CRF Word Alignment & Noisy Channel Translation

January 31, 2013

Last Time ...

$$p(\text{Translation}) = \sum_{\text{Alignment}} p(\text{Alignment}, \text{Translation})$$

Last Time ...

$$p(\text{Translation}) = \sum_{\text{Alignment}} p(\text{Alignment}, \text{Translation})$$

$$= \sum_{\text{Alignment}} p(\text{Alignment}) \times p(\text{Translation} \mid \text{Alignment})$$

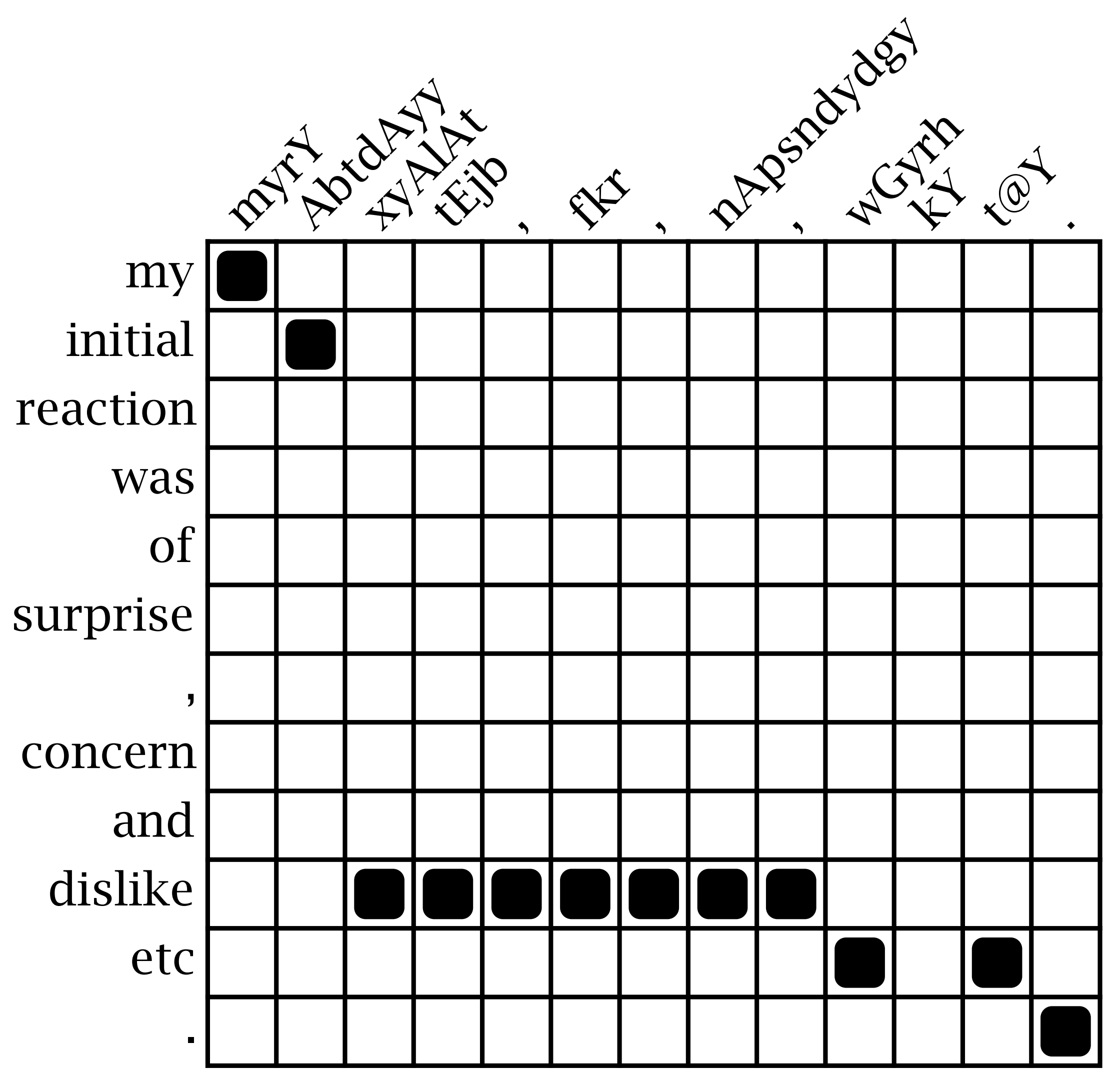
Last Time ...

$$p(\text{Translation}) = \sum_{\text{Alignment}} p(\text{Alignment}, \text{Translation})$$

$$= \sum_{\text{Alignment}} \underbrace{p(\text{Alignment})} \times p\underbrace{(\text{Translation} \mid \text{Alignment})}$$

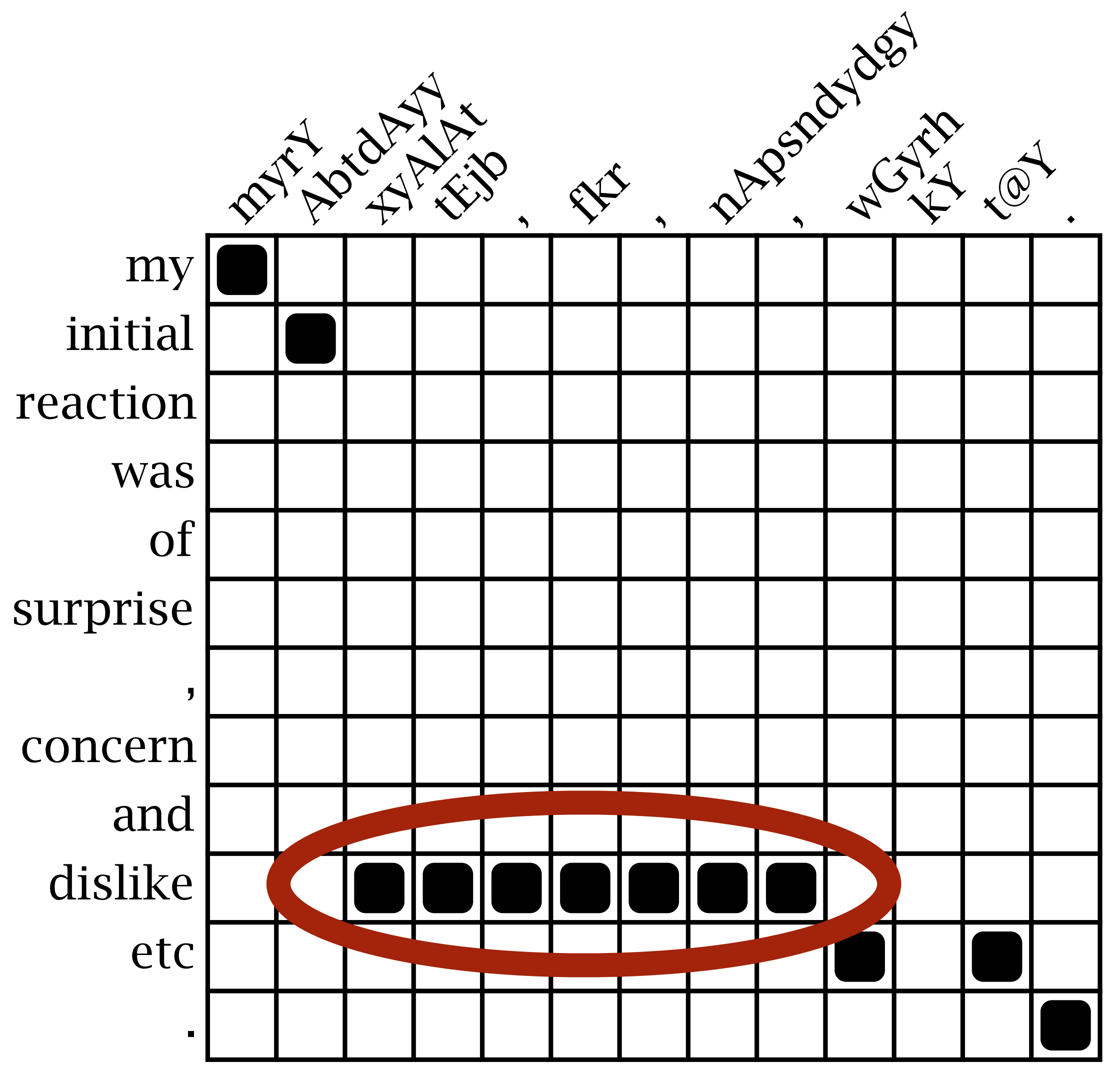
$$p(\mathbf{e} \mid \mathbf{f}, m) = \sum_{\mathbf{a} \in [0,n]^m} \overbrace{p(\mathbf{a} \mid \mathbf{f}, m)} \times \overbrace{\prod_{i=1}^{m} p(e_i \mid f_{a_i})}$$

MAP alignment



IBM Model 4 alignment

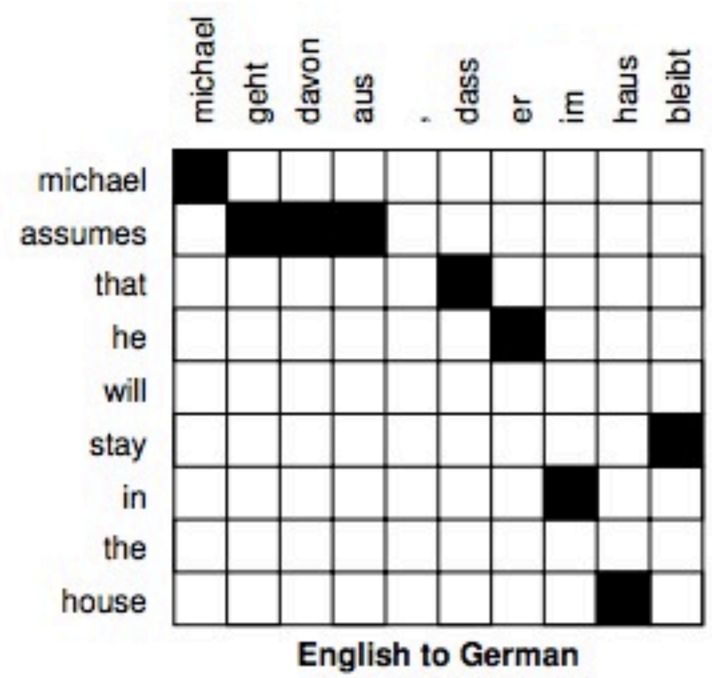
MAP alignment



IBM Model 4 alignment

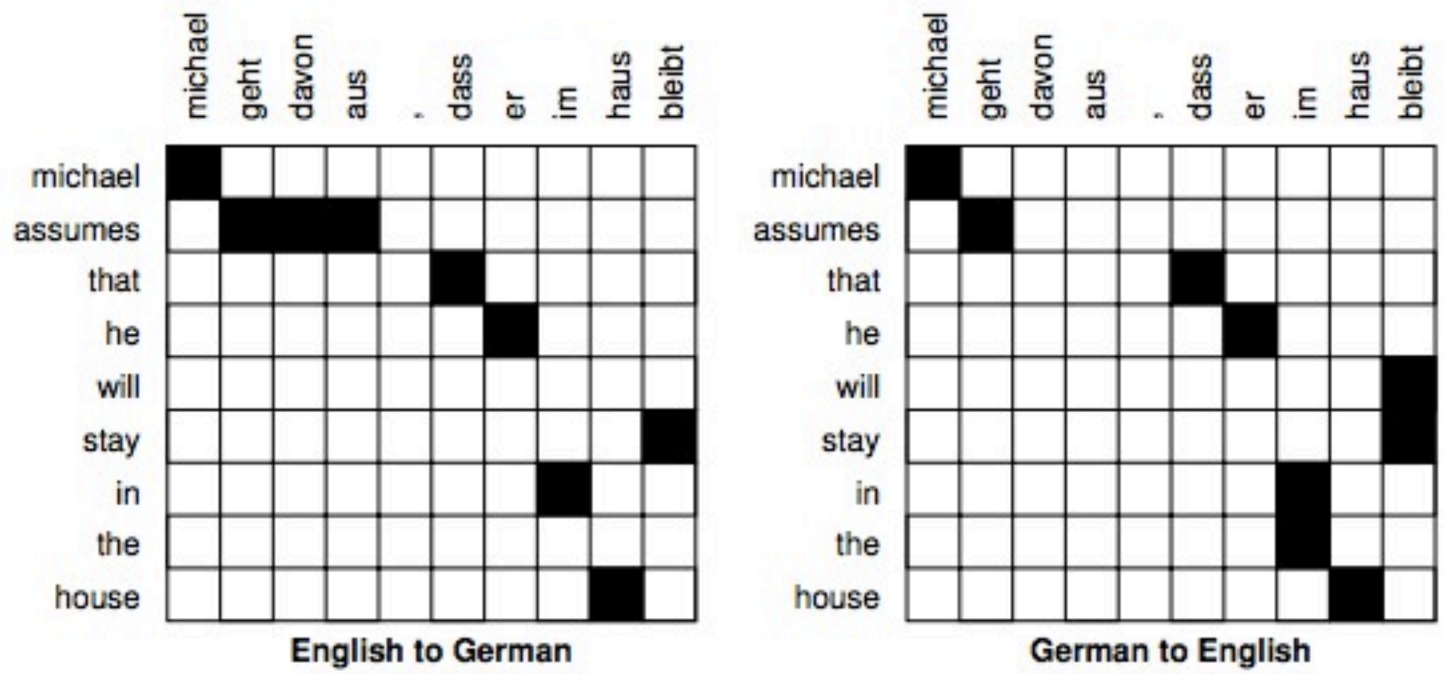
A few tricks...

p(f|e)



A few tricks...

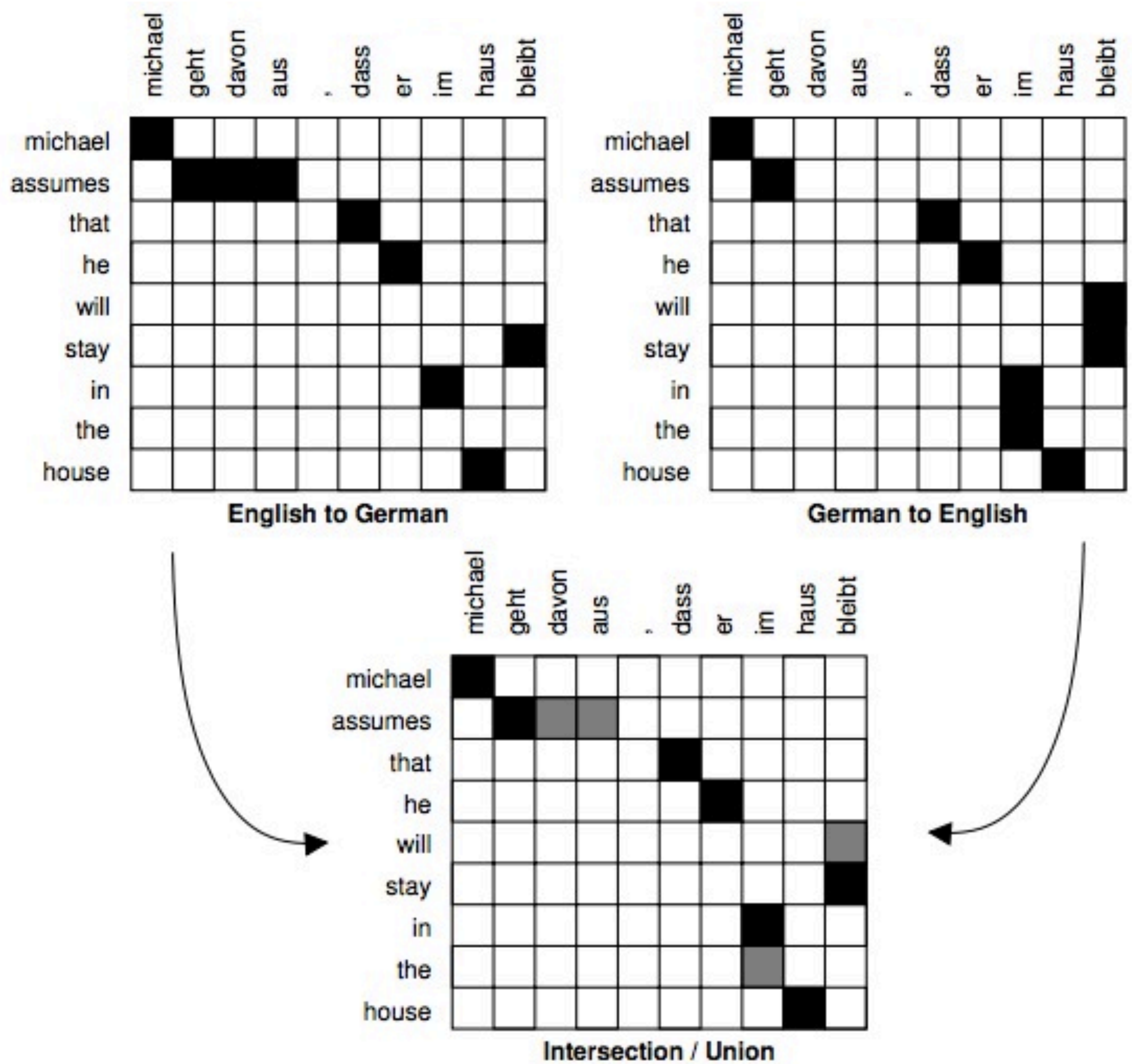
p(f|e)



p(e|f)

A few tricks...

p(f|e)



p(e|f)

Another View

With this model:

$$p(\mathbf{e} \mid \mathbf{f}, m) = \sum_{\mathbf{a} \in [0,n]^m} p(\mathbf{a} \mid \mathbf{f}, m) \times \prod_{i=1}^{m} p(e_i \mid f_{a_i})$$

The problem of word alignment is as:

$$\mathbf{a}^* = \arg \max_{\mathbf{a} \in [0,n]^m} p(\mathbf{a} \mid \mathbf{e}, \mathbf{f}, m)$$

Another View

With this model:

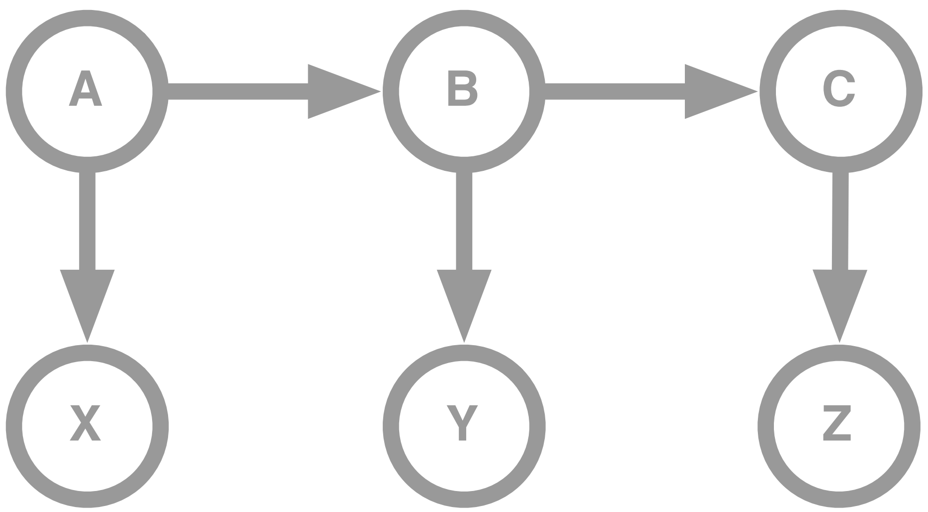
$$p(\mathbf{e} \mid \mathbf{f}, m) = \sum_{\mathbf{a} \in [0,n]^m} p(\mathbf{a} \mid \mathbf{f}, m) \times \prod_{i=1}^{m} p(e_i \mid f_{a_i})$$

The problem of word alignment is as:

$$\mathbf{a}^* = \arg \max_{\mathbf{a} \in [0,n]^m} \boxed{p(\mathbf{a} \mid \mathbf{e}, \mathbf{f}, m)}$$

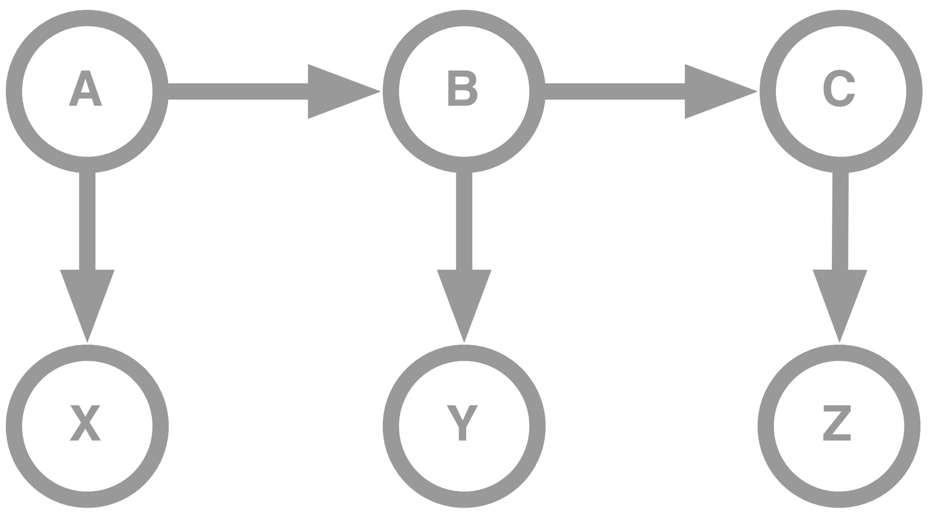
Can we model this distribution directly?

Markov Random Fields (MRFs)

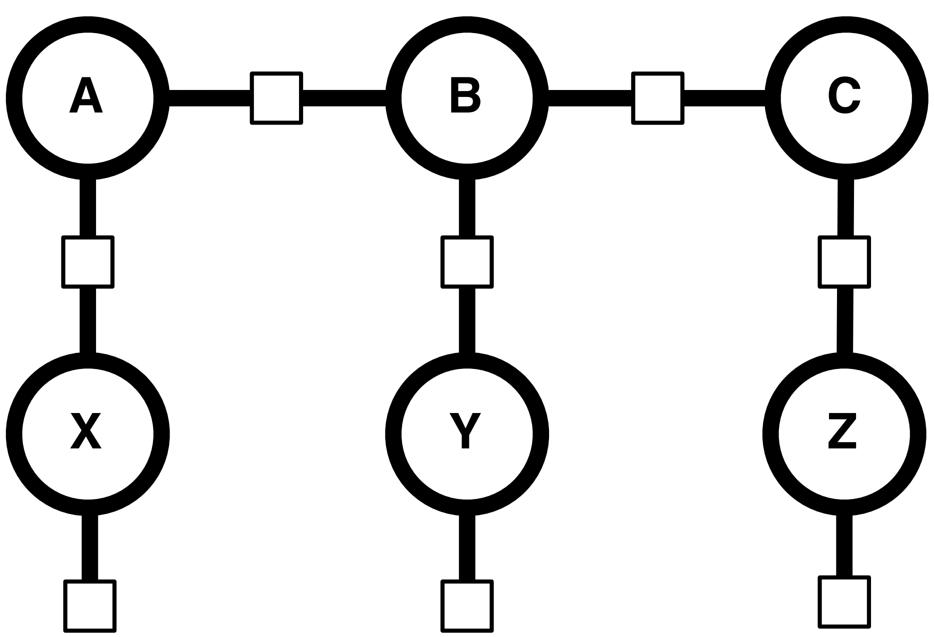


$$
\begin{aligned}
&p(A,B,C,X,Y,Z) = \\
&\quad p(A) \times p(B \mid A) \times p(C \mid B) \times \\
&\quad\quad p(X \mid A) p(Y \mid B) p(Z \mid C)
\end{aligned}
$$

Markov Random Fields (MRFs)

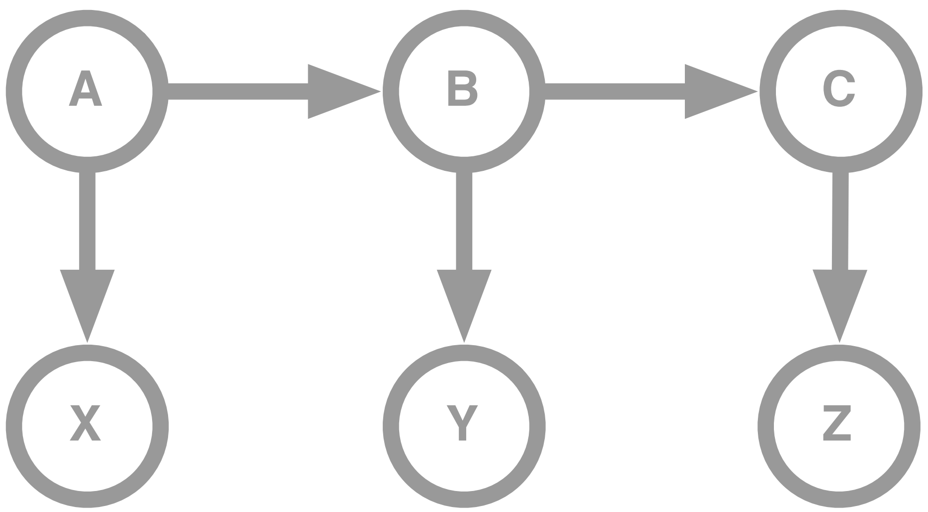


$$p(A, B, C, X, Y, Z) =$$
$$p(A) \times p(B \mid A) \times p(C \mid B) \times$$
$$p(X \mid A) p(Y \mid B) p(Z \mid C)$$

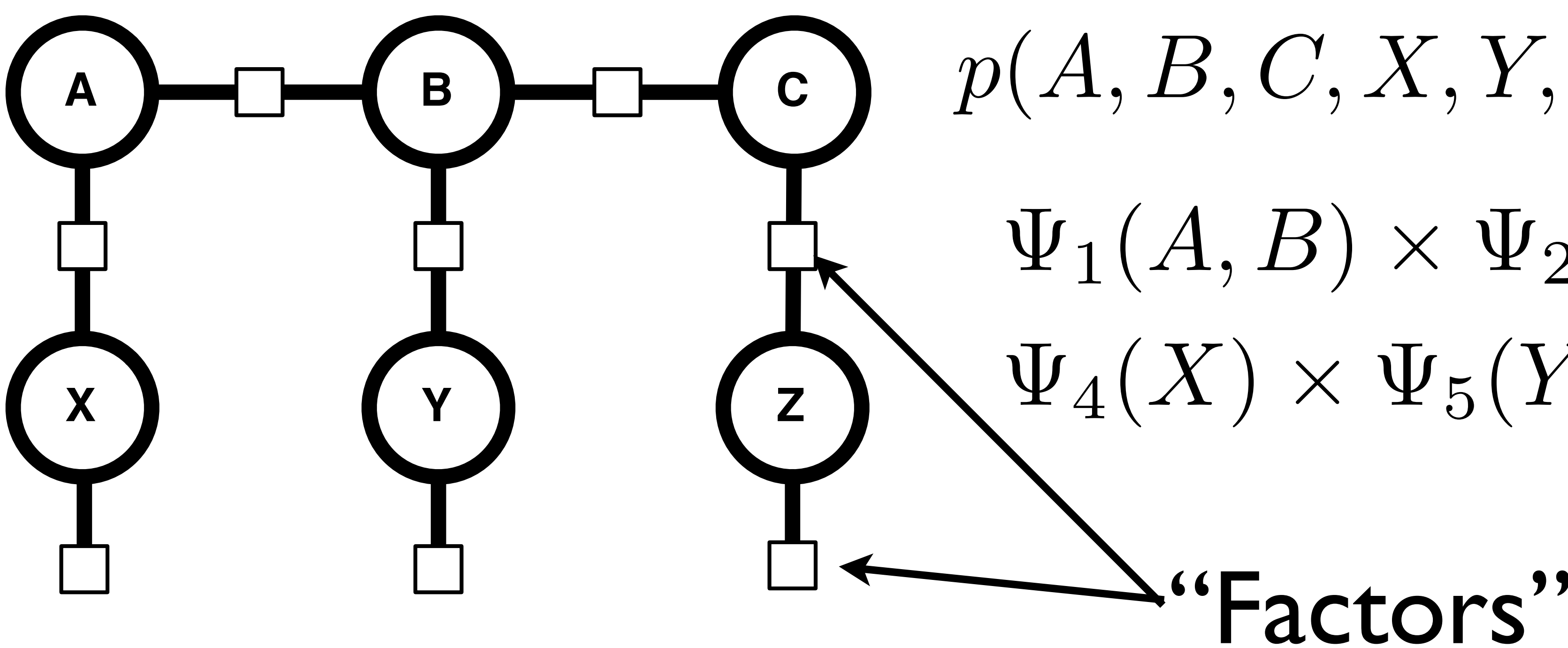


$$p(A, B, C, X, Y, Z) = \frac{1}{Z} \times$$
$$\Psi_1(A, B) \times \Psi_2(B, C) \times \Psi_3(C, D) \times$$
$$\Psi_4(X) \times \Psi_5(Y) \times \Psi_6(Z)$$

Markov Random Fields (MRFs)

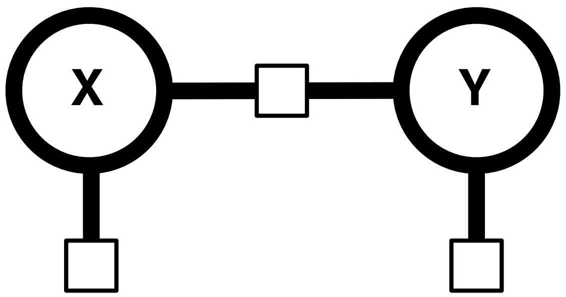


$$p(A, B, C, X, Y, Z) = p(A) \times p(B \mid A) \times p(C \mid B) \times p(X \mid A) p(Y \mid B) p(Z \mid C)$$



$$p(A, B, C, X, Y, Z) = \frac{1}{Z} \times \Psi_1(A, B) \times \Psi_2(B, C) \times \Psi_3(C, D) \times \Psi_4(X) \times \Psi_5(Y) \times \Psi_6(Z)$$

Computing Z



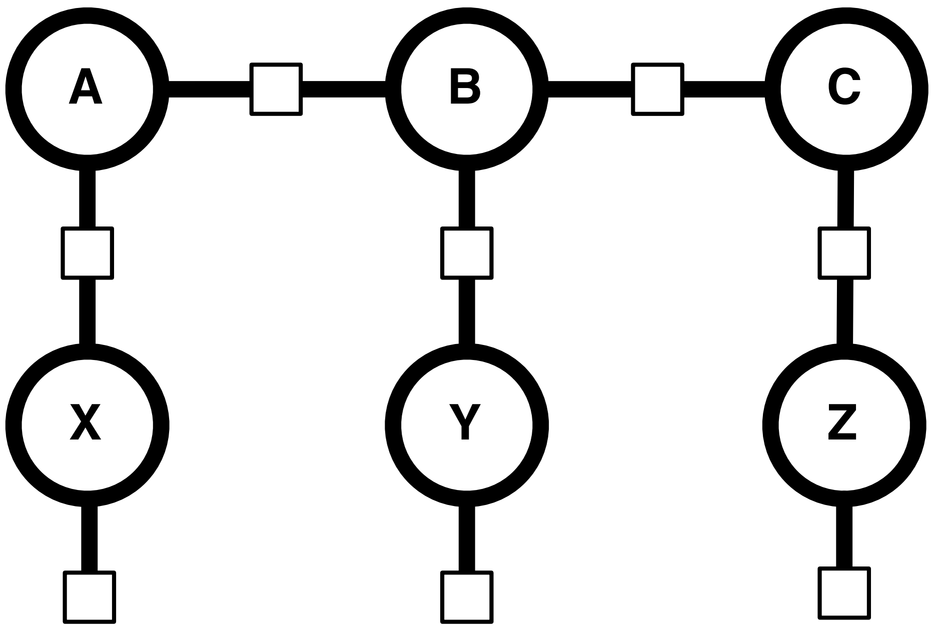
$$\mathcal{X} = \{\texttt{a}, \texttt{b}, \texttt{c}\}$$
$$X \in \mathcal{X}$$
$$Y \in \mathcal{X}$$

$$Z = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{X}} \Psi_1(x, y)\Psi_2(x)\Psi_3(y)$$

When the graph has certain structures (e.g., chains), you can factor to get polytime DP algorithms.

$$Z = \sum_{x \in \mathcal{X}} \Psi_2(x) \sum_{y \in \mathcal{X}} \Psi_1(x, y)\Psi_3(y)$$

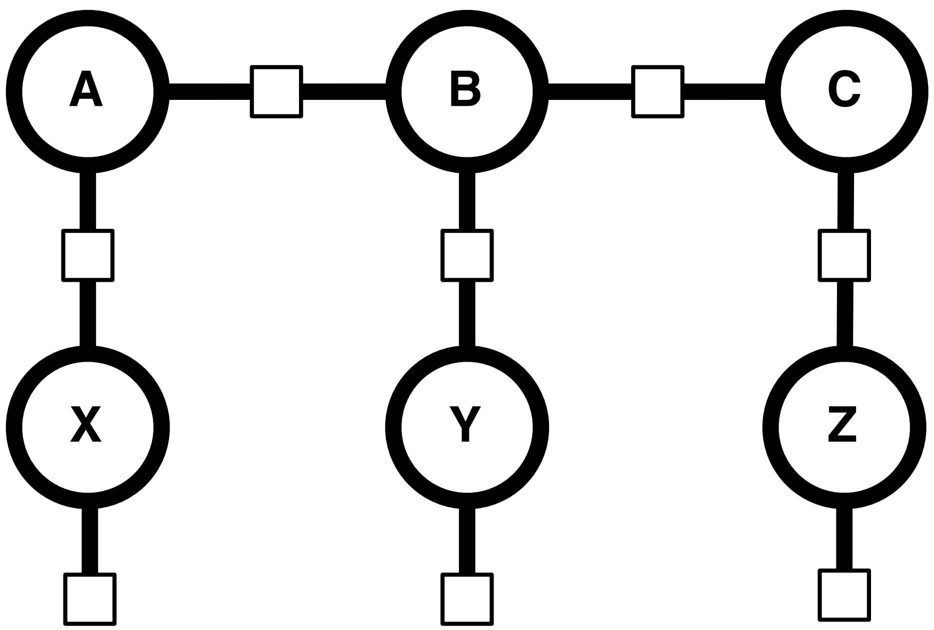
Log-linear models



$$p(A, B, C, X, Y, Z) = \frac{1}{Z} \times$$
$$\Psi_1(A, B) \times \Psi_2(B, C) \times \Psi_3(C, D) \times$$
$$\Psi_4(X) \times \Psi_5(Y) \times \Psi_6(Z)$$

$$\Psi_{1,2,3}(x, y) = \exp \sum_k w_k f_k(x, y)$$

Log-linear models

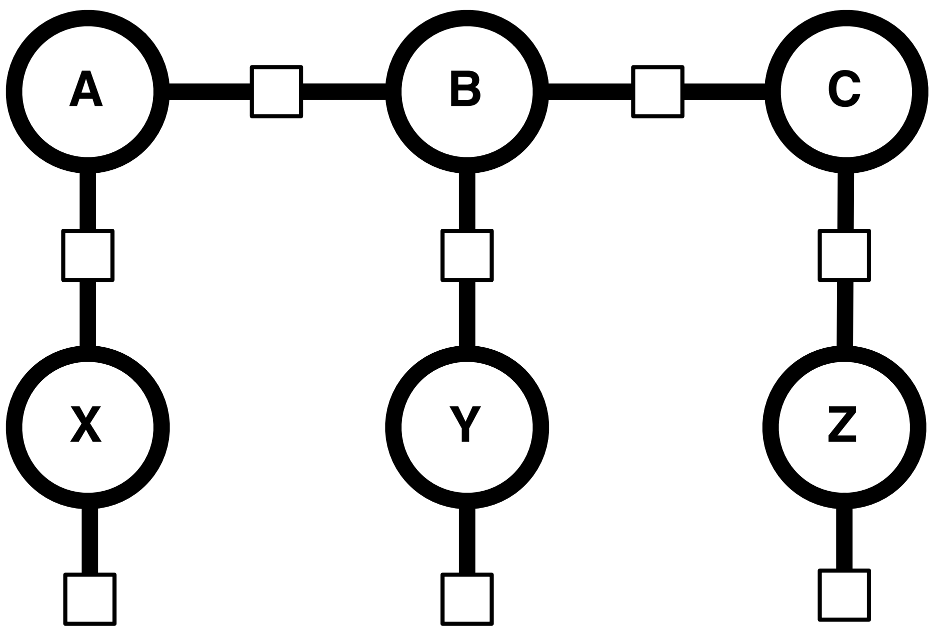


$$p(A, B, C, X, Y, Z) = \frac{1}{Z} \times$$
$$\Psi_1(A, B) \times \Psi_2(B, C) \times \Psi_3(C, D) \times$$
$$\Psi_4(X) \times \Psi_5(Y) \times \Psi_6(Z)$$

$$\Psi_{1,2,3}(x, y) = \exp \sum_k w_k f_k(x, y)$$

Weights (learned)

Log-linear models



$$p(A,B,C,X,Y,Z) = \frac{1}{Z} \times$$
$$\Psi_1(A,B) \times \Psi_2(B,C) \times \Psi_3(C,D) \times$$
$$\Psi_4(X) \times \Psi_5(Y) \times \Psi_6(Z)$$

$$\Psi_{1,2,3}(x,y) = \exp \sum_k w_k f_k(x,y)$$

Weights (learned)

Feature functions (specified)

Random Fields

- **Benefits**
  - Potential functions can be defined with respect to arbitrary features (functions) of the variables
  - Great way to incorporate knowledge
- **Drawbacks**
  - Likelihood involves computing Z
  - Maximizing likelihood usually requires computing Z (often over and over again!)

Conditional Random Fields

- Use MRFs to parameterize a conditional distribution. Very easy: let feature functions look at **anything** they want in the "input"

Conditional Random Fields

- Use MRFs to parameterize a conditional distribution. Very easy: let feature functions look at **anything** they want in the "input"

$$p(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z_{\mathbf{w}}(\mathbf{y})} \exp \sum_{F \in \mathcal{G}} \sum_{k} w_k f_k(F, \mathbf{x})$$

Conditional Random Fields

- Use MRFs to parameterize a conditional distribution. Very easy: let feature functions look at **anything** they want in the "input"

$$p(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z_{\mathbf{w}}(\mathbf{y})} \exp \sum_{F \in \mathcal{G}} \sum_{k} w_k f_k(F, \mathbf{x})$$

All factors in the graph of $\mathbf{y}$

Parameter Learning

- CRFs are trained to maximize conditional likelihood

$$\hat{\mathbf{w}}_{\text{MLE}} = \arg\max_{\mathbf{w}} \prod_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{D}} p(\mathbf{y}_i \mid \mathbf{x}_i \; ; \mathbf{w})$$

- Recall we want to directly model

$$p(\mathbf{a} \mid \mathbf{e}, \mathbf{f})$$

- The likelihood of what alignments?

Parameter Learning

- CRFs are trained to maximize conditional likelihood

$$\hat{\mathbf{w}}_{\mathrm{MLE}} = \arg\max_{\mathbf{w}} \prod_{(\mathbf{x}_i, \mathbf{y}_i) \in \mathcal{D}} p(\mathbf{y}_i \mid \mathbf{x}_i \, ; \mathbf{w})$$

- Recall we want to directly model

$$p(\mathbf{a} \mid \mathbf{e}, \mathbf{f})$$

- The likelihood of what alignments?

Gold reference alignments!

CRF for Alignment

- One of many possibilities, due to Blunsom & Cohn (2006)

$$p(\mathbf{a} \mid \mathbf{e}, \mathbf{f}) = \frac{1}{Z_{\mathbf{w}}(\mathbf{e}, \mathbf{f})} \exp \sum_{i=1}^{|\mathbf{e}|} \sum_{k} w_k f(a_i, a_{i-1}, i, \mathbf{e}, \mathbf{f})$$

- a has the same form as in the lexical translation models (still make a one-to-many assumption)
- w_k are the model parameters
- f_k are the feature functions

CRF for Alignment

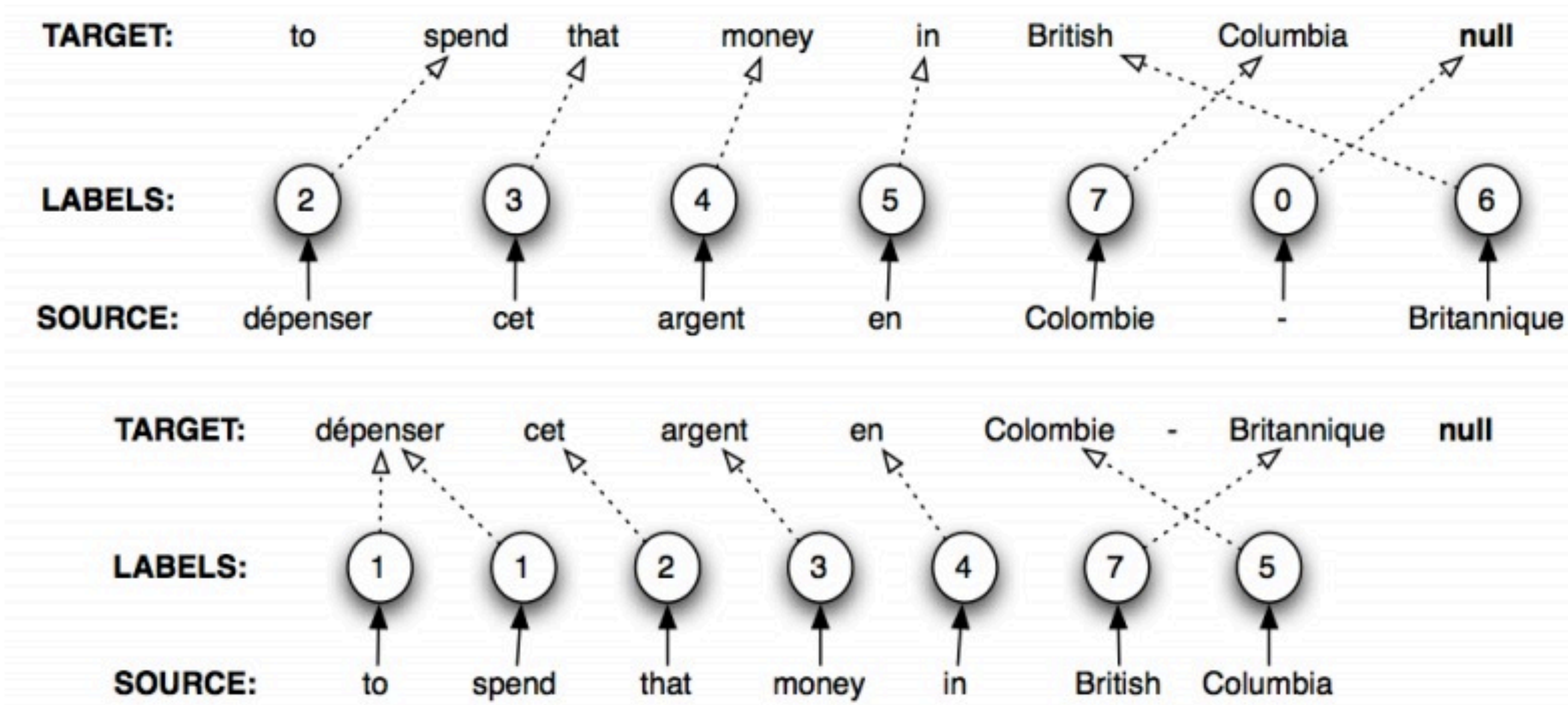
- One of many possibilities, due to Blunsom & Cohn (2006)

$$p(\mathbf{a} \mid \mathbf{e}, \mathbf{f}) = \frac{1}{Z_{\mathbf{w}}(\mathbf{e}, \mathbf{f})} \exp \sum_{i=1}^{|\mathbf{e}|} \sum_{k} w_k f(a_i, a_{i-1}, i, \mathbf{e}, \mathbf{f})$$

- a has the same form as in the lexical translation models (still make a one-to-many assumption)
- w_k are the model parameters
- f_k are the feature functions

$$\boxed{O(n^2 m) \approx O(n^3)}$$

Model



- Labels (one per target word) index source sentence
- Train model (e,f) and (f,e) [inverting the reference alignments]

Experiments

Alignment Experiments:

- French - English (Canadian Hansards corpus NAACL '03)
- 484 word-aligned sentences (100 training, 37 devel. and 347 testing)
- 1.1M sentence-aligned sentences
- We present GIZA++ model 4 results for comparison

pervez musharrafs langer abschied

Identical word

pervez musharraf ’s long goodbye

Identical word

pervez **musharrafs** langer abschied

Matching prefix

pervez **musharraf** ’s long goodbye

Identical word

Matching prefix

pervez musharrafs langer abschied

Matching suffix

pervez musharraf 's long goodbye

Identical word

Matching prefix

Matching suffix

pervez musharrafs $\boxed{\text{langer}}$ abschied

Orthographic similarity

pervez musharraf 's $\boxed{\text{long}}$ goodbye

Identical word

Matching prefix

Matching suffix

Orthographic similarity

pervez musharrafs langer abschied

In dictionary

pervez musharraf 's long goodbye

Identical word

Matching prefix

Matching suffix

Orthographic similarity

In dictionary

...

pervez musharrafs langer abschied

pervez musharraf 's long goodbye

Identical word

Matching prefix

Matching suffix

Orthographic similarity

In dictionary

...

Lexical Features

- Word $\leftrightarrow$ word indicator features
- Various word $\leftrightarrow$ word co-occurrence scores
  - IBM Model 1 probabilities (t→s , s→t)
  - Geometric mean of Model 1 probabilities
  - Dice's coefficient [binned]
  - Products of the above

Lexical Features

- Word class ↔ word class indicator
  - **NN** translates as **NN** (`NN_NN=1`)
  - **NN** does not translate as **MD** (`NN_MD=1`)
- Identical word feature
  - **2010 = 2010** (`IdentWord=1 IdentNum=1`)
- Identical prefix feature
  - **Obama ~ Obamu** (`IdentPrefix=1`)
- Orthographic similarity measure [binned]
  - **Al-Qaeda ~ Al-Kaida** (`OrthoSim050_080=1`)

Other Features

- Compute features from large amounts of unlabeled text
  - Does the Model 4 alignment contain this alignment point?
  - What is the Model 1 posterior probability of this alignment point?

Results

Alignment Results:

	Precision	Recall	F-score
French $\rightarrow$ English	0.97	0.86	0.91
French $\leftarrow$ English	**0.98**	0.83	0.91
French $\leftrightarrow$ English	0.96	0.90	0.93
French $\rightarrow$ English (+ibm model4)	**0.98**	0.88	0.93
French $\leftarrow$ English (+ibm model4)	**0.98**	0.87	0.93
French $\leftrightarrow$ English (+ibm model4)	**0.98**	0.91	**0.95**
GIZA++ (French $\leftrightarrow$ English)	0.87	**0.95**	0.91

Summary

- CRFs provide an efficient model for word alignment that outperforms current models, even when only a small number of word aligned sentences are available
- A diverse range of features can be beneficial to word alignment performance, in particular Markov sequence features improve f-score
- Incorporating features from unsupervised models such as IBM Model 4 can lead to a large increase in f-score

Summary

- CRFs provide an efficient model for word alignment that outperforms current models, even when only a small number of word aligned sentences are available
- A diverse range of features can be beneficial to word alignment performance, in particular Markov sequence features improve f-score
- Incorporating features from unsupervised models such as IBM Model 4 can lead to a large increase in f-score

Unfortunately, you need **gold alignments**!

Putting the pieces together

$$p(\mathbf{e})$$

$$p(\mathbf{e} \mid \mathbf{f}, m)$$

$$p(\mathbf{e}, \mathbf{a} \mid \mathbf{f}, m)$$

$$p(\mathbf{a} \mid \mathbf{e}, \mathbf{f})$$

Putting the pieces together

- We have seen how to model the following:

$$p(\mathbf{e})$$

$$p(\mathbf{e} \mid \mathbf{f}, m)$$

$$p(\mathbf{e}, \mathbf{a} \mid \mathbf{f}, m)$$

$$p(\mathbf{a} \mid \mathbf{e}, \mathbf{f})$$

Putting the pieces together

- We have seen how to model the following:

$$p(\mathbf{e})$$

$$\boxed{p(\mathbf{e} \mid \mathbf{f}, m)}$$

$$p(\mathbf{e}, \mathbf{a} \mid \mathbf{f}, m)$$

$$p(\mathbf{a} \mid \mathbf{e}, \mathbf{f})$$

Putting the pieces together

- We have seen how to model the following:

$$p(\mathbf{e})$$

$$\boxed{p(\mathbf{e} \mid \mathbf{f}, m)}$$

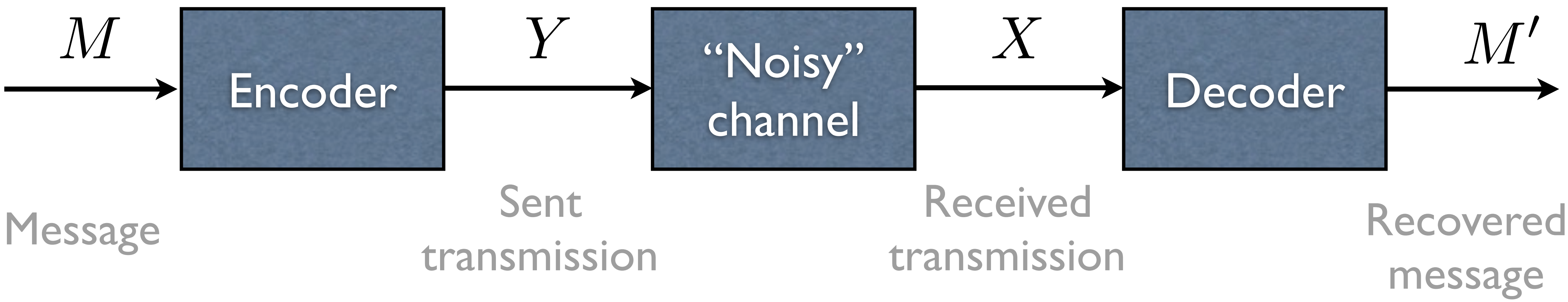
$$p(\mathbf{e}, \mathbf{a} \mid \mathbf{f}, m)$$

$$p(\mathbf{a} \mid \mathbf{e}, \mathbf{f})$$

- Goal: a better model of $p(\mathbf{e} \mid \mathbf{f}, m)$ that knows about $p(\mathbf{e})$

One naturally wonders if the problem of translation could conceivably be treated as a problem in cryptography. When I look at an article in Russian, I say: '***This is really written in English, but it has been coded in some strange symbols. I will now proceed to decode.***'

Warren Weaver to Norbert Wiener, March, 1947



Claude Shannon. "A Mathematical Theory of Communication" 1948.

$$p(y) \quad p(x|y) \quad p(x)$$

Claude Shannon. "A Mathematical Theory of Communication" 1948.

$$\boxed{p(y) \quad p(x|y)} \quad p(x)$$

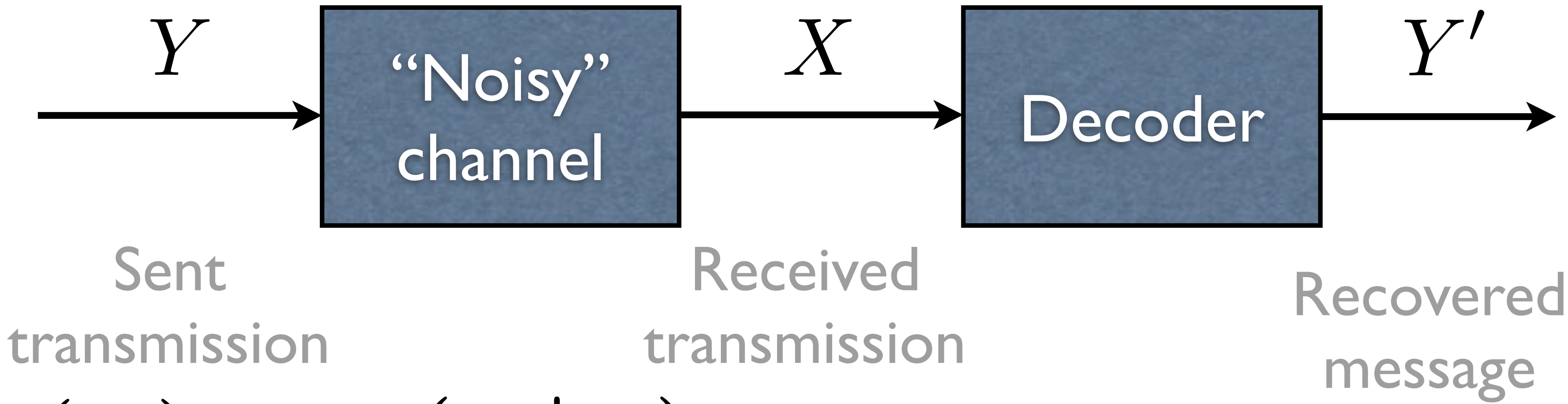
Claude Shannon. "A Mathematical Theory of Communication" 1948.

$$p(y) \quad p(x|y)$$

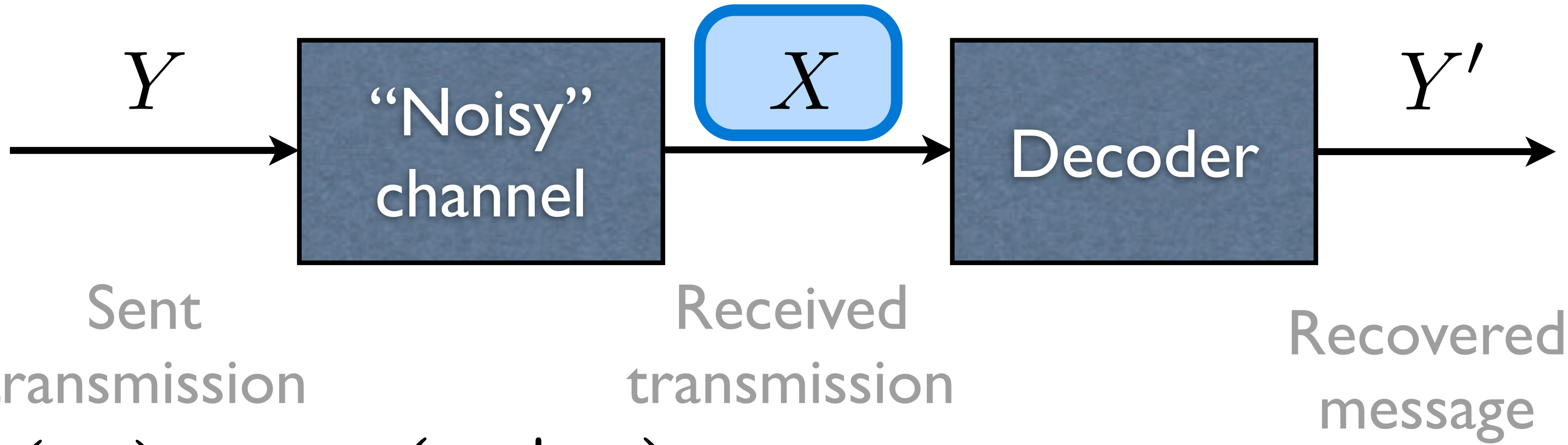
Shannon's theory tells us:

1) how much data you can send
2) the limits of compression
3) why your download is so slow
4) how to translate

Claude Shannon. "A Mathematical Theory of Communication" 1948.

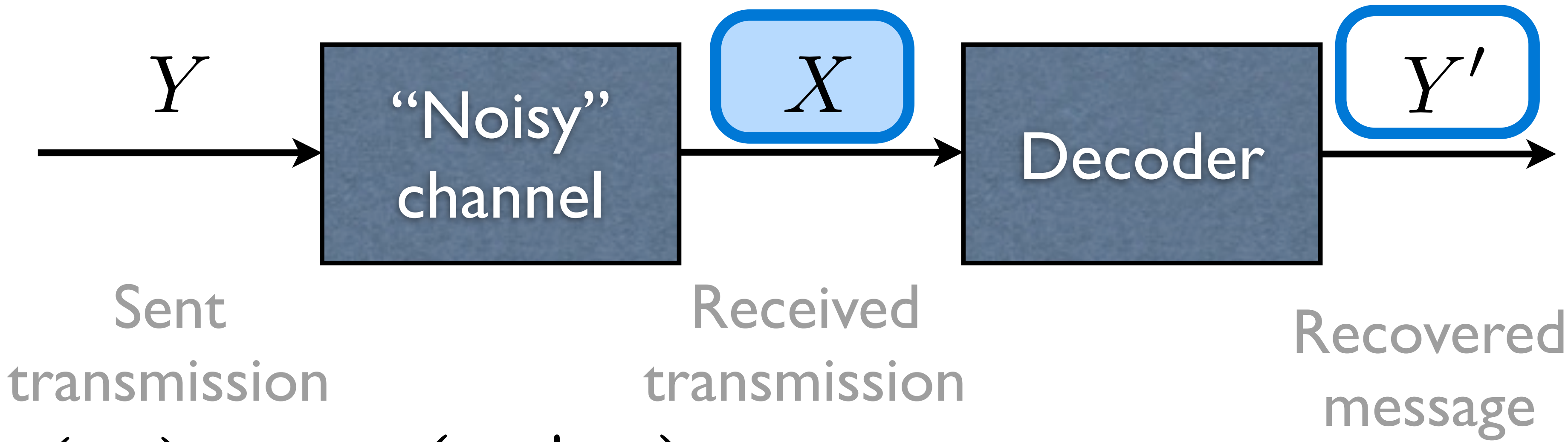

Y
"Noisy" channel
X
Decoder
Y'
Sent transmission
Received transmission
Recovered message

$$p(y) \quad p(x|y)$$

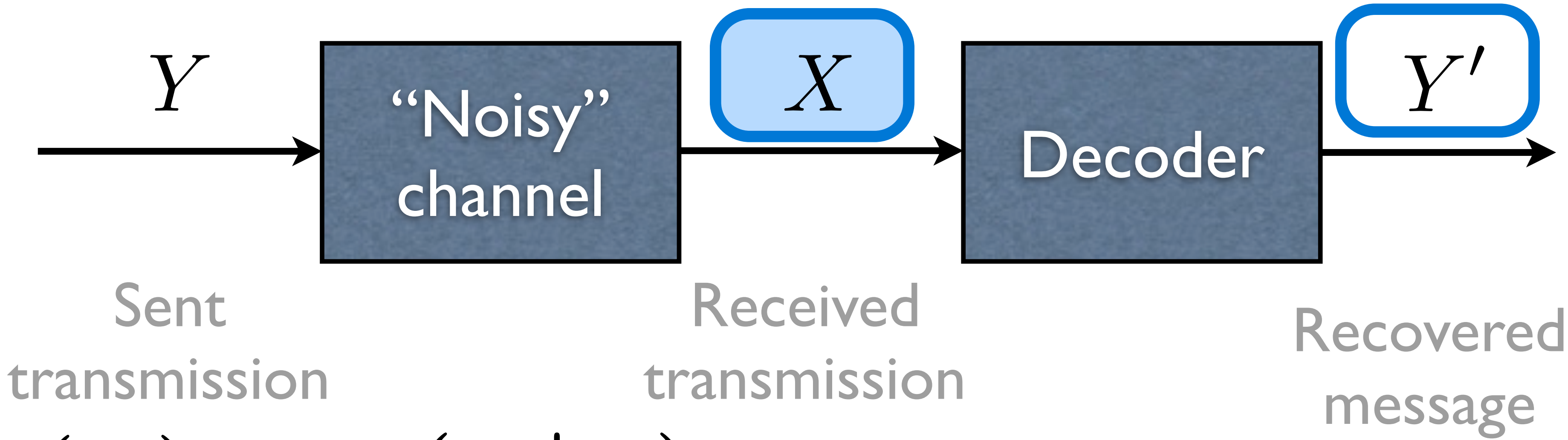


Y
"Noisy" channel
X
Decoder
Y'
Sent transmission
Received transmission
Recovered message

$$p(y) \quad p(x|y)$$

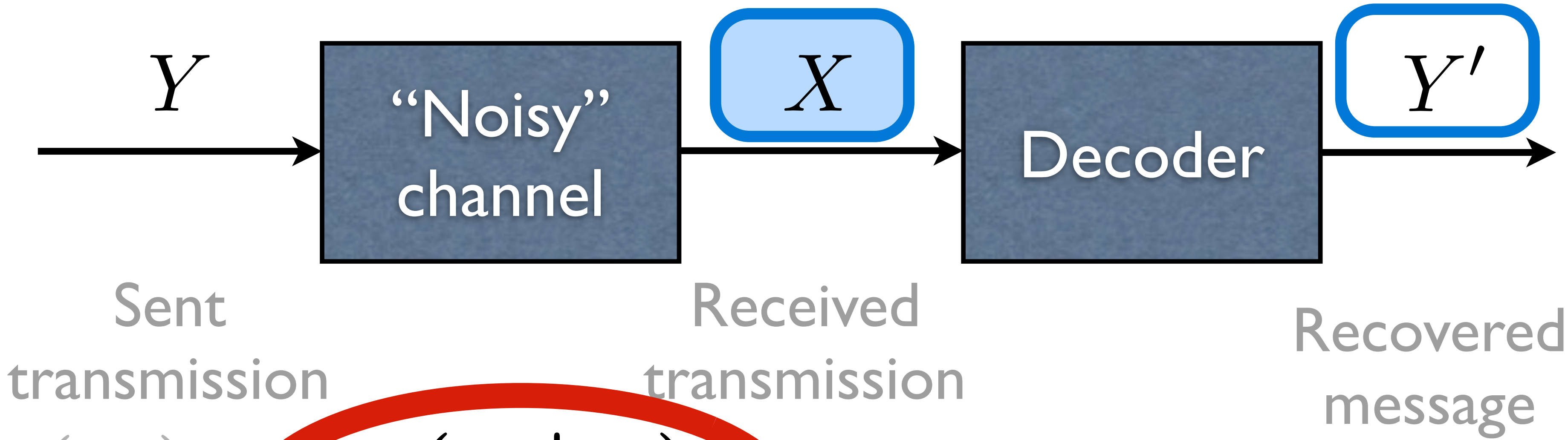

Y
“Noisy” channel
X
Decoder
Y'
Sent transmission
Received transmission
Recovered message

$$p(y) \quad p(x|y)$$



$$p(y) \quad p(x|y)$$

$$\boxed{y'} = \arg\max_y p(y|x)$$



$$p(y)\,p(x|y) \not\propto$$

$$\boxed{y'} = \arg\max_y p(y|x)$$

Y → "Noisy" channel → X → Decoder → Y'

Sent transmission

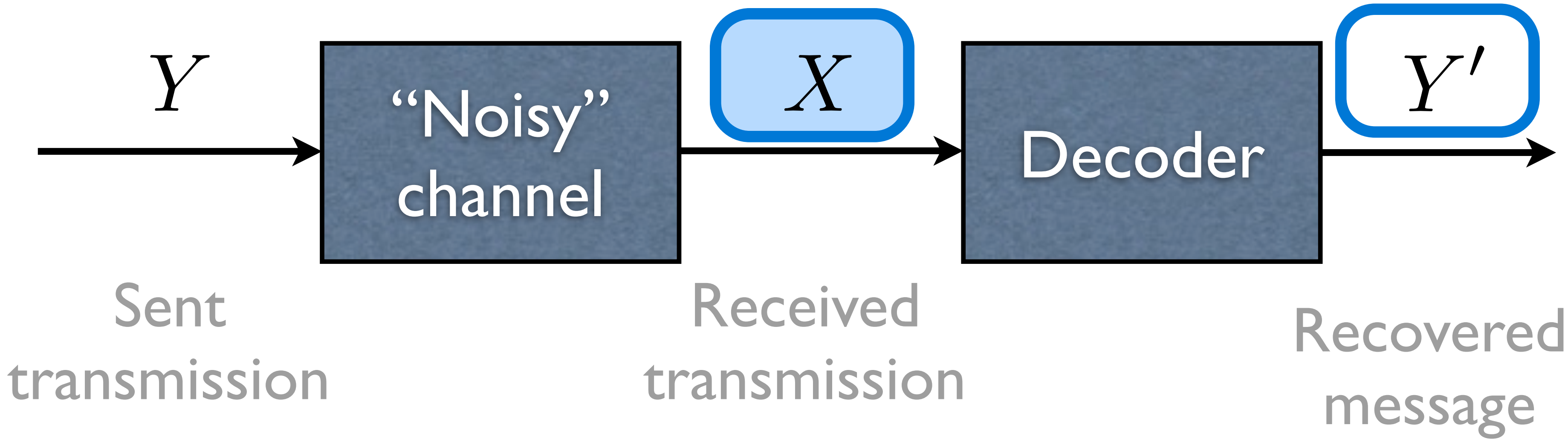
Received transmission

Recovered message

$p(y)$ $p(x|y)$

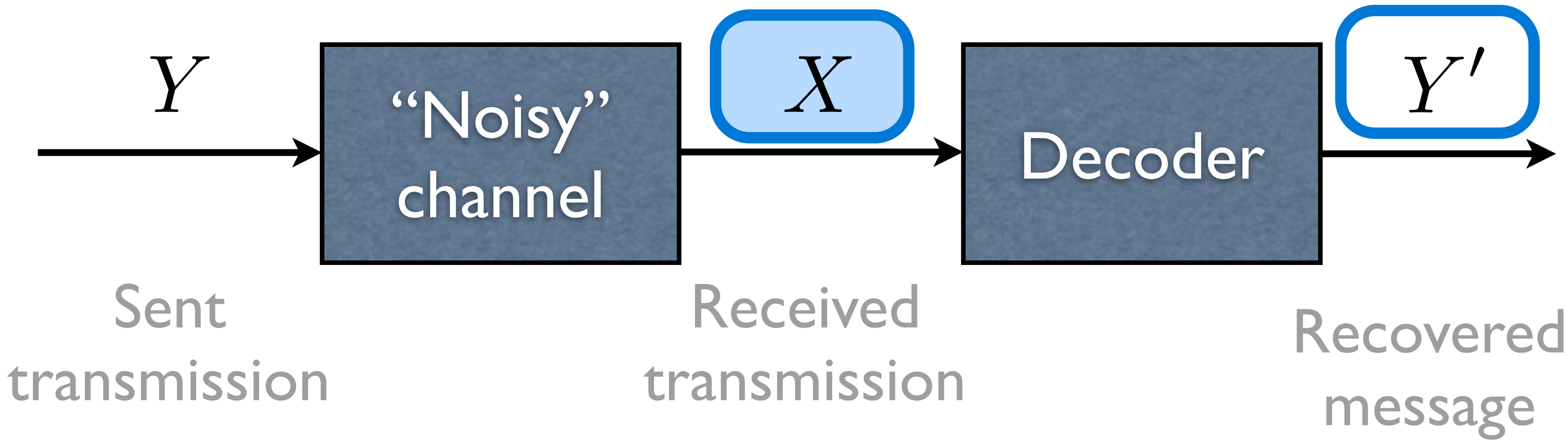
$$\boxed{y'} = \arg\max_{y} p(y|x)$$

I can help.



$$\boxed{y'} = \arg\max_y p(y|x)$$

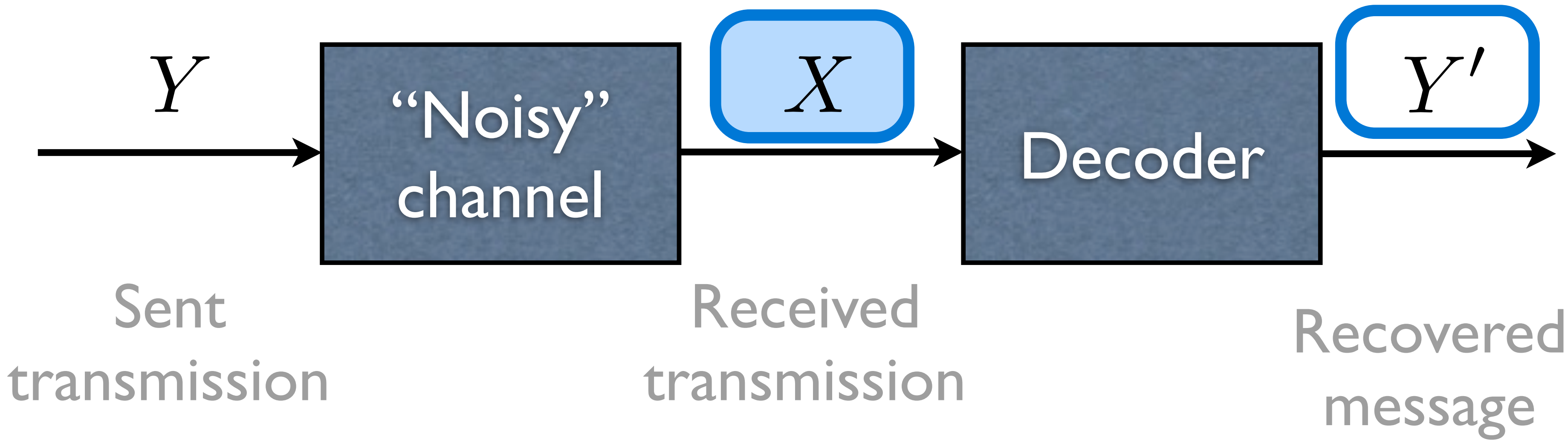
$$= \arg\max_y \frac{p(x|y)p(y)}{p(x)}$$



$$\boxed{y'} = \arg\max_y p(y|x)$$

$$= \arg\max_y \frac{p(x|y)p(y)}{p(x)}$$

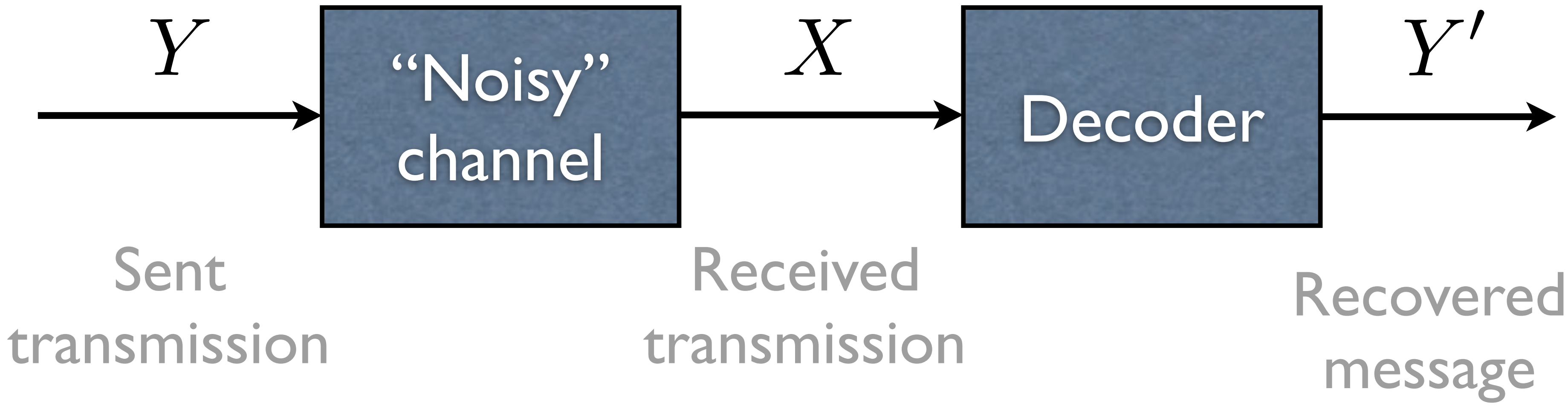
Denominator doesn't depend on y.



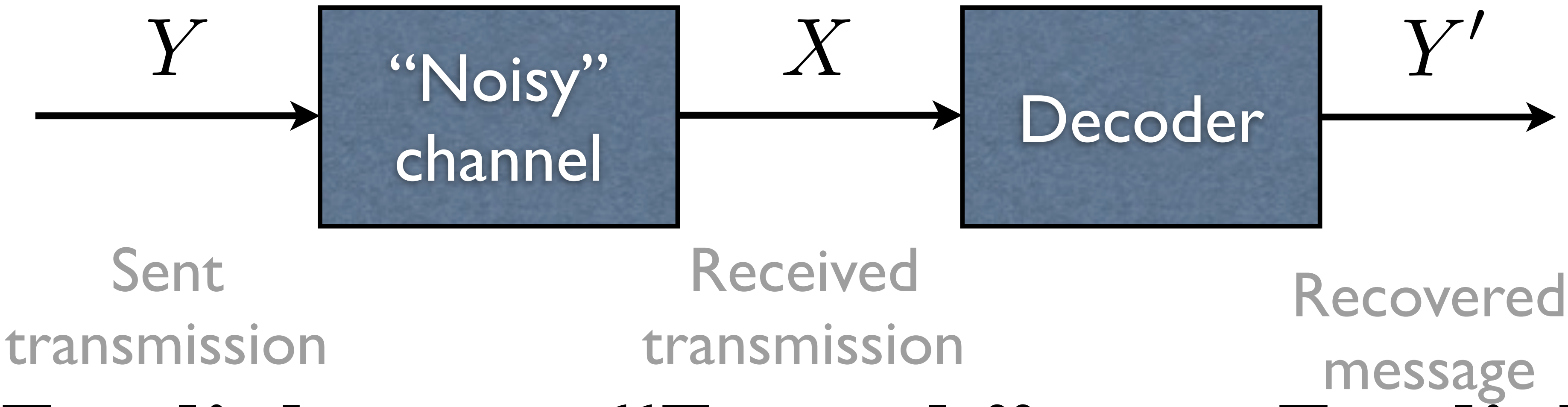
$$\boxed{y'} = \arg\max_y p(y|x)$$

$$= \arg\max_y \frac{p(x|y)p(y)}{p(x)}$$

$$= \arg\max_y p(x|y)p(y)$$



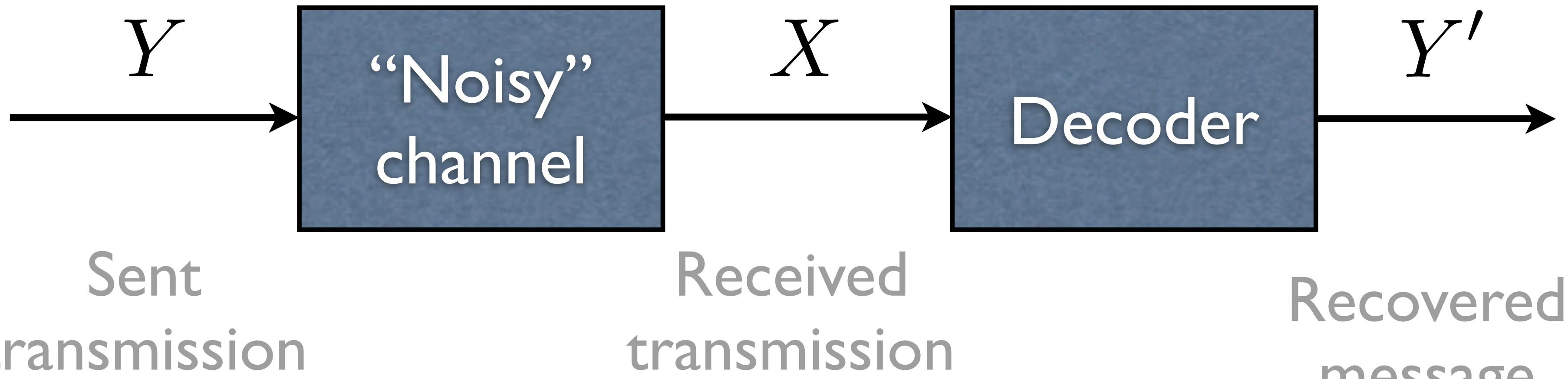
$$y' = \arg\max_y p(x|y)p(y)$$



English **"French"** **English'**

$$\cancel{y' = \arg\max_{y} p(x|y)p(y)}$$

$$\mathbf{e}' = \arg\max_{\mathbf{e}} p(\mathbf{f}|\mathbf{e})p(\mathbf{e})$$

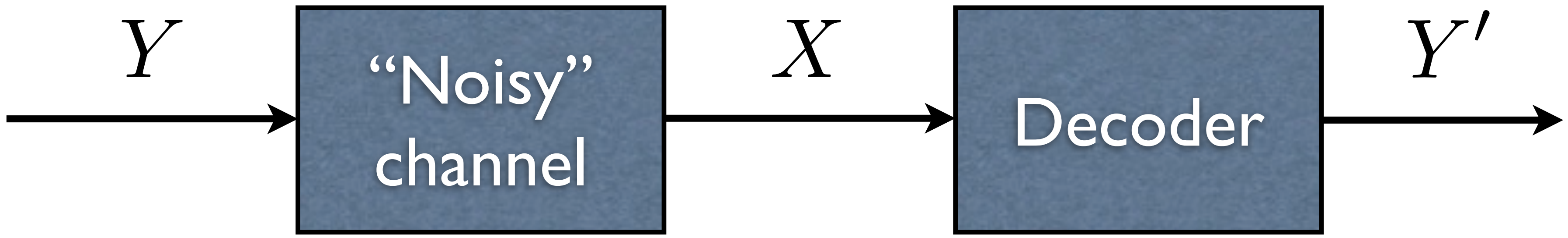


English **"French"** **English'**

$$\cancel{y' = \arg\max_y p(x|y)p(y)}$$

$$\mathbf{e}' = \arg\max_{\mathbf{e}} p(\mathbf{f}|\mathbf{e})p(\mathbf{e})$$

translation model



Sent transmission

English

Received transmission

"French"

Recovered message

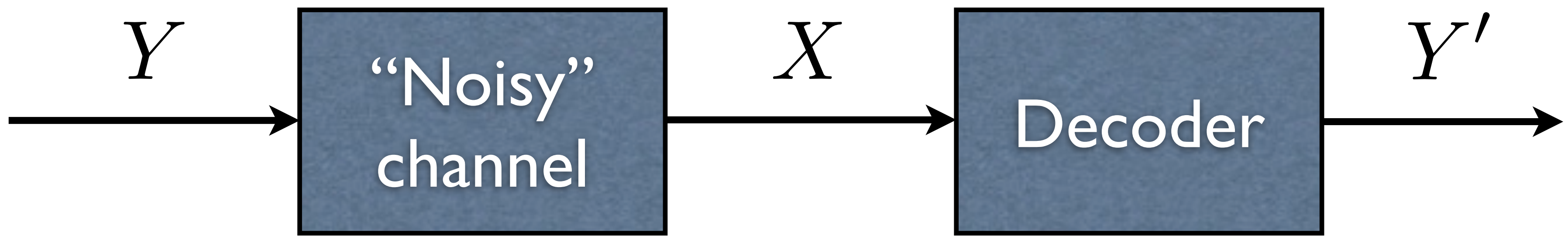
English'

$$\cancel{y' = \arg\max_{y} p(x|y)p(y)}$$

$$\mathbf{e}' = \arg\max_{\mathbf{e}} p(\mathbf{f}|\mathbf{e})p(\mathbf{e})$$

translation model

language model



Sent transmission

English

Received transmission

"French"

Recovered message

English'

$$\cancel{y' = \arg\max_{y} p(x|y)p(y)}$$

$$\mathbf{e}' = \arg\max_{\mathbf{e}} p(\mathbf{f}|\mathbf{e})p(\mathbf{e})$$

translation model

language model

Other noisy channel applications: OCR, speech recognition, spelling correction...

Division of labor

- **Translation model**
  - probability of translation *back* into the source
  - ensures **adequacy** of translation
- **Language model**
  - is a translation hypothesis “good” English?
  - ensures **fluency** of translation

$p(\mathbf{e})$ English

$p(\mathbf{f} \mid \mathbf{e})$

$p(\mathbf{e})$ English $\longrightarrow$ Հայերէն

$p(\mathbf{f} \mid \mathbf{e})$

$p(\mathbf{e})$ English → Հայերէն

$$
\begin{aligned}
\mathbf{e}^* &= \arg\max_{\mathbf{e}} p(\mathbf{e} \mid \mathbf{f}) \\
&= \arg\max_{\mathbf{e}} p(\mathbf{f} \mid \mathbf{e}) \times p(\mathbf{e})
\end{aligned}
$$

Announcements

- Upcoming language-in-10
  - Tuesday: Jon/Austin - Русский
- Leaderboard is functional

		Assignments				
Rank	Handle	#0	#1 AER	#3 Spearman's	#2 model score	#4 BLEU
	oracle	8	0			
1	db	16	0.433932			
	baseline	10	0.434484			
2	zero	18	0.434484			
3	Victor	24	0.438705			
	default	9	0.788911			
4	HBH	10				