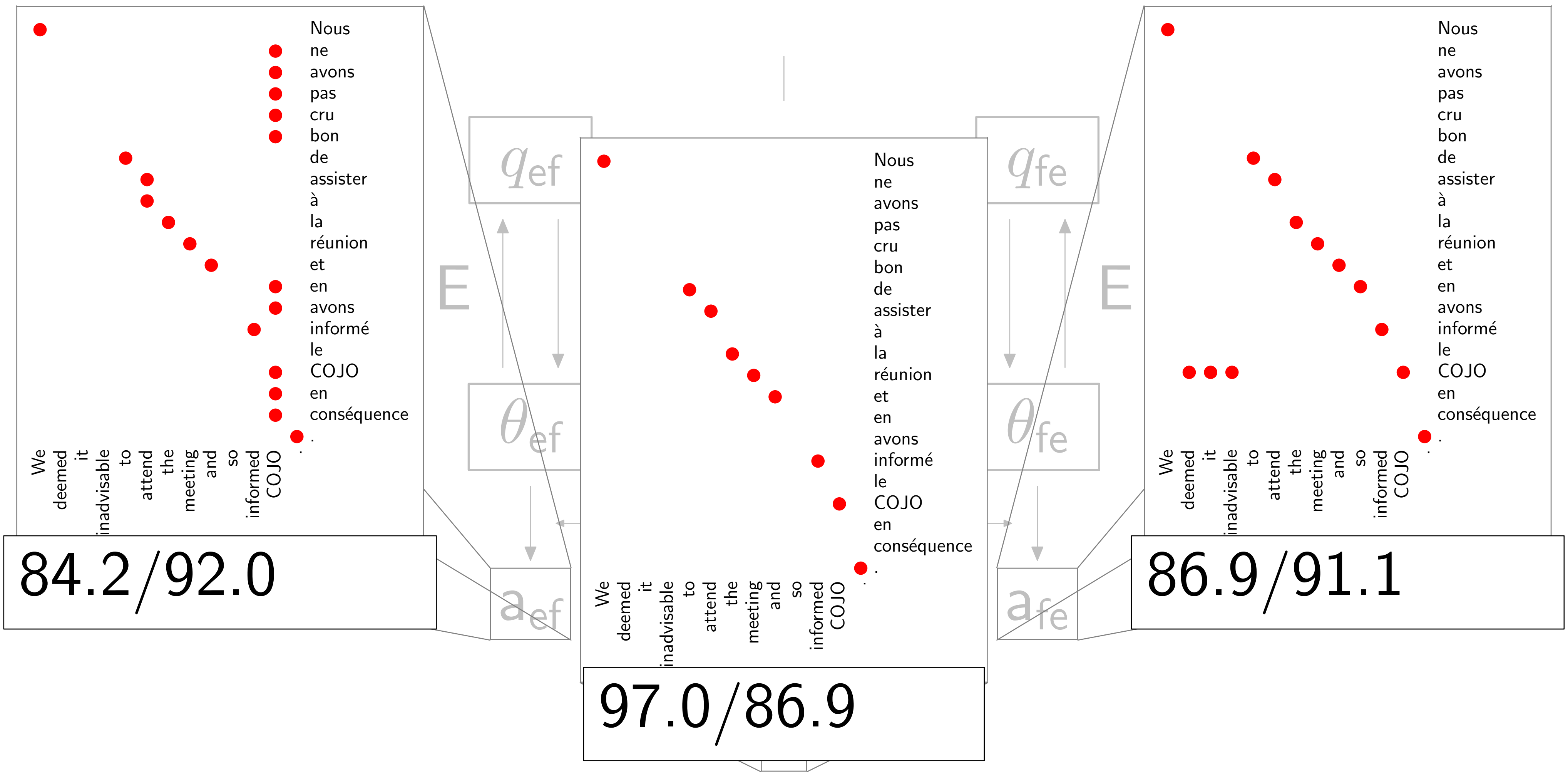
Two complementary models

Precision improves . . .



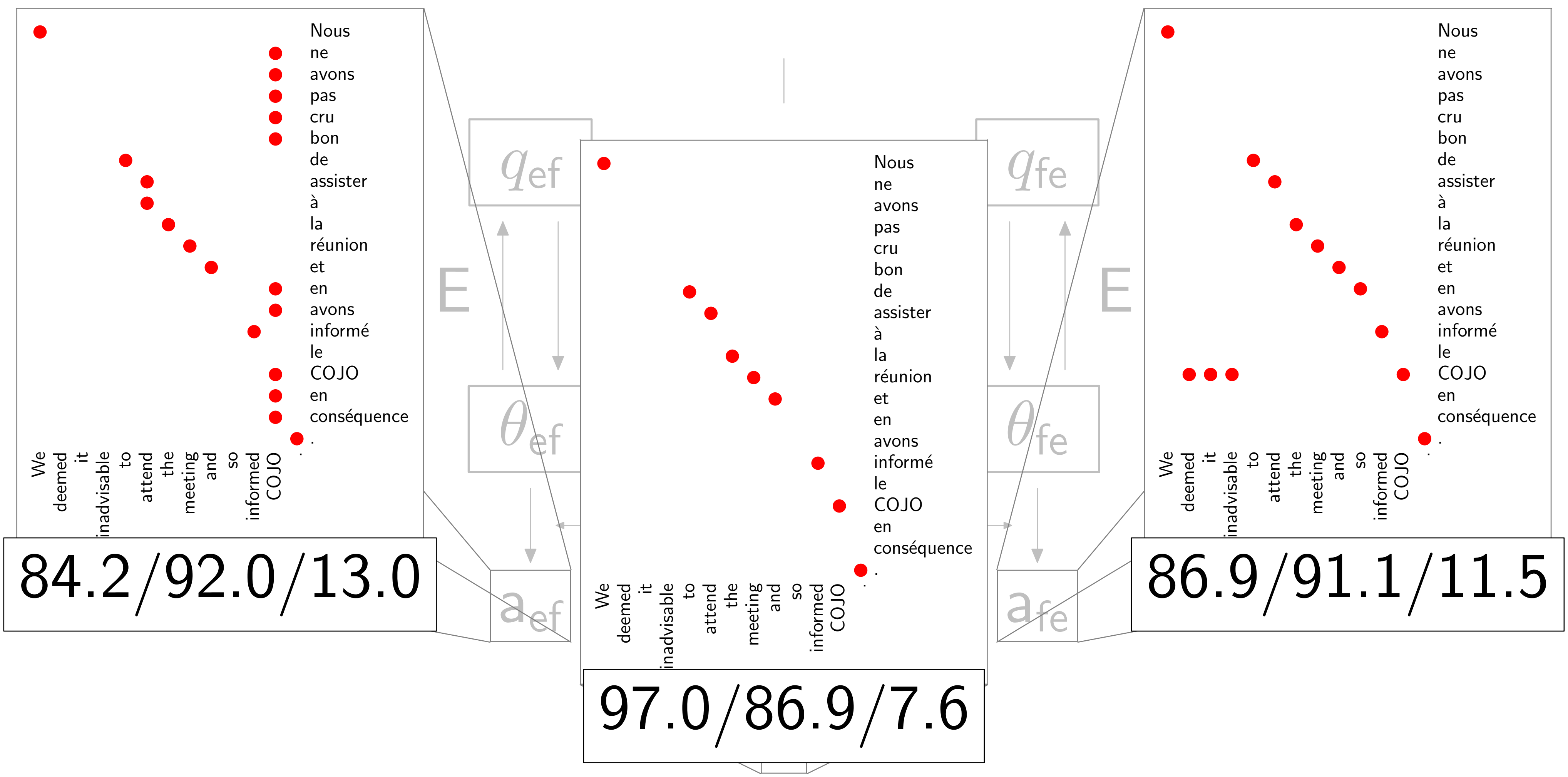
Two complementary models

Precision improves . . . Recall suffers . . .



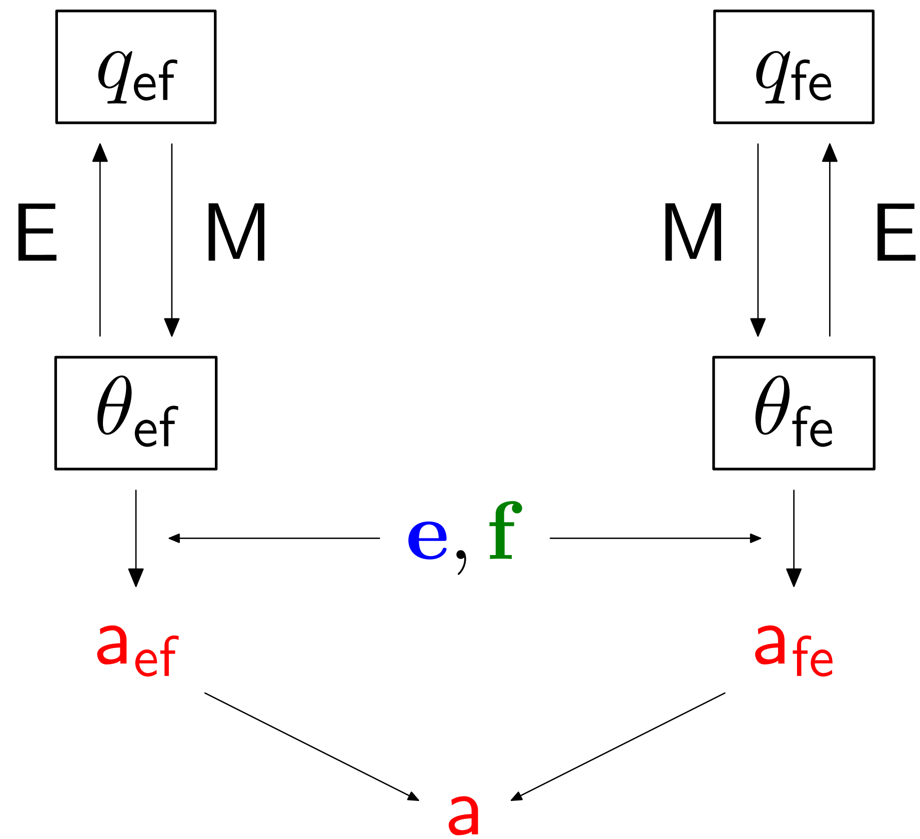
Two complementary models

Precision improves . . . Recall suffers . . . AER improves.



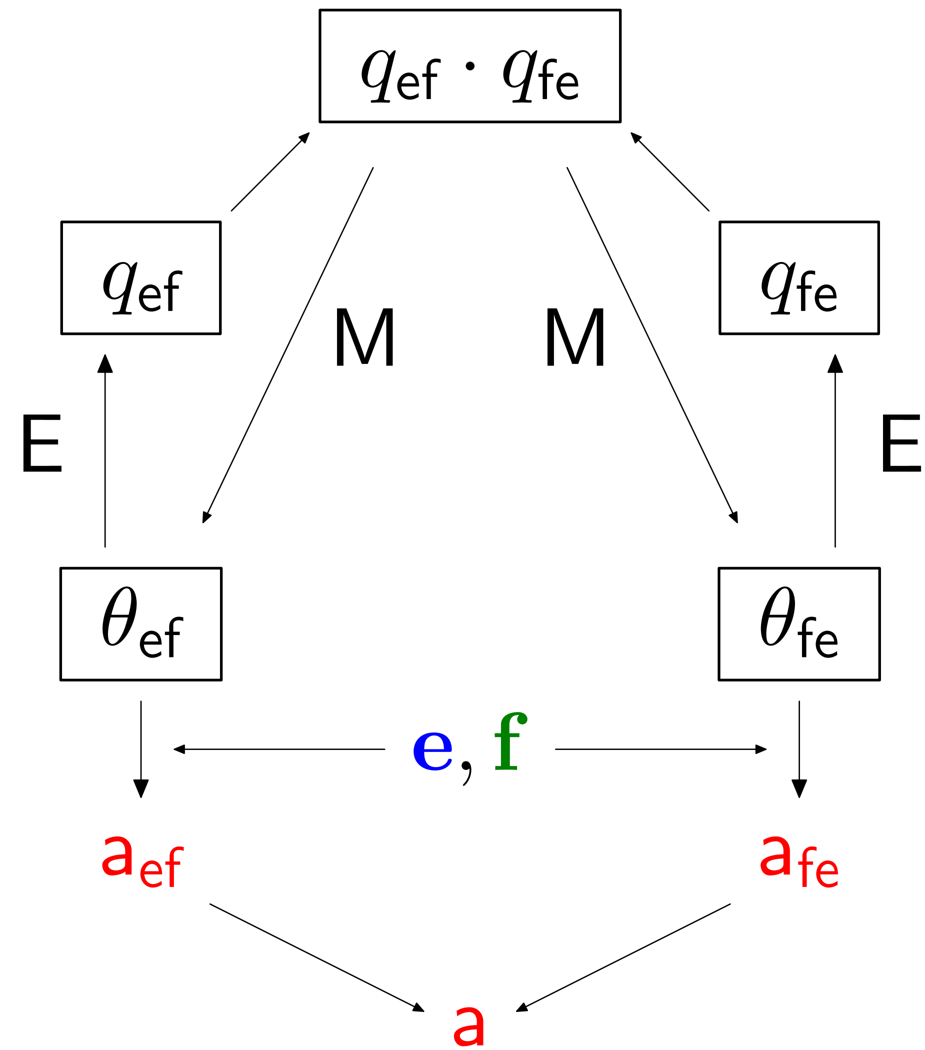
Two complementary models

Can we extend the agreement idea?



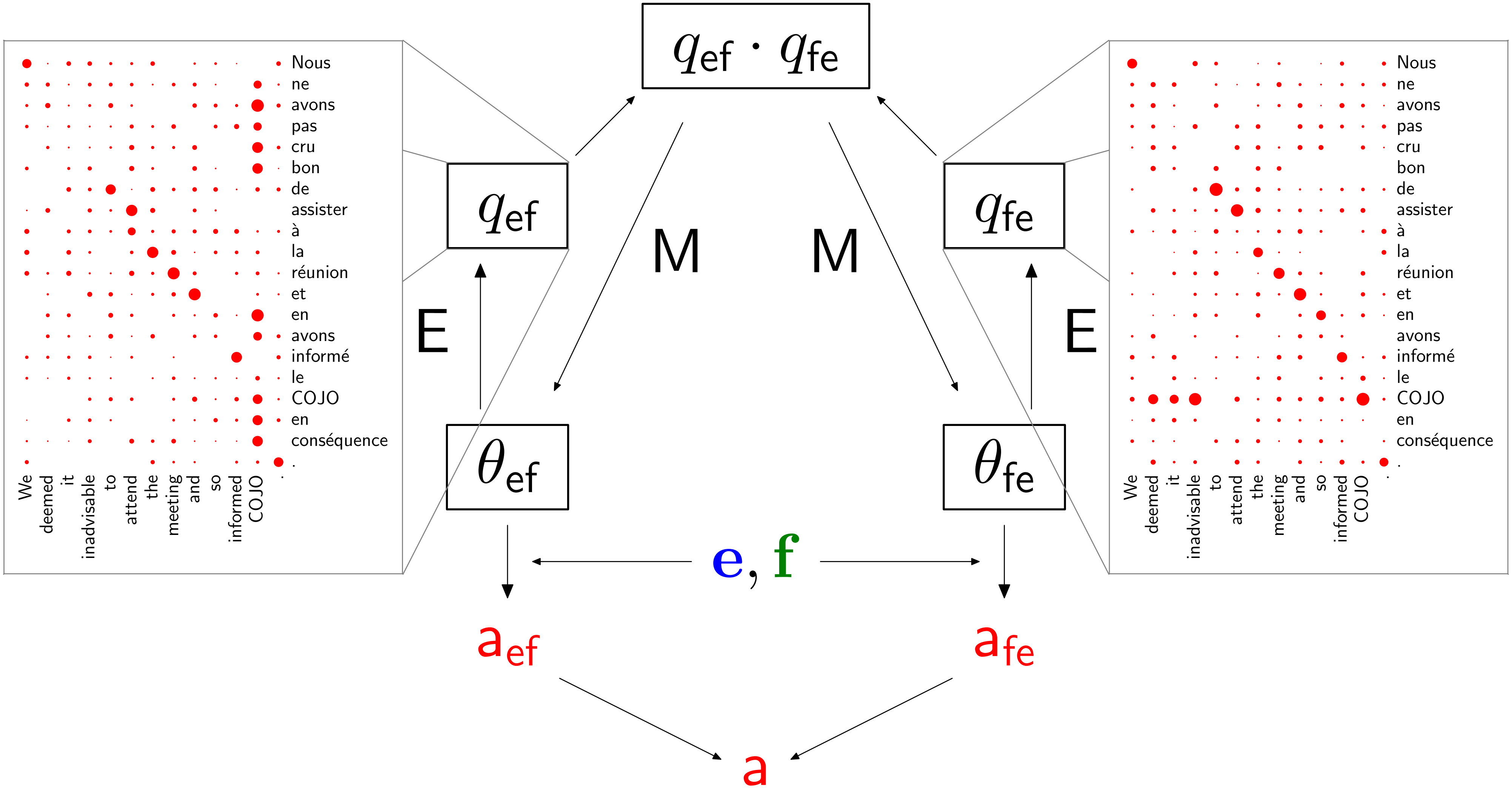
Two complementary models

Key: intersect alignments at training time



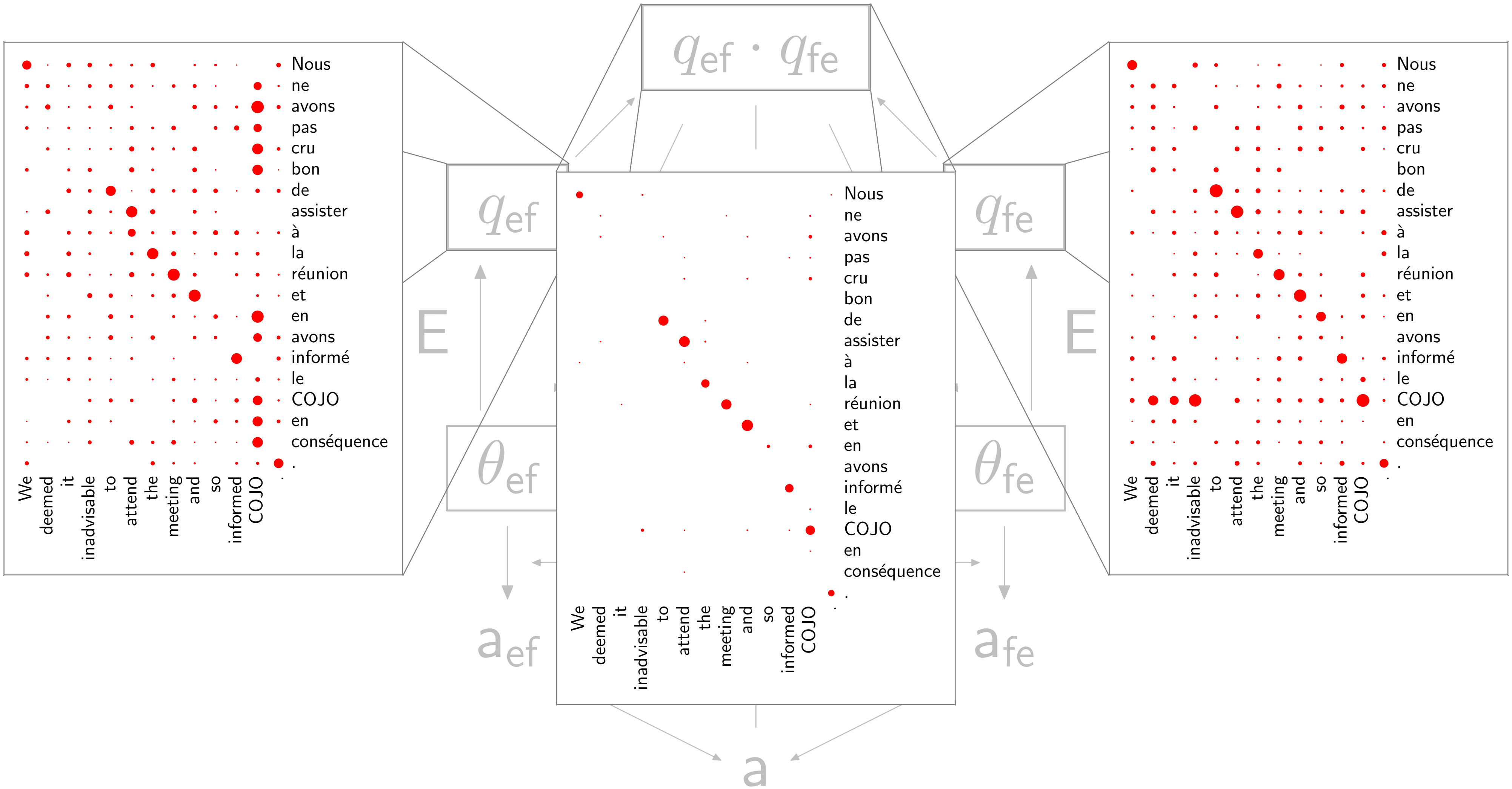
Two complementary models

Fractional alignments



Two complementary models

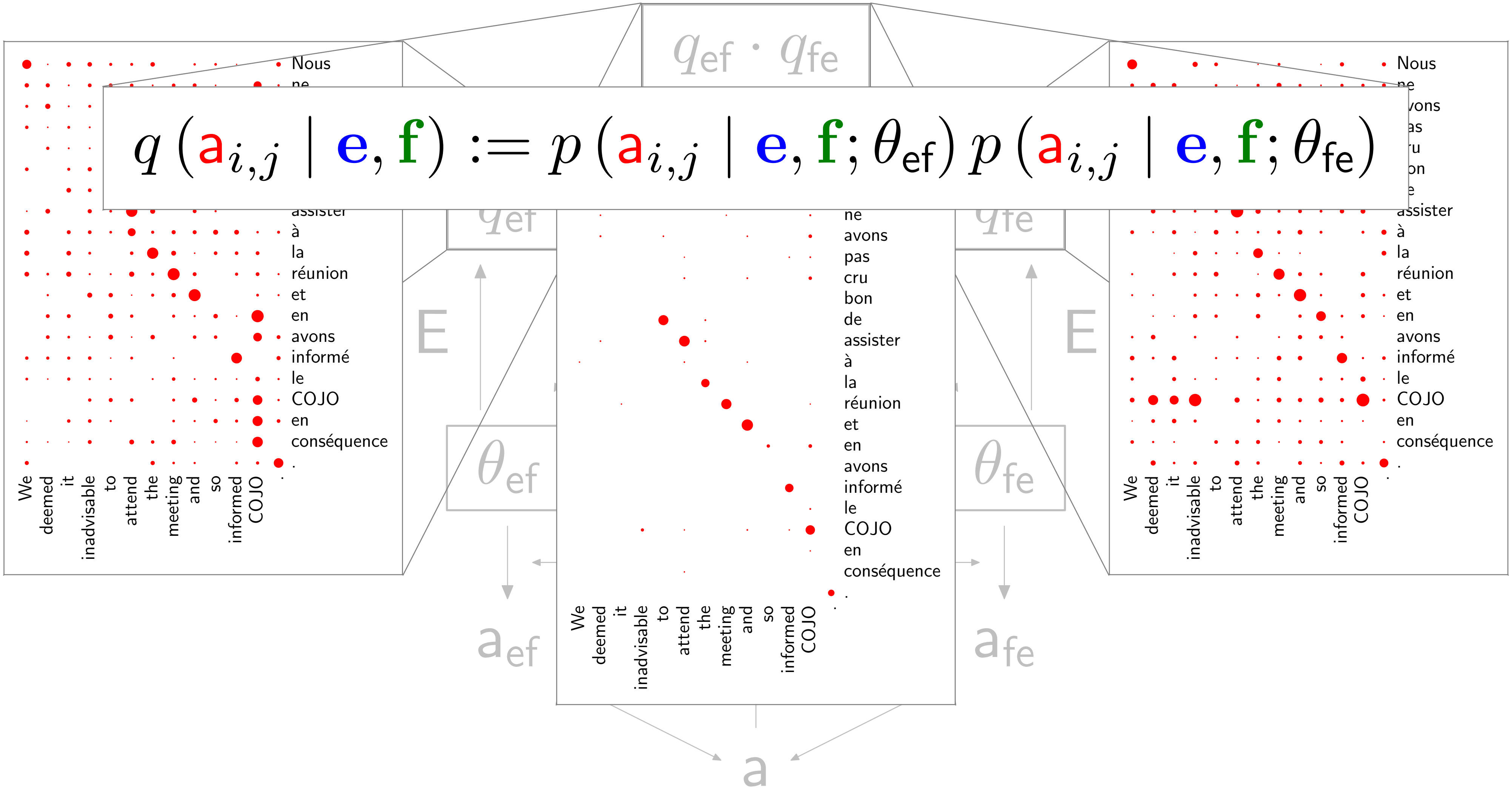
Soft intersection: multiply fractional alignment



Two complementary models

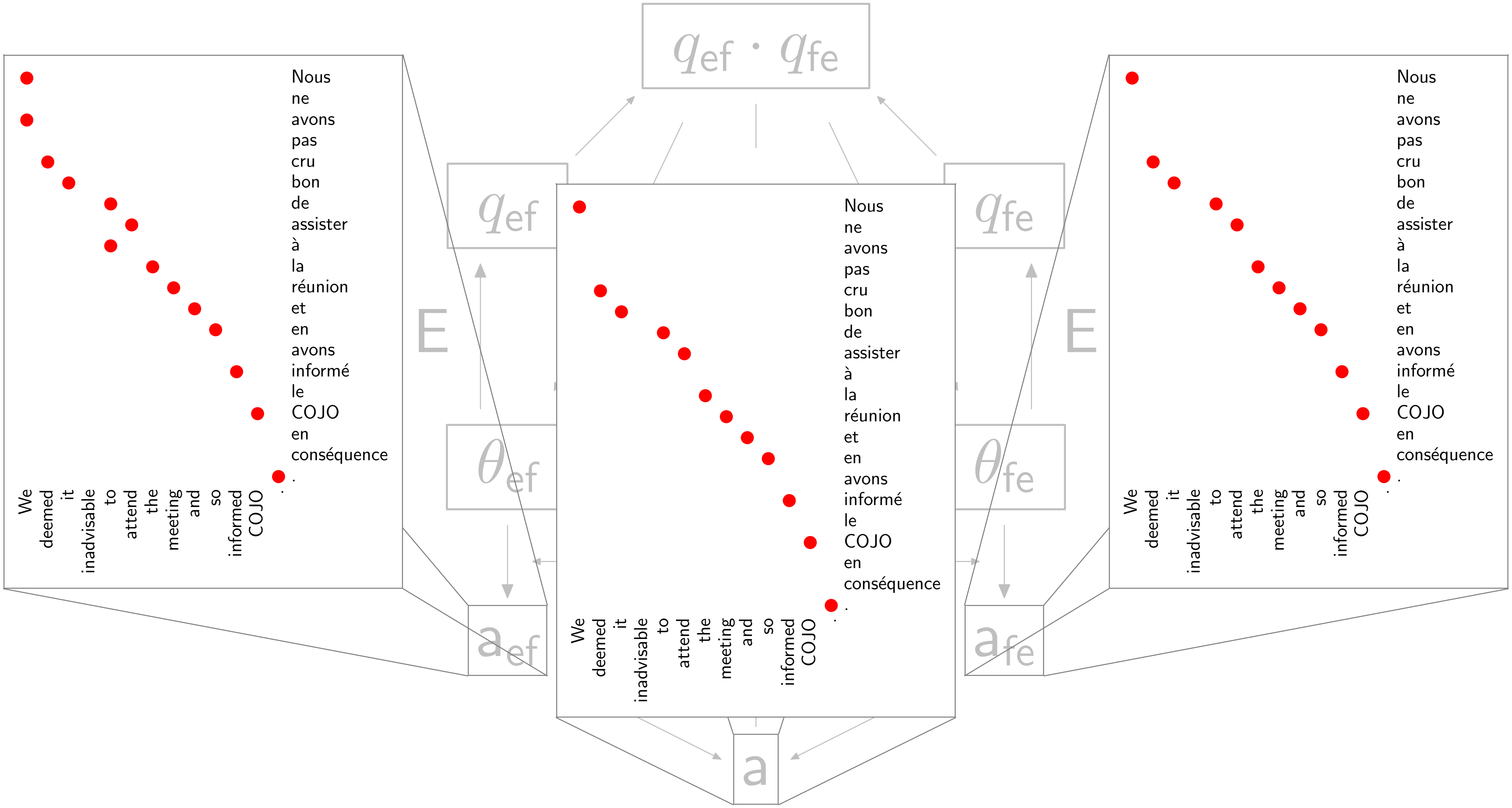
Soft intersection: multiply fractional alignment

$$q\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}\right) := p\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}; \theta_{\mathsf{ef}}\right) p\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}; \theta_{\mathsf{fe}}\right)$$



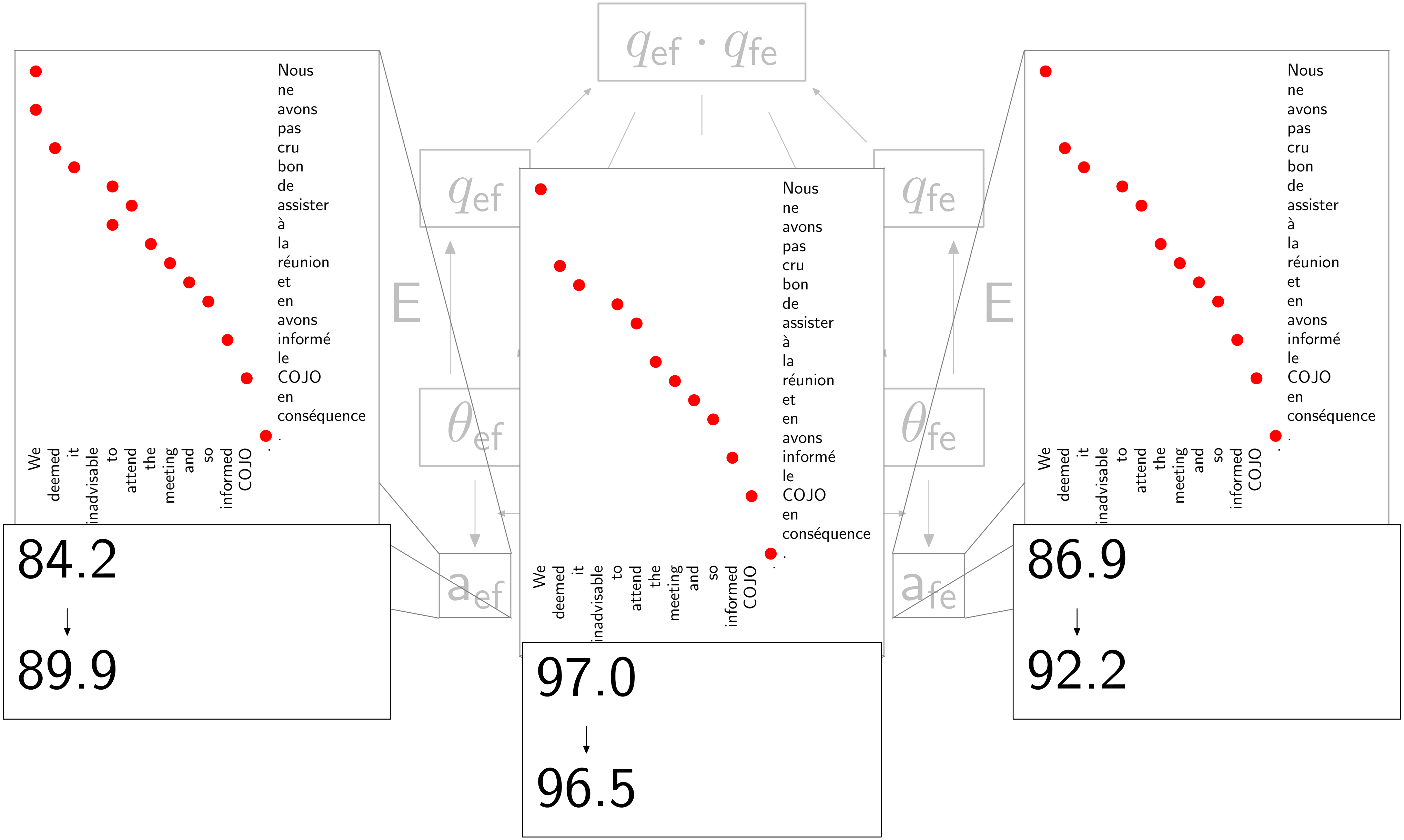
Two complementary models

Models that are trained to agree predict better.



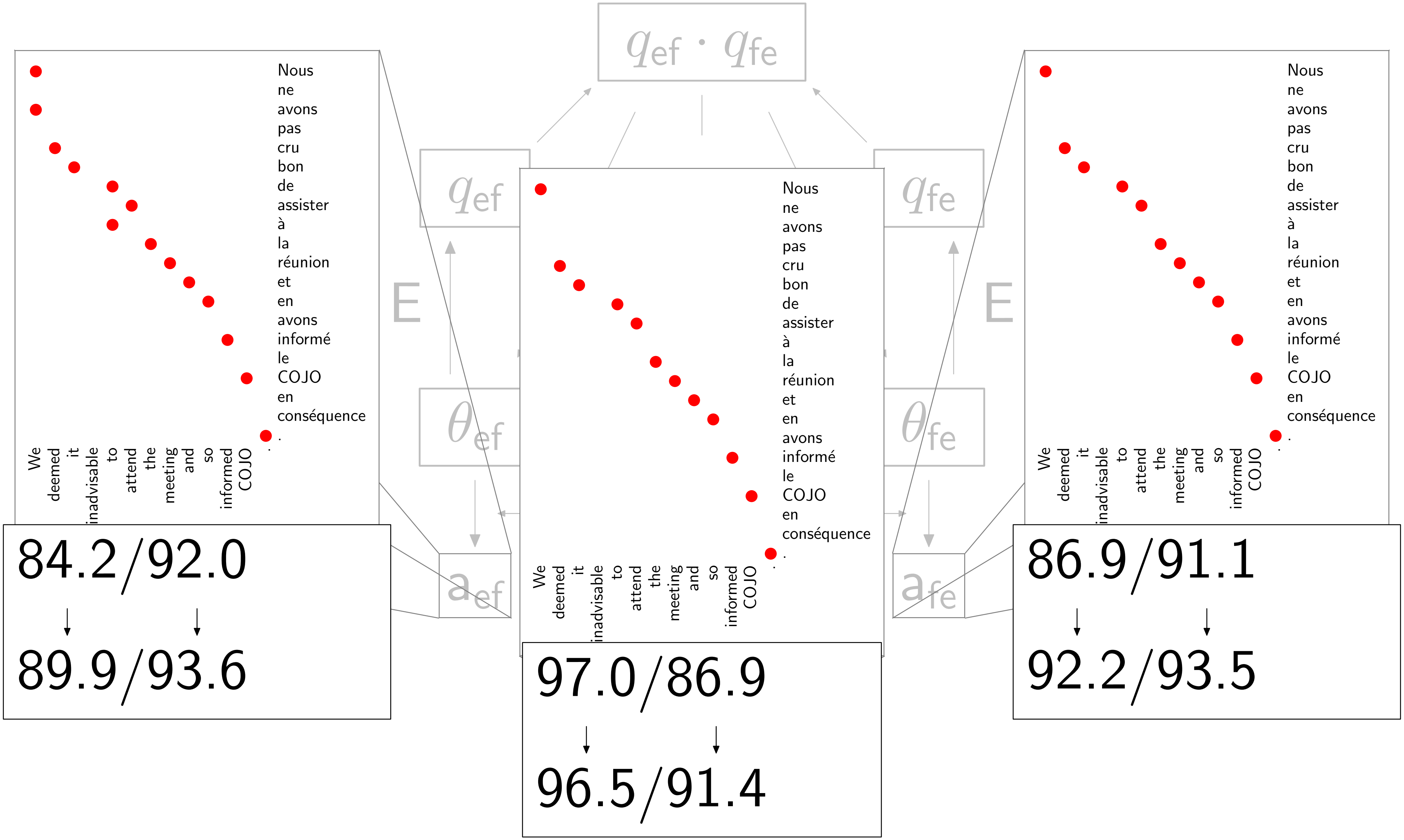
Two complementary models

Models that are trained to agree predict better.



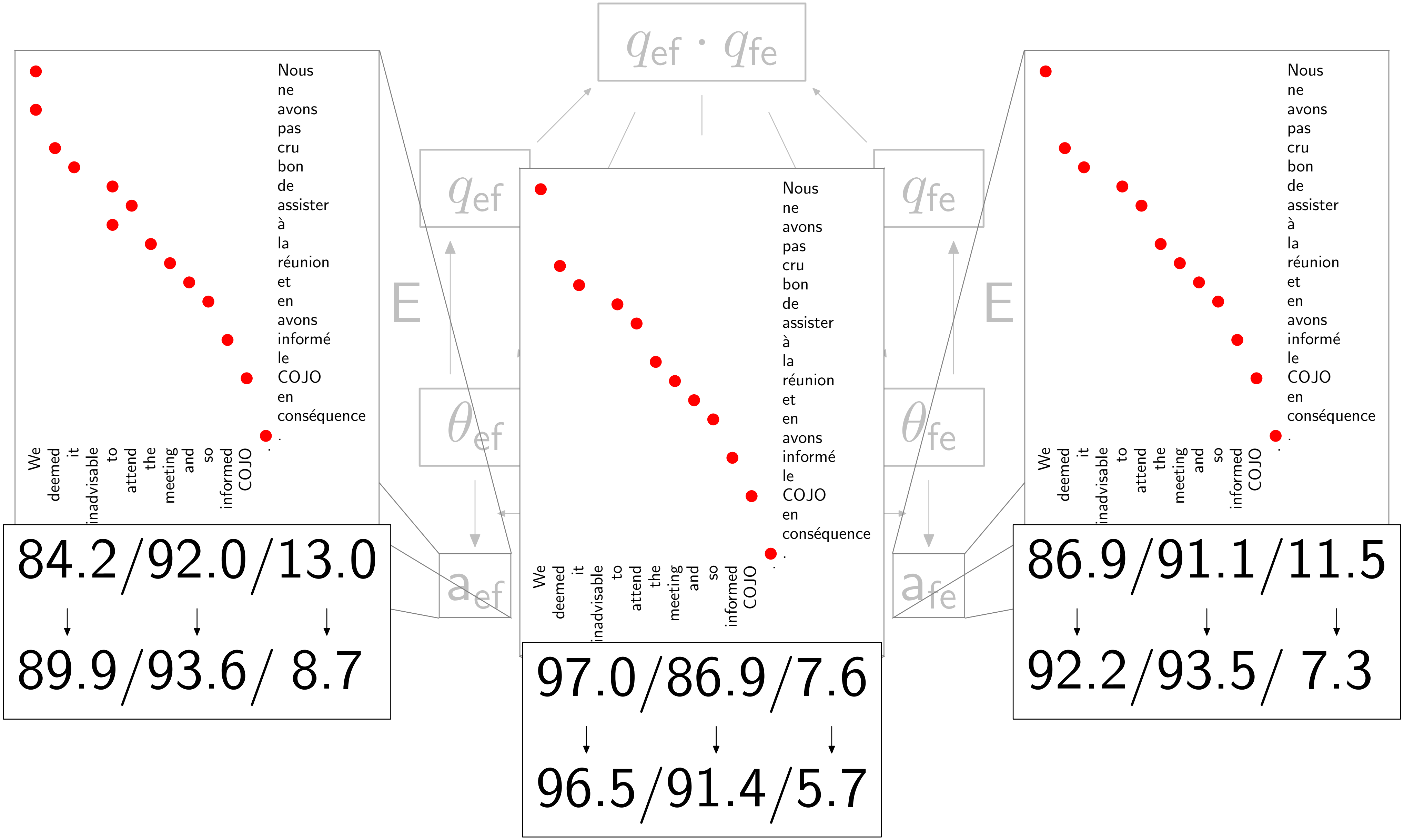
Two complementary models

Models that are trained to agree predict better.



Two complementary models

Models that are trained to agree predict better.



Initialization

Jointly-trained models less sensitive to initialization

Initialization	Indep. HMMs
Uniform	AER>50

Initialization

Jointly-trained models less sensitive to initialization

Initialization	Indep. HMMs
Uniform	AER>50
Model 1	6.6

Initialization

Jointly-trained models less sensitive to initialization

Initialization	Indep. HMMs	Joint HMMs
Uniform	AER>50	5.7
Model 1	6.6	5.2

Initialization

Jointly-trained models less sensitive to initialization

Initialization	Indep. HMMs	Joint HMMs
Uniform	AER>50	5.7
Model 1	6.6	5.2

- Two models have somewhat disjoint capacities for producing bad alignments
- Agreement biases parameters away from troublesome areas

Agreement provides staged training

E-step:

$$q\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}\right) := p\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}; \theta_{\mathsf{ef}}\right) p\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}; \theta_{\mathsf{fe}}\right)$$

Agreement provides staged training

E-step:

$$q\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}\right) := p\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}; \theta_{\mathsf{ef}}\right) p\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}; \theta_{\mathsf{fe}}\right)$$

M-step:

$$\theta_{\mathsf{t}}(\mathit{to} \longrightarrow \mathit{de}) \propto \sum_{\mathsf{e}_i = \mathit{to}, \mathsf{f}_j = \mathit{de}} q\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}\right)$$

- Magnitude of fractional q = influence in M-step

Agreement provides staged training

E-step:

$$q\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}\right) := p\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}; \theta_{\mathsf{ef}}\right) p\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}; \theta_{\mathsf{fe}}\right)$$

M-step:

$$\theta_{\mathsf{t}}(\textit{to} \to \textit{de}) \propto \sum_{\mathsf{e}_i = \textit{to}, \mathsf{f}_j = \textit{de}} q\left(\mathsf{a}_{i,j} \mid \mathbf{e}, \mathbf{f}\right)$$

- Magnitude of fractional q = influence in M-step
- Downweight hard cases where two models disagree
- As models get better, harder examples contribute

General unsupervised approach

- Input $\mathbf{x} = (\mathbf{e}, \mathbf{f})$, output $\mathbf{z} = \mathbf{a}$
- Two complementary models $p_1(\mathbf{x}, \mathbf{z}; \theta_1), p_2(\mathbf{x}, \mathbf{z}; \theta_2)$

General unsupervised approach

- Input $\mathbf{x} = (\mathbf{e}, \mathbf{f})$, output $\mathbf{z} = \mathbf{a}$
- Two complementary models $p_1(\mathbf{x}, \mathbf{z}; \theta_1), p_2(\mathbf{x}, \mathbf{z}; \theta_2)$

$$\overbrace{\max_{\theta_1, \theta_2} \log p_1(\mathbf{x}; \theta_1) + \log p_2(\mathbf{x}; \theta_2)}^{\text{Independent training}}$$

General unsupervised approach

- Input $\mathbf{x} = (\mathbf{e}, \mathbf{f})$, output $\mathbf{z} = \mathbf{a}$
- Two complementary models $p_1(\mathbf{x}, \mathbf{z}; \theta_1), p_2(\mathbf{x}, \mathbf{z}; \theta_2)$
- Joint training objective:

$$\overbrace{\max_{\theta_1,\theta_2} \log p_1(\mathbf{x}; \theta_1) + \log p_2(\mathbf{x}; \theta_2)}^{\text{Independent training}}$$

$$+ \underbrace{\log \sum_{\mathbf{z}} p_1(\mathbf{z} \mid \mathbf{x}; \theta_1) p_2(\mathbf{z} \mid \mathbf{x}; \theta_2)}_{\text{agreement}}$$

General unsupervised approach

- Input $\mathbf{x} = (\mathbf{e}, \mathbf{f})$, output $\mathbf{z} = \mathbf{a}$
- Two complementary models $p_1(\mathbf{x}, \mathbf{z}; \theta_1), p_2(\mathbf{x}, \mathbf{z}; \theta_2)$
- Joint training objective:

$$\overbrace{\max_{\theta_1, \theta_2} \log p_1(\mathbf{x}; \theta_1) + \log p_2(\mathbf{x}; \theta_2)}^{\text{Independent training}}$$

$$+ \underbrace{\log \sum_{\mathbf{z}} p_1(\mathbf{z} \mid \mathbf{x}; \theta_1) p_2(\mathbf{z} \mid \mathbf{x}; \theta_2)}_{\text{agreement}}$$

E-step: $q(\mathbf{z} \mid \mathbf{x}) \propto p_1(\mathbf{z} \mid \mathbf{x}; \theta_1) p_2(\mathbf{z} \mid \mathbf{x}; \theta_2)$

General unsupervised approach

- Input $\mathbf{x} = (\mathbf{e}, \mathbf{f})$, output $\mathbf{z} = \mathbf{a}$
- Two complementary models $p_1(\mathbf{x}, \mathbf{z}; \theta_1), p_2(\mathbf{x}, \mathbf{z}; \theta_2)$
- Joint training objective:

$$\overbrace{\max_{\theta_1, \theta_2} \log p_1(\mathbf{x}; \theta_1) + \log p_2(\mathbf{x}; \theta_2)}^{\text{Independent training}}$$

$$+ \underbrace{\log \sum_{\mathbf{z}} p_1(\mathbf{z} \mid \mathbf{x}; \theta_1) p_2(\mathbf{z} \mid \mathbf{x}; \theta_2)}_{\text{agreement}}$$

E-step: $q(\mathbf{z} \mid \mathbf{x}) \propto p_1(\mathbf{z} \mid \mathbf{x}; \theta_1) p_2(\mathbf{z} \mid \mathbf{x}; \theta_2)$

Useful in grammar induction [Klein, Manning '04]

General unsupervised approach

- Input $\mathbf{x} = (\mathbf{e}, \mathbf{f})$, output $\mathbf{z} = \mathbf{a}$
- Two complementary models $p_1(\mathbf{x}, \mathbf{z}; \theta_1), p_2(\mathbf{x}, \mathbf{z}; \theta_2)$
- Joint training objective:

$$\max_{\theta_1, \theta_2} \overbrace{\log p_1(\mathbf{x}; \theta_1) + \log p_2(\mathbf{x}; \theta_2)}^{\text{Independent training}}$$

$$+ \underbrace{\log \sum_{\mathbf{z}} p_1(\mathbf{z} \mid \mathbf{x}; \theta_1) p_2(\mathbf{z} \mid \mathbf{x}; \theta_2)}_{\text{agreement}}$$

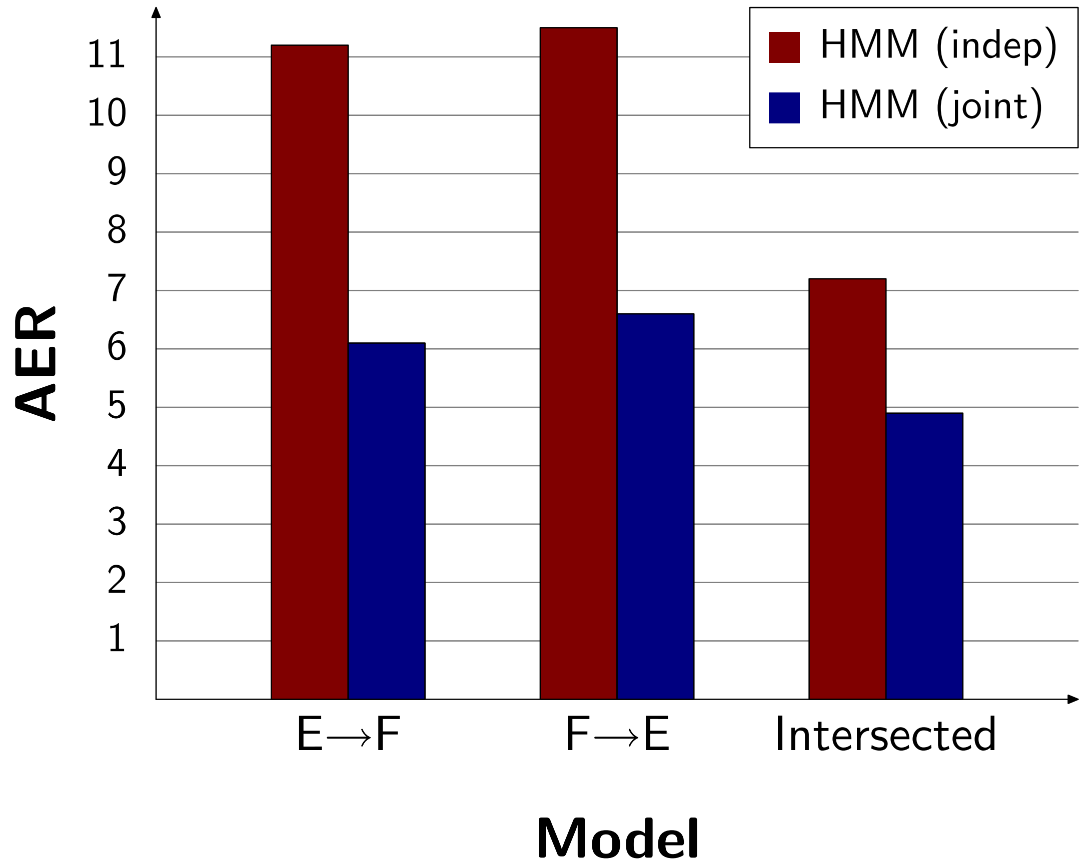
E-step: $q(\mathbf{z} \mid \mathbf{x}) \propto p_1(\mathbf{z} \mid \mathbf{x}; \theta_1) p_2(\mathbf{z} \mid \mathbf{x}; \theta_2)$

Useful in grammar induction [Klein, Manning '04]

Related work: co-training [Blum, Mitchell '98]
CoBoost [Collins, Singer '99]

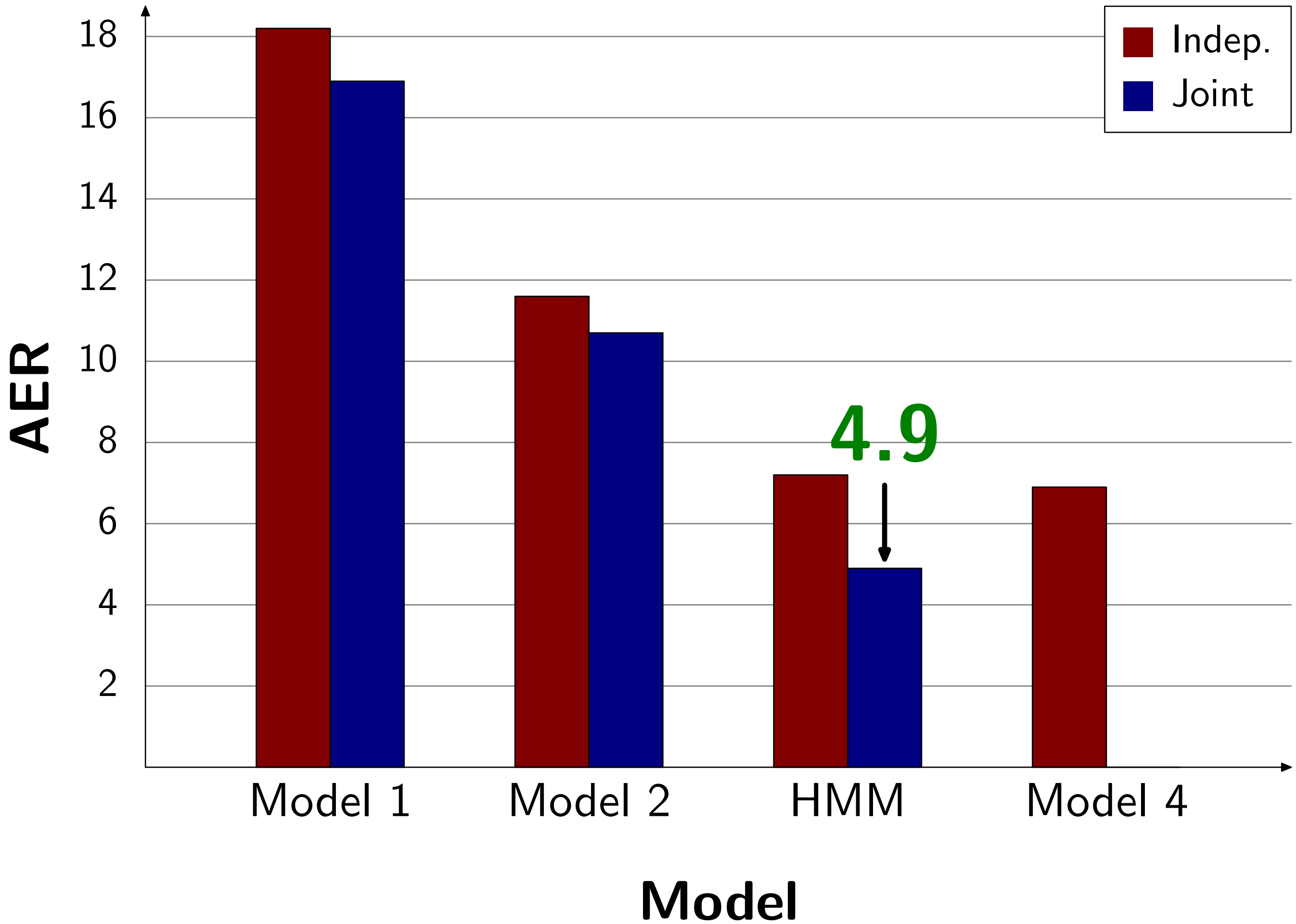
Final results

Hansards (1.1M training sentences, 347 test sentences)



Joint training improves both directional and intersected models

Final results



Significant error reduction for various models

29% reduction in AER over model 4

Conclusion

- Simple and efficient procedure $\longrightarrow$ 4.9% AER

Conclusion

- Simple and efficient procedure $\rightarrow$ 4.9% AER

- Suggests a general approach for unsupervised learning

Conclusion

- Simple and efficient procedure $\rightarrow$ 4.9% AER
- Suggests a general approach for unsupervised learning
- Achieves insignificantly better BLEU score

Conclusion

- Simple and efficient procedure $\rightarrow$ 4.9% AER
- Suggests a general approach for unsupervised learning
- Achieves insignificantly better BLEU score
- Provides features for discriminative methods