

# Grammatical Inference: News from the Machine Translation Front

François Yvon

with the help of G. Adda, M. Adda, A. Allauzen,  
H. Bonneau-Maynard, J. Crego, A. Max & J.L. Gauvain

LIMSI-CNRS & Université Paris-Sud 11

September 22, 2008

# Outline

SMT  
=  
corpora  
+  
machine learning algorithms

# Outline

SMT  
=  
large corpora  
+  
simple machine learning algorithms

# Outline

SMT for restricted domain and look-alike languages

=

large corpora

+

simple machine learning algorithms

# Outline

General SMT  
=  
linguistically analyzed corpora  
+  
structure aware machine learning algorithms

# Some problems with machine translation

Is machine translation possible at all ?

**f=** *Ich werde Ihnen die entsprechenden Anmerkungen  
aushändigen*

**e=** *I will pass on to you the corresponding comments*

# Mainstream Statistical Machine Translation

Introducing Phrase-Based Statistical Machine Translation

# Mainstream Statistical Machine Translation

## Introducing Phrase-Based Statistical Machine Translation

1. take a set of parallel sentences (*bitext*)
  - ▶ align each pair  $(\mathbf{f}, \mathbf{e})$ , word for word
  - ▶ train translation model: the “phrase” table  $\{(f, e)\}$
2. take a set of monolingual texts
3. make sure to tune your system
4. translate  $\mathbf{f}$  = solve

$$\operatorname{argmax}_{\mathbf{e} \in E} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

5. and get some numbers
6. not happy ? goto 1

# Mainstream Statistical Machine Translation

## Introducing Phrase-Based Statistical Machine Translation

1. take a set of parallel sentences (*bitext*)
  - ▶ align each pair  $(\mathbf{f}, \mathbf{e})$ , word for word
  - ▶ **train translation model: the “phrase” table  $\{(f, e)\}$**
2. take a set of monolingual texts
  - ▶ train statistical target language model
3. make sure to tune your system
4. translate  $\mathbf{f}$  = solve

$$\operatorname{argmax}_{\mathbf{e} \in E} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

5. and get some numbers
6. not happy ? goto 1

# Mainstream Statistical Machine Translation

## Introducing Phrase-Based Statistical Machine Translation

1. take a set of parallel sentences (*bitext*)
  - ▶ align each pair  $(\mathbf{f}, \mathbf{e})$ , word for word
  - ▶ train translation model: the “phrase” table  $\{(f, e)\}$
2. take a set of monolingual texts
  - ▶ train statistical target language model
3. make sure to tune your system
4. translate  $\mathbf{f}$  = solve

$$\operatorname{argmax}_{\mathbf{e} \in E} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

5. and get some numbers
6. not happy ? goto 1

# Mainstream Statistical Machine Translation

## Introducing Phrase-Based Statistical Machine Translation

1. take a set of parallel sentences (*bitext*)
  - ▶ align each pair  $(\mathbf{f}, \mathbf{e})$ , word for word
  - ▶ train translation model: the “phrase” table  $\{(f, e)\}$
2. take a set of monolingual texts
  - ▶ **train statistical target language model**
3. make sure to tune your system
4. translate  $\mathbf{f}$  = solve

$$\operatorname{argmax}_{\mathbf{e} \in E} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

5. and get some numbers
6. not happy ? goto 1

# Mainstream Statistical Machine Translation

## Introducing Phrase-Based Statistical Machine Translation

1. take a set of parallel sentences (*bitext*)
  - ▶ align each pair  $(\mathbf{f}, \mathbf{e})$ , word for word
  - ▶ train translation model: the “phrase” table  $\{(f, e)\}$
2. take a set of monolingual texts
  - ▶ train statistical target language model
3. make sure to tune your system
4. translate  $\mathbf{f}$  = solve

$$\operatorname{argmax}_{\mathbf{e} \in E} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

5. and get some numbers
6. not happy ? goto 1

# Mainstream Statistical Machine Translation

## Introducing Phrase-Based Statistical Machine Translation

1. take a set of parallel sentences (*bitext*)
  - ▶ align each pair  $(\mathbf{f}, \mathbf{e})$ , word for word
  - ▶ train translation model: the “phrase” table  $\{(f, e)\}$
2. take a set of monolingual texts
  - ▶ train statistical target language model
3. make sure to tune your system
4. translate  $\mathbf{f}$  = solve

$$\operatorname{argmax}_{\mathbf{e} \in E} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

5. and get some numbers
6. not happy ? goto 1

# Mainstream Statistical Machine Translation

## Introducing Phrase-Based Statistical Machine Translation

1. take a set of parallel sentences (*bitext*)
  - ▶ align each pair  $(\mathbf{f}, \mathbf{e})$ , word for word
  - ▶ train translation model: the “phrase” table  $\{(f, e)\}$
2. take a set of monolingual texts
  - ▶ train statistical target language model
3. make sure to tune your system
4. translate  $\mathbf{f}$  = solve

$$\operatorname{argmax}_{\mathbf{e} \in E} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

5. and get some numbers
6. not happy ? goto 1

# Mainstream Statistical Machine Translation

## Introducing Phrase-Based Statistical Machine Translation

1. take a set of parallel sentences (*bitext*)
  - ▶ align each pair  $(\mathbf{f}, \mathbf{e})$ , word for word
  - ▶ train translation model: the “phrase” table  $\{(f, e)\}$
2. take a set of monolingual texts
  - ▶ train statistical target language model
3. make sure to tune your system
4. translate  $\mathbf{f}$  = solve

$$\operatorname{argmax}_{\mathbf{e} \in E} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

5. and get some numbers
6. not happy ? goto 1

# Take a set of parallel sentences

- ▶ bilingual corpus, per sentence alignment

---

**f=** *Pourquoi donc les producteurs d'armes de l'UE devraient-ils s'enrichir sur le dos de personnes innocentes ?*

**e=** *So why should EU arms producers profit at the expense of innocent people ?*

---

- ▶ Main sources:
  - ▶ documents from multilingual institutions, literature, touristic guides, technical documentations
  - ▶ news, web sites, blogs, speech transcripts
- ▶ Not enough ? Mine *comparable* corpora (eg. [26])

Large corpora available, yet data scarcity still a serious bottleneck

# Take a set of parallel sentences

- ▶ bilingual corpus, per sentence alignment
- ▶ **Main sources:**
  - ▶ documents from multilingual institutions, literature, touristic guides, technical documentations
  - ▶ news, web sites, blogs, speech transcripts
- ▶ Not enough ? Mine *comparable* corpora (eg. [26])

Large corpora available, yet data scarcity still a serious bottleneck

# Take a set of parallel sentences

- ▶ bilingual corpus, per sentence alignment
- ▶ Main sources:
  - ▶ documents from multilingual institutions, literature, touristic guides, technical documentations
  - ▶ news, web sites, blogs, speech transcripts
- ▶ Not enough ? Mine *comparable* corpora (eg. [26])

Large corpora available, yet data scarcity still a serious bottleneck

# Take a set of parallel sentences

- ▶ bilingual corpus, per sentence alignment
- ▶ Main sources:
  - ▶ documents from multilingual institutions, literature, touristic guides, technical documentations
  - ▶ **news, web sites, blogs, speech transcripts**
- ▶ Not enough ? Mine *comparable* corpora (eg. [26])

Large corpora available, yet data scarcity still a serious bottleneck

# Take a set of parallel sentences

- ▶ bilingual corpus, per sentence alignment
- ▶ Main sources:
  - ▶ documents from multilingual institutions, literature, touristic guides, technical documentations
  - ▶ news, web sites, blogs, speech transcripts
- ▶ Not enough ? Mine *comparable corpora* (eg. [26])

Large corpora available, yet data scarcity still a serious bottleneck

# Take a set of parallel sentences

- ▶ bilingual corpus, per sentence alignment
- ▶ Main sources:
  - ▶ documents from multilingual institutions, literature, touristic guides, technical documentations
  - ▶ news, web sites, blogs, speech transcripts
- ▶ Not enough ? Mine *comparable* corpora (eg. [26])

Large corpora available, yet data scarcity still a serious bottleneck

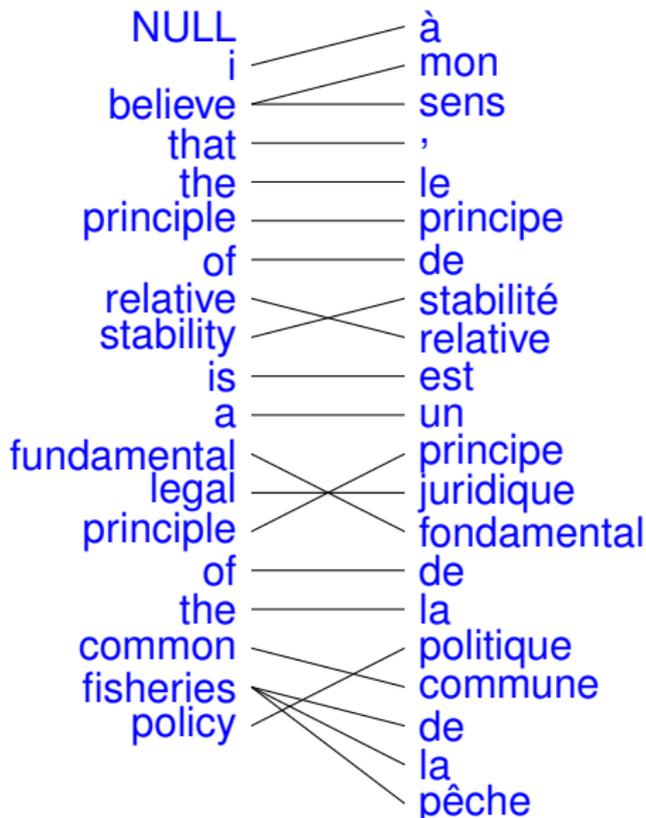
# Take a set of parallel sentences

- ▶ bilingual corpus, per sentence alignment
- ▶ Main sources:
  - ▶ documents from multilingual institutions, literature, touristic guides, technical documentations
  - ▶ news, web sites, blogs, speech transcripts
- ▶ Not enough ? Mine *comparable* corpora (eg. [26])

Large corpora available, yet data scarcity still a serious bottleneck

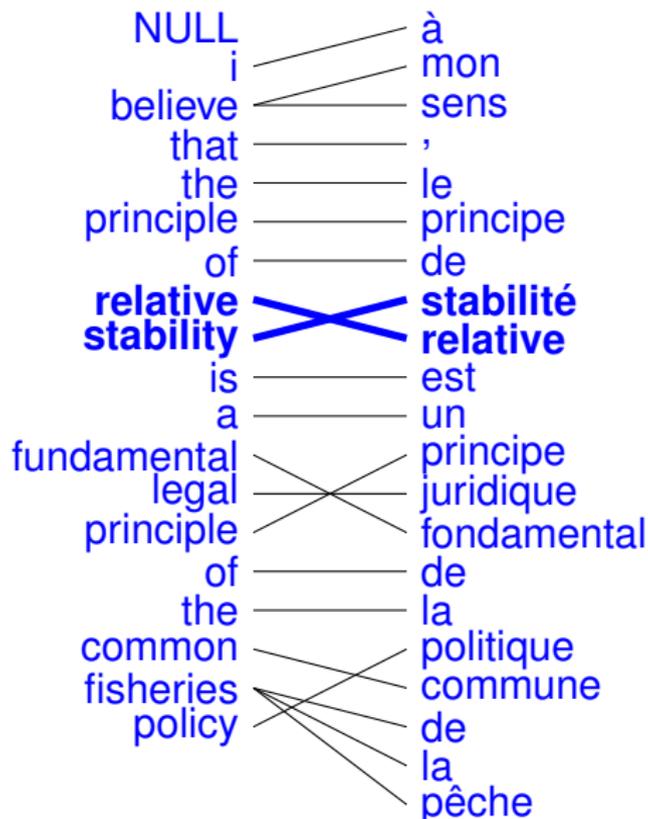
# Training 1.a: build word alignments

Local reordering within the noun phrase



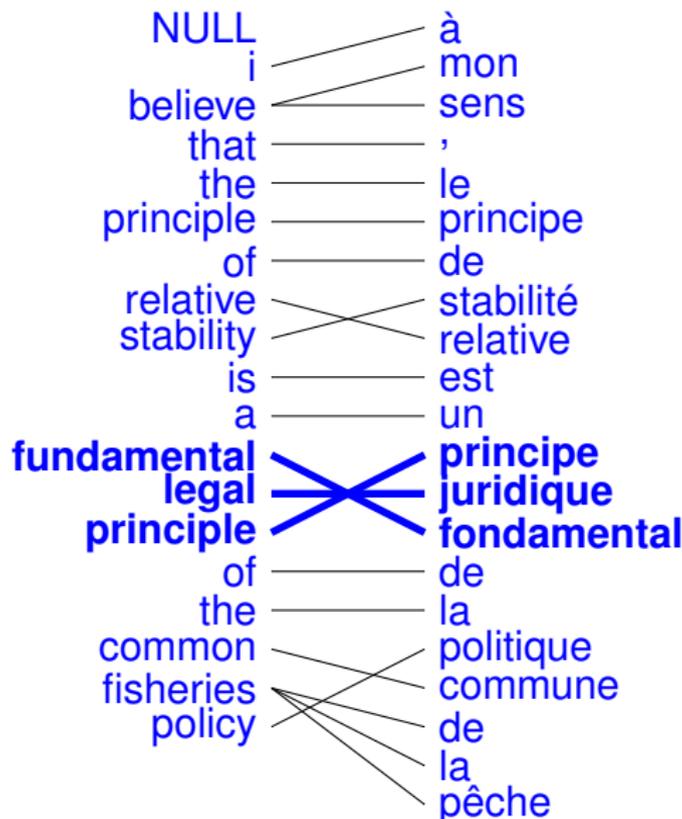
# Training 1.a: build word alignments

Local reordering within the noun phrase



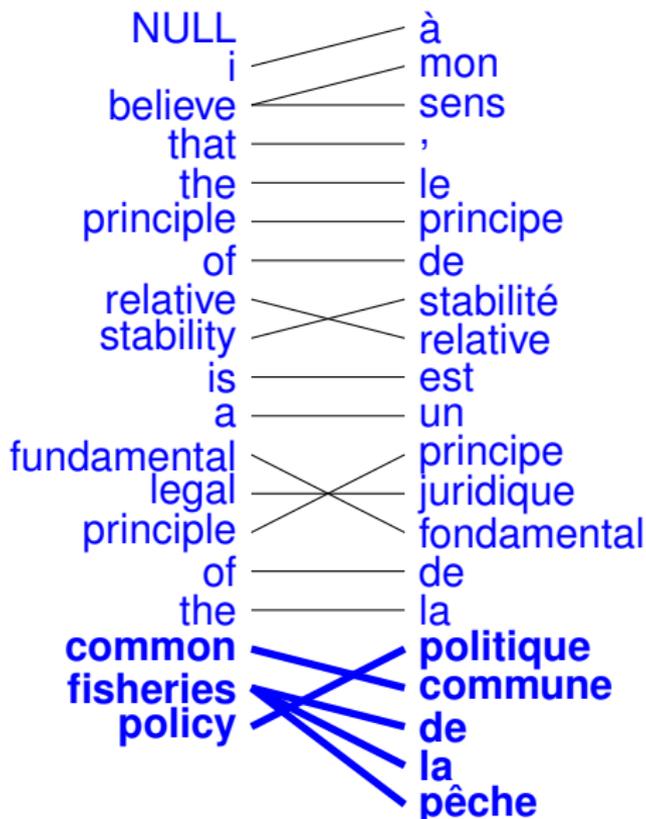
# Training 1.a: build word alignments

Local reordering within the noun phrase



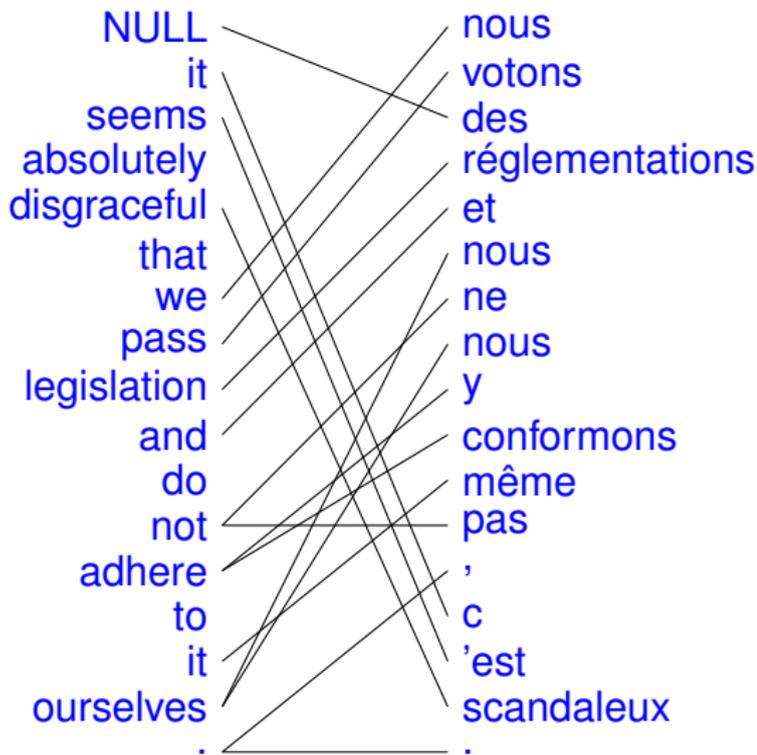
# Training 1.a: build word alignments

Local reordering within the noun phrase



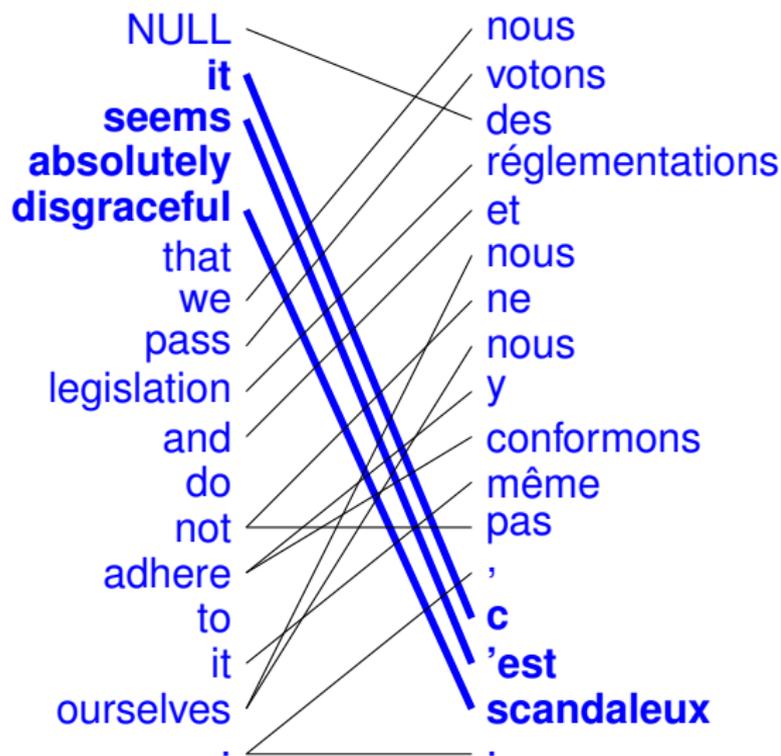
# Training 1.a: build word alignments

A more noisy case



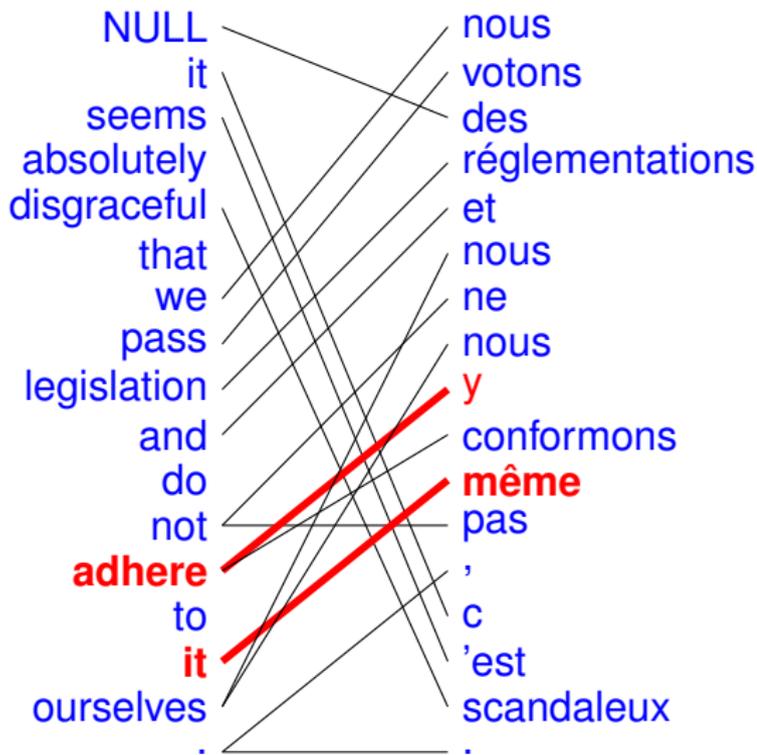
# Training 1.a: build word alignments

A more noisy case



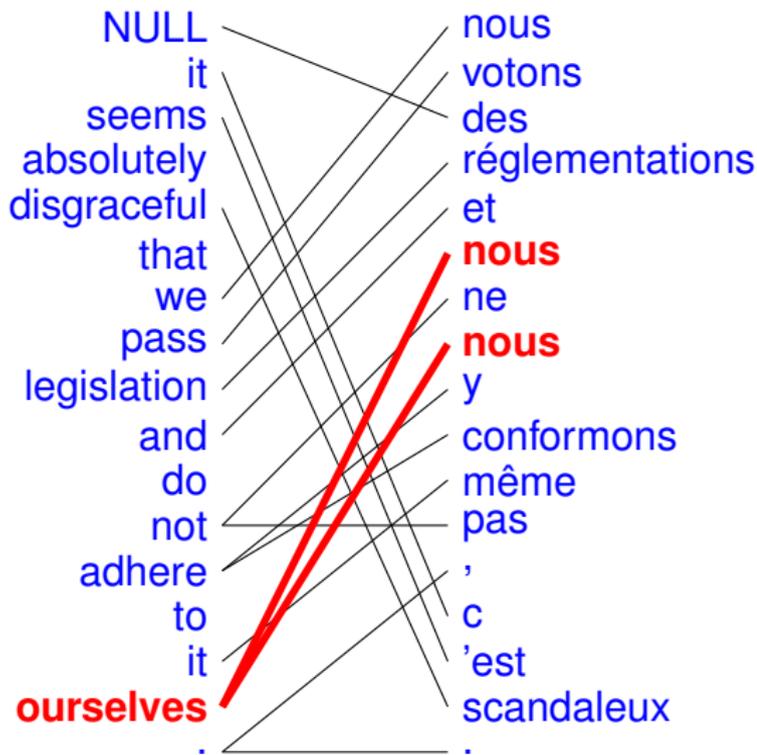
# Training 1.a: build word alignments

A more noisy case



# Training 1.a: build word alignments

A more noisy case



# Training 1.a: build word alignments

- ▶ asymmetric (= many-to-one) alignments (IBM1-IBM5 [6], HMMs [36])
  - ▶ train: estimate  $P(\mathbf{a}, \mathbf{f}|\mathbf{e})$  (EM like)
  - ▶ align:  $\mathbf{a}^* = \operatorname{argmax} P(\mathbf{a}|\mathbf{f}, \mathbf{e}) = \operatorname{argmax} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$
  - ▶ translate:  
$$\mathbf{e}^* = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{f}|\mathbf{e})P(\mathbf{e}) = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{e}) \operatorname{argmax}_{\mathbf{a}} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$$
- ▶ public domain implementations (Giza++ [28] ; MTTK [13])
- ▶ discriminative training (and many more features) helps a bit [24, 1, 4]
- ▶ but supervision data is scarce and unreliable

for asymmetric models, an almost solved issue ?

# Training 1.a: build word alignments

- ▶ asymmetric (= many-to-one) alignments (IBM1-IBM5 [6], HMMs [36])
  - ▶ **train: estimate  $P(\mathbf{a}, \mathbf{f}|\mathbf{e})$  (EM like)**
  - ▶ align:  $\mathbf{a}^* = \operatorname{argmax} P(\mathbf{a}|\mathbf{f}, \mathbf{e}) = \operatorname{argmax} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$
  - ▶ translate:  
$$\mathbf{e}^* = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{f}|\mathbf{e})P(\mathbf{e}) = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{e}) \operatorname{argmax}_{\mathbf{a}} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$$
- ▶ public domain implementations (Giza++ [28] ; MTTK [13])
- ▶ discriminative training (and many more features) helps a bit [24, 1, 4]
- ▶ but supervision data is scarce and unreliable

for asymmetric models, an almost solved issue ?

# Training 1.a: build word alignments

- ▶ asymmetric (= many-to-one) alignments (IBM1-IBM5 [6], HMMs [36])
  - ▶ train: estimate  $P(\mathbf{a}, \mathbf{f}|\mathbf{e})$  (EM like)
  - ▶ align:  $\mathbf{a}^* = \operatorname{argmax} P(\mathbf{a}|\mathbf{f}, \mathbf{e}) = \operatorname{argmax} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$
  - ▶ translate:  
$$\mathbf{e}^* = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{f}|\mathbf{e})P(\mathbf{e}) = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{e}) \operatorname{argmax}_{\mathbf{a}} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$$
- ▶ public domain implementations (Giza++ [28] ; MTTK [13])
- ▶ discriminative training (and many more features) helps a bit [24, 1, 4]
- ▶ but supervision data is scarce and unreliable

for asymmetric models, an almost solved issue ?

# Training 1.a: build word alignments

- ▶ asymmetric (= many-to-one) alignments (IBM1-IBM5 [6], HMMs [36])
  - ▶ train: estimate  $P(\mathbf{a}, \mathbf{f}|\mathbf{e})$  (EM like)
  - ▶ align:  $\mathbf{a}^* = \operatorname{argmax} P(\mathbf{a}|\mathbf{f}, \mathbf{e}) = \operatorname{argmax} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$
  - ▶ translate:  
$$\mathbf{e}^* = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{f}|\mathbf{e})P(\mathbf{e}) = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{e}) \operatorname{argmax}_{\mathbf{a}} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$$
- ▶ public domain implementations (Giza++ [28] ; MTTK [13])
- ▶ discriminative training (and many more features) helps a bit [24, 1, 4]
- ▶ but supervision data is scarce and unreliable

for asymmetric models, an almost solved issue ?

# Training 1.a: build word alignments

- ▶ asymmetric (= many-to-one) alignments (IBM1-IBM5 [6], HMMs [36])
  - ▶ train: estimate  $P(\mathbf{a}, \mathbf{f}|\mathbf{e})$  (EM like)
  - ▶ align:  $\mathbf{a}^* = \operatorname{argmax} P(\mathbf{a}|\mathbf{f}, \mathbf{e}) = \operatorname{argmax} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$
  - ▶ translate:  
$$\mathbf{e}^* = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{f}|\mathbf{e})P(\mathbf{e}) = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{e}) \operatorname{argmax}_{\mathbf{a}} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$$
- ▶ public domain implementations (Giza++ [28] ; MTTK [13])
- ▶ discriminative training (and many more features) helps a bit [24, 1, 4]
- ▶ but supervision data is scarce and unreliable

for asymmetric models, an almost solved issue ?

# Training 1.a: build word alignments

- ▶ asymmetric (= many-to-one) alignments (IBM1-IBM5 [6], HMMs [36])
  - ▶ train: estimate  $P(\mathbf{a}, \mathbf{f}|\mathbf{e})$  (EM like)
  - ▶ align:  $\mathbf{a}^* = \operatorname{argmax} P(\mathbf{a}|\mathbf{f}, \mathbf{e}) = \operatorname{argmax} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$
  - ▶ translate:  
$$\mathbf{e}^* = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{f}|\mathbf{e})P(\mathbf{e}) = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{e}) \operatorname{argmax}_{\mathbf{a}} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$$
- ▶ public domain implementations (Giza++ [28] ; MTTK [13])
- ▶ **discriminative training (and many more features) helps a bit [24, 1, 4]**
- ▶ but supervision data is scarce and unreliable

for asymmetric models, an almost solved issue ?

# Training 1.a: build word alignments

- ▶ asymmetric (= many-to-one) alignments (IBM1-IBM5 [6], HMMs [36])
  - ▶ train: estimate  $P(\mathbf{a}, \mathbf{f}|\mathbf{e})$  (EM like)
  - ▶ align:  $\mathbf{a}^* = \operatorname{argmax} P(\mathbf{a}|\mathbf{f}, \mathbf{e}) = \operatorname{argmax} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$
  - ▶ translate:  
$$\mathbf{e}^* = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{f}|\mathbf{e})P(\mathbf{e}) = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{e}) \operatorname{argmax}_{\mathbf{a}} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$$
- ▶ public domain implementations (Giza++ [28] ; MTTK [13])
- ▶ discriminative training (and many more features) helps a bit [24, 1, 4]
- ▶ but supervision data is scarce and unreliable

for asymmetric models, an almost solved issue ?

## Training 1.a: build word alignments

- ▶ asymmetric (= many-to-one) alignments (IBM1-IBM5 [6], HMMs [36])
  - ▶ train: estimate  $P(\mathbf{a}, \mathbf{f}|\mathbf{e})$  (EM like)
  - ▶ align:  $\mathbf{a}^* = \operatorname{argmax} P(\mathbf{a}|\mathbf{f}, \mathbf{e}) = \operatorname{argmax} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$
  - ▶ translate:  
$$\mathbf{e}^* = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{f}|\mathbf{e})P(\mathbf{e}) = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{e}) \operatorname{argmax}_{\mathbf{a}} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$$
- ▶ public domain implementations (Giza++ [28] ; MTTK [13])
- ▶ discriminative training (and many more features) helps a bit [24, 1, 4]
- ▶ but supervision data is scarce and unreliable

for asymmetric models, an almost solved issue ?

## Training 1.a: build word alignments

- ▶ asymmetric (= many-to-one) alignments (IBM1-IBM5 [6], HMMs [36])
  - ▶ train: estimate  $P(\mathbf{a}, \mathbf{f}|\mathbf{e})$  (EM like)
  - ▶ align:  $\mathbf{a}^* = \operatorname{argmax} P(\mathbf{a}|\mathbf{f}, \mathbf{e}) = \operatorname{argmax} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$
  - ▶ translate:  
$$\mathbf{e}^* = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{f}|\mathbf{e})P(\mathbf{e}) = \operatorname{argmax}_{\mathbf{e}} P(\mathbf{e}) \operatorname{argmax}_{\mathbf{a}} P(\mathbf{a}, \mathbf{f}|\mathbf{e})$$
- ▶ public domain implementations (Giza++ [28] ; MTTK [13])
- ▶ discriminative training (and many more features) helps a bit [24, 1, 4]
- ▶ but supervision data is scarce and unreliable

for asymmetric models, an almost solved issue ?

# Training 1.b : accumulate “phrases” and their statistics

**f**= michael geht davon aus, dass er im hause bleibt

**e**= michael assumes he that will stay in the house

(example from P. Koehn)

	michael	geht	davon	aus	,	dass	er	im	hause	bleibt
house									■	
the										
in								■		
stay										■
will										
he							■			
that					■					
assumes		■	■	■						
michael	■									

*A symmetrized alignment*

# Training 1.b : accumulate “phrases” and their statistics

f= michael geht davon aus, dass er im hause bleibt

e= michael assumes he that will stay in the house

(example from P. Koehn)

	michael	geht	davon	aus	,	dass	er	im	hause	bleibt
house									■	
the										
in								■		
stay										■
will										
he							■			
that					■					
assumes		■	■	■						
michael	■									

$$N(\textit{michael}, \textit{michael})_{++}$$

# Training 1.b : accumulate “phrases” and their statistics

**f**= michael geht davon aus, dass er im hause bleibt

**e**= michael assumes he that will stay in the house

(example from P. Koehn)

	michael	geht	davon	aus	,	dass	er	im	hause	bleibt
house									■	
the										
in								■		
stay										■
will										
he							■			
that						■				
assumes	■	■	■	■						
michael	■	■	■	■						

$N(\text{michael assumes} ; \text{michael geht davon aus})++$

# Training 1.b : accumulate “phrases” and their statistics

**f**= michael geht davon aus, dass er im hause bleibt

**e**= michael assumes he that will stay in the house

(example from P. Koehn)

	michael	geht	davon	aus	,	dass	er	im	hause	bleibt
house									■	
the										
in								■		
stay										■
will										
he							■			
that	■	■	■	■	■	■				
assumes	■	■	■	■	■	■				
michael	■	■	■	■	■	■				

$N(\text{michael assumes that} ; \text{michael geht davon aus, dass})_{++}$

# Training 1.b : accumulate “phrases” and their statistics

**f**= michael geht davon aus, dass er im hause bleibt

**e**= michael assumes he that will stay in the house  
(example from P. Koehn)

	michael	geht	davon	aus	,	dass	er	im	hause	bleibt
house									■	
the										
in								■		
stay							■	■	■	■
will							■	■	■	■
he							■	■	■	■
that						■				
assumes		■	■	■						
michael	■									

$$N(\textit{he will stay} ; \textit{er him hause bleibt}) += 0$$

# Training 1.b : accumulate “phrases” and their statistics

f= michael geht davon aus, dass er im hause bleibt

e= michael assumes he that will stay in the house

(example from P. Koehn)

	michael	geht	davon	aus	,	dass	er	im	hause	bleibt
house										
the										
in										
stay										
will										
he										
that										
assumes										
michael										

$N(\text{stay in the house} ; \text{im hause bleibt})_{++}$

# Training 1.b : accumulate “phrases” and their statistics

- ▶ **translation model = “phrase” table  $\{(e, f), w(e, f) = P(f|e)\}$**
- ▶ crudely heuristic and very noisy
  - ▶ forced alignment of non aligned words
  - ▶ non-internal translations
- ▶ sparsity: smoothing  $P(f|e) = \frac{N(e,f)}{N(e)}$  helps [40, 15]
- ▶ linguistics does not help [20]
- ▶ size an issue ? pruning helps runtimes [16]
- ▶ size NOT an issue ? Use gappy phrases [9]

▶ a real-world PT

The largest the phrase table, the better the translation

# Training 1.b : accumulate “phrases” and their statistics

- ▶ translation model = “phrase” table  $\{(e, f), w(e, f) = P(f|e)\}$
- ▶ **crudely heuristic and very noisy** ▶ a real-world PT
  - ▶ forced alignment of non aligned words
  - ▶ non literal translations
- ▶ sparsity: smoothing  $P(f|e) = \frac{N(e,f)}{N(e)}$  helps [40, 15]
- ▶ linguistics does not help [20]
- ▶ size an issue ? pruning helps runtimes [16]
- ▶ size NOT an issue ? Use gappy phrases [9]

The largest the phrase table, the better the translation

# Training 1.b : accumulate “phrases” and their statistics

- ▶ translation model = “phrase” table  $\{(e, f), w(e, f) = P(f|e)\}$
- ▶ crudely heuristic and very noisy
  - ▶ **forced alignment of non aligned words**
  - ▶ non literal translations
- ▶ sparsity: smoothing  $P(f|e) = \frac{N(e,f)}{N(e)}$  helps [40, 15]
- ▶ linguistics does not help [20]
- ▶ size an issue ? pruning helps runtimes [16]
- ▶ size NOT an issue ? Use gappy phrases [9]

▶ a real-world PT

The largest the phrase table, the better the translation

# Training 1.b : accumulate “phrases” and their statistics

- ▶ translation model = “phrase” table  $\{(e, f), w(e, f) = P(f|e)\}$
- ▶ crudely heuristic and very noisy
  - ▶ forced alignment of non aligned words
  - ▶ **non literal translations**
- ▶ sparsity: smoothing  $P(f|e) = \frac{N(e,f)}{N(e)}$  helps [40, 15]
- ▶ linguistics does not help [20]
- ▶ size an issue ? pruning helps runtimes [16]
- ▶ size NOT an issue ? Use gappy phrases [9]

▶ a real-world PT

The largest the phrase table, the better the translation

# Training 1.b : accumulate “phrases” and their statistics

- ▶ translation model = “phrase” table  $\{(e, f), w(e, f) = P(f|e)\}$
- ▶ crudely heuristic and very noisy ▶ a real-world PT
  - ▶ forced alignment of non aligned words
  - ▶ non literal translations
- ▶ **sparsity: smoothing**  $P(f|e) = \frac{N(e,f)}{N(e)}$  helps [40, 15]
- ▶ linguistics does not help [20]
- ▶ size an issue ? pruning helps runtimes [16]
- ▶ size NOT an issue ? Use gappy phrases [9]

The largest the phrase table, the better the translation

# Training 1.b : accumulate “phrases” and their statistics

- ▶ translation model = “phrase” table  $\{(e, f), w(e, f) = P(f|e)\}$
- ▶ crudely heuristic and very noisy ▶ a real-world PT
  - ▶ forced alignment of non aligned words
  - ▶ non literal translations
- ▶ sparsity: smoothing  $P(f|e) = \frac{N(e,f)}{N(e)}$  helps [40, 15]
- ▶ **linguistics does not help [20]**
- ▶ size an issue ? pruning helps runtimes [16]
- ▶ size NOT an issue ? Use gappy phrases [9]

The largest the phrase table, the better the translation

# Training 1.b : accumulate “phrases” and their statistics

- ▶ translation model = “phrase” table  $\{(e, f), w(e, f) = P(f|e)\}$
- ▶ crudely heuristic and very noisy ▶ a real-world PT
  - ▶ forced alignment of non aligned words
  - ▶ non literal translations
- ▶ sparsity: smoothing  $P(f|e) = \frac{N(e,f)}{N(e)}$  helps [40, 15]
- ▶ linguistics does not help [20]
- ▶ **size an issue ? pruning helps runtimes [16]**
- ▶ size NOT an issue ? Use gappy phrases [9]

The largest the phrase table, the better the translation

# Training 1.b : accumulate “phrases” and their statistics

- ▶ translation model = “phrase” table  $\{(e, f), w(e, f) = P(f|e)\}$
- ▶ crudely heuristic and very noisy ▶ a real-world PT
  - ▶ forced alignment of non aligned words
  - ▶ non literal translations
- ▶ sparsity: smoothing  $P(f|e) = \frac{N(e,f)}{N(e)}$  helps [40, 15]
- ▶ linguistics does not help [20]
- ▶ size an issue ? pruning helps runtimes [16]
- ▶ **size NOT an issue ? Use gappy phrases [9]**

The largest the phrase table, the better the translation

# Training 1.b : accumulate “phrases” and their statistics

- ▶ translation model = “phrase” table  $\{(e, f), w(e, f) = P(f|e)\}$
- ▶ crudely heuristic and very noisy ▶ a real-world PT
  - ▶ forced alignment of non aligned words
  - ▶ non literal translations
- ▶ sparsity: smoothing  $P(f|e) = \frac{N(e,f)}{N(e)}$  helps [40, 15]
- ▶ linguistics does not help [20]
- ▶ size an issue ? pruning helps runtimes [16]
- ▶ size NOT an issue ? Use gappy phrases [9]

The largest the phrase table, the better the translation

# Training 1.b : accumulate “phrases” and their statistics

- ▶ translation model = “phrase” table  $\{(e, f), w(e, f) = P(f|e)\}$
- ▶ crudely heuristic and very noisy ▶ a real-world PT
  - ▶ forced alignment of non aligned words
  - ▶ non literal translations
- ▶ sparsity: smoothing  $P(f|e) = \frac{N(e,f)}{N(e)}$  helps [40, 15]
- ▶ linguistics does not help [20]
- ▶ size an issue ? pruning helps runtimes [16]
- ▶ size NOT an issue ? Use gappy phrases [9]

The largest the phrase table, the better the translation

# Training 2: learn a target language model

The same old story

- ▶ ***n*-gram language models**
- ▶ large span ( $\geq 5$ -gram) models help
- ▶ more training data helps...
- ▶ ... much more than smart smoothing
- ▶ ... that can't be computed anyway

© 2016 MIT 6.860

scaling up [10, 33] more important than modeling ?

# Training 2: learn a target language model

The same old story

- ▶  $n$ -gram language models
- ▶ large span ( $\geq 5$ -gram) models help
- ▶ more training data helps...
- ▶ ... much more than smart smoothing
- ▶ ... that can't be computed anyway

© 2016 MIT 6.034

scaling up [10, 33] more important than modeling ?

# Training 2: learn a target language model

The same old story

- ▶  $n$ -gram language models
- ▶ large span ( $\geq 5$ -gram) models help
- ▶ **more training data helps...**
- ▶ ... much more than smart smoothing
- ▶ ... that can't be computed anyway

▶ Results from [5]

scaling up [10, 33] more important than modeling ?

# Training 2: learn a target language model

The same old story

- ▶  $n$ -gram language models
- ▶ large span ( $\geq 5$ -gram) models help
- ▶ more training data helps...
- ▶ ... much more than smart smoothing
- ▶ ... that can't be computed anyway

▶ Results from [5]

scaling up [10, 33] more important than modeling ?

# Training 2: learn a target language model

The same old story

- ▶  $n$ -gram language models
- ▶ large span ( $\geq 5$ -gram) models help
- ▶ more training data helps...
- ▶ ... much more than smart smoothing
- ▶ ... that can't be computed anyway

▶ Results from [5]

scaling up [10, 33] more important than modeling ?

# Training 2: learn a target language model

The same old story

- ▶  $n$ -gram language models
- ▶ large span ( $\geq 5$ -gram) models help
- ▶ more training data helps...
- ▶ ... much more than smart smoothing
- ▶ ... that can't be computed anyway

▶ Results from [5]

scaling up [10, 33] more important than modeling ?

# Training 2: learn a target language model

The same old story

- ▶  $n$ -gram language models
- ▶ large span ( $\geq 5$ -gram) models help
- ▶ more training data helps...
- ▶ ... much more than smart smoothing
- ▶ ... that can't be computed anyway

▶ Results from [5]

scaling up [10, 33] more important than modeling ?

## Training 3: tune the score function

Translation score

$$s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

where  $F_k(\mathbf{e}, \mathbf{f})$  corresponds to:

translation models, language model, distortion models,  
length model, segmentation model, etc

→ use held-out data  $D$  to optimize weights  $\{\lambda_k, k = 1 \dots K\}$

$$\lambda^* = \underset{\lambda}{\operatorname{argmin}} \operatorname{LOSS}(D, \lambda) \quad [27]$$

→  $\operatorname{LOSS}()$  typically not differentiable in  $\lambda$

doing in right makes a difference [8, 25]

## Training 3: tune the score function

Translation score

$$s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

where  $F_k(\mathbf{e}, \mathbf{f})$  corresponds to:

translation models, language model, distortion models,  
length model, segmentation model, etc

- ▶ use held-out data  $D$  to optimize weights  $\{\lambda_k, k = 1 \dots K\}$

$$\lambda^* = \underset{\lambda}{\operatorname{argmin}} \operatorname{LOSS}(D, \lambda) \text{ [27]}$$

- ▶  $\operatorname{LOSS}()$  typically not differentiable in  $\lambda$

doing it right makes a difference [8, 25]

## Training 3: tune the score function

Translation score

$$s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

where  $F_k(\mathbf{e}, \mathbf{f})$  corresponds to:

translation models, language model, distortion models,  
length model, segmentation model, etc

- ▶ use held-out data  $D$  to optimize weights  $\{\lambda_k, k = 1 \dots K\}$

$$\lambda^* = \underset{\lambda}{\operatorname{argmin}} \operatorname{LOSS}(D, \lambda) \text{ [27]}$$

- ▶ **LOSS() typically not differentiable in  $\lambda$**

doing in right makes a difference [8, 25]

## Training 3: tune the score function

### Translation score

$$s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

where  $F_k(\mathbf{e}, \mathbf{f})$  corresponds to:

translation models, language model, distortion models,  
length model, segmentation model, etc

- ▶ use held-out data  $D$  to optimize weights  $\{\lambda_k, k = 1 \dots K\}$

$$\lambda^* = \underset{\lambda}{\operatorname{argmin}} \operatorname{LOSS}(D, \lambda) \quad [27]$$

- ▶  $\operatorname{LOSS}()$  typically not differentiable in  $\lambda$

doing in right makes a difference [8, 25]

## Training 3: tune the score function

### Translation score

$$s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

where  $F_k(\mathbf{e}, \mathbf{f})$  corresponds to:

translation models, language model, distortion models,  
length model, segmentation model, etc

- ▶ use held-out data  $D$  to optimize weights  $\{\lambda_k, k = 1 \dots K\}$

$$\lambda^* = \underset{\lambda}{\operatorname{argmin}} \operatorname{LOSS}(D, \lambda) \text{ [27]}$$

- ▶  $\operatorname{LOSS}()$  typically not differentiable in  $\lambda$

doing it right makes a difference [8, 25]

# Decoding, an optimisation problem

Solve  $\operatorname{argmax}_{\mathbf{e}} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$

- ▶ very large hypothesis space ( $\Rightarrow$  a NP-hard problem [17])
  - ▶ all segmentations of source sentence
  - ▶ all translations of each source phrase
  - ▶ *every permutation of the source phrases*
- ▶ heuristic search + fine-tuned pruning
- ▶ high performance, fast decoding doable

Not so much an issue ... for laboratory systems

# Decoding, an optimisation problem

Solve  $\operatorname{argmax}_{\mathbf{e}} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$

- ▶ very large hypothesis space ( $\Rightarrow$  a NP-hard problem [17])
  - ▶ **all segmentations of source sentence**
  - ▶ all translations of each source phrase
  - ▶ *every permutation of the source phrases*
- ▶ heuristic search + fine-tuned pruning
- ▶ high performance, fast decoding doable

Not so much an issue ... for laboratory systems

# Decoding, an optimisation problem

Solve  $\operatorname{argmax}_{\mathbf{e}} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$

- ▶ very large hypothesis space ( $\Rightarrow$  a NP-hard problem [17])
  - ▶ all segmentations of source sentence
  - ▶ **all translations of each source phrase**
  - ▶ *every permutation of the source phrases*
- ▶ heuristic search + fine-tuned pruning
- ▶ high performance, fast decoding doable

Not so much an issue ... for laboratory systems

# Decoding, an optimisation problem

$$\text{Solve } \operatorname{argmax}_{\mathbf{e}} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

- ▶ very large hypothesis space ( $\Rightarrow$  a NP-hard problem [17])
  - ▶ all segmentations of source sentence
  - ▶ all translations of each source phrase
  - ▶ *every permutation of the source phrases*
- ▶ heuristic search + fine-tuned pruning
- ▶ high performance, fast decoding doable

▶ Monotonic search

Not so much an issue ... for laboratory systems

# Decoding, an optimisation problem

$$\text{Solve } \operatorname{argmax}_{\mathbf{e}} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

- ▶ very large hypothesis space ( $\Rightarrow$  a NP-hard problem [17])
  - ▶ all segmentations of source sentence
  - ▶ all translations of each source phrase
  - ▶ *every permutation of the source phrases*
- ▶ **heuristic search + fine-tuned pruning**
- ▶ high performance, fast decoding doable

▶ Monotonic search

Not so much an issue ... for laboratory systems

# Decoding, an optimisation problem

$$\text{Solve } \operatorname{argmax}_{\mathbf{e}} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

- ▶ very large hypothesis space ( $\Rightarrow$  a NP-hard problem [17])
  - ▶ all segmentations of source sentence
  - ▶ all translations of each source phrase
  - ▶ *every permutation of the source phrases*
- ▶ heuristic search + fine-tuned pruning
- ▶ **high performance, fast decoding doable**

▶ Monotonic search

Not so much an issue ... for laboratory systems

# Decoding, an optimisation problem

$$\text{Solve } \operatorname{argmax}_{\mathbf{e}} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

- ▶ very large hypothesis space ( $\Rightarrow$  a NP-hard problem [17])
  - ▶ all segmentations of source sentence
  - ▶ all translations of each source phrase
  - ▶ *every permutation of the source phrases*
- ▶ heuristic search + fine-tuned pruning
- ▶ high performance, fast decoding doable

▶ Monotonic search

Not so much an issue ... for laboratory systems

# Decoding, an optimisation problem

$$\text{Solve } \operatorname{argmax}_{\mathbf{e}} s(\mathbf{e}, \mathbf{f}) = \sum_{k=1}^K \lambda_k F_k(\mathbf{e}, \mathbf{f})$$

- ▶ very large hypothesis space ( $\Rightarrow$  a NP-hard problem [17])
  - ▶ all segmentations of source sentence
  - ▶ all translations of each source phrase
  - ▶ *every permutation of the source phrases*
- ▶ heuristic search + fine-tuned pruning
- ▶ high performance, fast decoding doable

▶ Monotonic search

Not so much an issue ... for laboratory systems

# Get some numbers

## Evaluating machine translation

- ▶ **subjective evaluation is very costly**
- ▶ objective evaluation is challenging
- ▶ a fragile consensus: **BLEU** [29]

BLEU measures the surface similarity with reference translations  
→ not the generative power of the system's outputs

# Get some numbers

## Evaluating machine translation

- ▶ subjective evaluation is very costly
- ▶ **objective evaluation is challenging**
- ▶ a fragile consensus: BLEU [29]
  - ▶ measures the surface similarity with reference translation(s)
  - ▶ based on the geometric mean of the n-gram precisions

# Get some numbers

## Evaluating machine translation

- ▶ subjective evaluation is very costly
- ▶ objective evaluation is challenging
- ▶ a fragile consensus: BLEU [29]
  - ▶ measures the surface similarity with reference translation(s)
  - ▶ as the geometric mean of the  $n$ -gram precision

# Get some numbers

## Evaluating machine translation

- ▶ subjective evaluation is very costly
- ▶ objective evaluation is challenging
- ▶ a fragile consensus: BLEU [29]
  - ▶ measures the surface similarity with reference translation(s)
  - ▶ as the geometric mean of the  $n$ -gram precision

# Get some numbers

## Evaluating machine translation

- ▶ subjective evaluation is very costly
- ▶ objective evaluation is challenging
- ▶ a fragile consensus: BLEU [29]
  - ▶ measures the surface similarity with reference translation(s)
  - ▶ as the geometric mean of the  $n$ -gram precision

# Get some numbers

## Evaluating machine translation

- ▶ subjective evaluation is very costly
- ▶ objective evaluation is challenging
- ▶ a fragile consensus: BLEU [29]
  - ▶ measures the surface similarity with reference translation(s)
  - ▶ as the geometric mean of the  $n$ -gram precision

I am feeling good

Ref1: I am happy

Ref2: I am feeling very good

# Get some numbers

## Evaluating machine translation

- ▶ subjective evaluation is very costly
- ▶ objective evaluation is challenging
- ▶ a fragile consensus: BLEU [29]
  - ▶ measures the surface similarity with reference translation(s)
  - ▶ as the geometric mean of the  $n$ -gram precision

I am feeling good

Ref1: I am happy

Ref2: I am feeling very good

$$p_1 = 1$$

# Get some numbers

## Evaluating machine translation

- ▶ subjective evaluation is very costly
- ▶ objective evaluation is challenging
- ▶ a fragile consensus: BLEU [29]
  - ▶ measures the surface similarity with reference translation(s)
  - ▶ as the geometric mean of the  $n$ -gram precision

I am feeling good

Ref1: I am happy

Ref2: I am feeling very good

$$p_1 = 1 \quad p_2 = \frac{2}{3}$$

# Get some numbers

## Evaluating machine translation

- ▶ subjective evaluation is very costly
- ▶ objective evaluation is challenging
- ▶ a fragile consensus: BLEU [29]
  - ▶ measures the surface similarity with reference translation(s)
  - ▶ as the geometric mean of the  $n$ -gram precision

I am feeling good

Ref1: I am happy

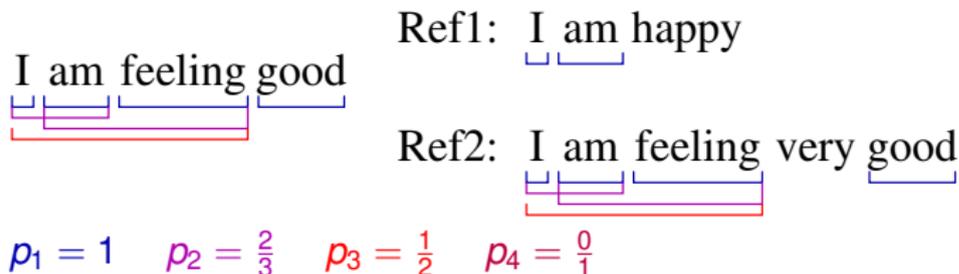
Ref2: I am feeling very good

$$p_1 = 1 \quad p_2 = \frac{2}{3} \quad p_3 = \frac{1}{2} \quad p_4 = \frac{0}{1}$$

# Get some numbers

## Evaluating machine translation

- ▶ subjective evaluation is very costly
- ▶ objective evaluation is challenging
- ▶ a fragile consensus: BLEU [29]
  - ▶ measures the surface similarity with reference translation(s)
  - ▶ as the geometric mean of the  $n$ -gram precision



an active research topic, many proposals are on the table

# A step back: finite-state SMT

- ▶ **phrase-table lookup  $[pt]$  is finite-state**

▶ a simple phrase table

- ▶  $n$ -gram models  $lm$  can be implemented as weighted fSA

- ▶ monotonic decode of  $\mathbf{f}$ :

$$\mathbf{e}^* = \mathit{bestpath}(\pi_2(\mathbf{f} \circ pt) \circ lm) [7]$$

- ▶ decode with reordering

$$\mathbf{e}^* = \mathit{bestpath}(\pi_2(\mathbf{perm}(\mathbf{f}) \circ pt) \circ lm) [3]$$

# A step back: finite-state SMT

- ▶ phrase-table lookup  $[pt]$  is finite-state

▶ a simple phrase table

- ▶ *n*-gram models  $lm$  can be implemented as weighted fSA

- ▶ monotonic decode of  $f$ :

$$e^* = \text{bestpath}(\pi_2(f \circ pt) \circ lm) [7]$$

- ▶ decode with reordering

$$e^* = \text{bestpath}(\pi_2(\text{perm}(f) \circ pt) \circ lm) [3]$$

# A step back: finite-state SMT

- ▶ phrase-table lookup [ $pt$ ] is finite-state
- ▶  $n$ -gram models  $lm$  can be implemented as weighted fSA

▶ a simple phrase table

- ▶ **monotonic decode of  $\mathbf{f}$ :**

$$\mathbf{e}^* = \mathit{bestpath}(\pi_2(\mathbf{f} \circ pt) \circ lm) [7]$$

- ▶ decode with reordering

$$\mathbf{e}^* = \mathit{bestpath}(\pi_2(\mathbf{perm}(\mathbf{f}) \circ pt) \circ lm) [3]$$

# A step back: finite-state SMT

- ▶ phrase-table lookup  $[pt]$  is finite-state ▶ a simple phrase table
- ▶  $n$ -gram models  $lm$  can be implemented as weighted fSA
- ▶ monotonic decode of  $\mathbf{f}$ :  
 $\mathbf{e}^* = \text{bestpath}(\pi_2(\mathbf{f} \circ pt) \circ lm)$  [7]
- ▶ decode with reordering  
 $\mathbf{e}^* = \text{bestpath}(\pi_2(\mathbf{perm}(\mathbf{f}) \circ pt) \circ lm)$  [3]

# A step back: finite-state SMT

- ▶ phrase-table lookup  $[pt]$  is finite-state ▶ a simple phrase table
- ▶  $n$ -gram models  $lm$  can be implemented as weighted fSA
- ▶ monotonic decode of  $\mathbf{f}$ :  
 $\mathbf{e}^* = \text{bestpath}(\pi_2(\mathbf{f} \circ pt) \circ lm)$  [7]
- ▶ decode with reordering  
 $\mathbf{e}^* = \text{bestpath}(\pi_2(\mathbf{perm}(\mathbf{f}) \circ pt) \circ lm)$  [3]

efficient implementations, scalability, training procedures,  
non-deterministic input-outputs, integration of various  
knowledge-sources [18, 22]

# A step back: finite-state SMT

- ▶ phrase-table lookup  $[pt]$  is finite-state ▶ a simple phrase table
- ▶  $n$ -gram models  $lm$  can be implemented as weighted fSA
- ▶ monotonic decode of  $\mathbf{f}$ :  
 $\mathbf{e}^* = \text{bestpath}(\pi_2(\mathbf{f} \circ pt) \circ lm)$  [7]
- ▶ decode with reordering  
 $\mathbf{e}^* = \text{bestpath}(\pi_2(\mathbf{perm}(\mathbf{f}) \circ pt) \circ lm)$  [3]

How to model  $perm(\mathbf{f})$  ?

# Approaches to reordering

Some attempts at modeling  $perm(\mathbf{f})$

- ▶ **brute-force approach**

▶ try all permutations

+ pruning based on distortion weights

- ▶ a priori defined permutations

▶  $perm(\mathbf{f}, perm) = \mathbf{f} \circ perm$

▶  $perm(\mathbf{f}, perm) = \{T, S^2, (TS)^2\}$

- ▶ empirically defined permutations

▶  $perm(\mathbf{f}, perm) = \mathbf{f} \circ T$

▶  $perm(\mathbf{f}, perm) = \{T, S^2, (TS)^2\}$

- ▶ hand-crafted reordering rules

▶  $perm(\mathbf{f}, perm) = \mathbf{f}$

- ▶ any combination thereof

Small to mild gains with respect to monotonic translation; huge gap in performance between “easy” and “difficult” language pairs.

# Approaches to reordering

Some attempts at modeling  $perm(\mathbf{f})$

- ▶ **brute-force approach**
  - + **pruning based on distortion weights**
- ▶ a priori defined permutations
  - define  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - define  $G$ ,  $perm(\mathbf{f}) = \{T, S^k \circ T\}$
- ▶ empirically defined permutations
  - learn  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - learn  $G$ ,  $perm(\mathbf{f}) = \{T, S^k \circ T\}$
- ▶ hand-crafted reordering rules
- ▶ any combination thereof

▶ try all permutations

▶ learn permutations

Small to mild gains with respect to monotonic translation; huge gap in performance between “easy” and “difficult” language pairs.

# Approaches to reordering

Some attempts at modeling  $perm(\mathbf{f})$

- ▶ brute-force approach  
+ pruning based on distortion weights

▶ try all permutations

- ▶ a priori defined permutations

- ▶ define  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$

▶ finite-state models

- ▶ define  $G$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \xrightarrow{*} (f, f')\}$

▶ context-free models

- ▶ empirically defined permutations

- ▶ learn  $T$  from  $(\mathbf{f}, \mathbf{f}')$

- ▶ learn  $G$  from  $(\mathbf{f}, S \xrightarrow{*} (f, f'))$

- ▶ hand-crafted reordering rules

▶ simple heuristics

- ▶ any combination thereof

Small to mild gains with respect to monotonic translation; huge gap in performance between “easy” and “difficult” language pairs.

# Approaches to reordering

Some attempts at modeling  $perm(\mathbf{f})$

- ▶ brute-force approach
  - + pruning based on distortion weights
- ▶ a priori defined permutations
  - ▶ define  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ define  $G$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \xrightarrow{*} (f; f')\}$
- ▶ empirically defined permutations
- ▶ hand-crafted reordering rules
- ▶ any combination thereof

▶ try all permutations

▶ finite-state models

▶ context-free models

▶ simple reordering

Small to mild gains with respect to monotonic translation; huge gap in performance between “easy” and “difficult” language pairs.

# Approaches to reordering

Some attempts at modeling  $perm(\mathbf{f})$

- ▶ brute-force approach
  - + pruning based on distortion weights
- ▶ a priori defined permutations
  - ▶ define  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ define  $G$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{*}{\Rightarrow} (f; f')\}$
- ▶ empirically defined permutations
  - + learn/train  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - + learn/train  $G$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{*}{\Rightarrow} (f; f')\}$
- ▶ hand-crafted reordering rules
- ▶ any combination thereof

▶ try all permutations

▶ finite-state models

▶ context-free models

▶ simple heuristics

Small to mild gains with respect to monotonic translation; huge gap in performance between “easy” and “difficult” language pairs.

# Approaches to reordering

Some attempts at modeling  $perm(\mathbf{f})$

- ▶ brute-force approach
  - + pruning based on distortion weights
- ▶ a priori defined permutations
  - ▶ define  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ define  $G$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{*}{\Rightarrow} (f; f')\}$
- ▶ empirically defined permutations
  - ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{*}{\Rightarrow} (f; f')\}$
- ▶ hand-crafted reordering rules
- ▶ any combination thereof

▶ try all permutations

▶ finite-state models

▶ context-free models

▶ finite-state models

▶ context-free models

▶ finite-state models

Small to mild gains with respect to monotonic translation; huge gap in performance between “easy” and “difficult” language pairs.

# Approaches to reordering

Some attempts at modeling  $perm(\mathbf{f})$

- ▶ brute-force approach
  - + pruning based on distortion weights

▶ try all permutations

- ▶ a priori defined permutations

- ▶ define  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$

▶ finite-state models

- ▶ define  $G$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{*}{\Rightarrow} (f; f')\}$

▶ context-free models

- ▶ empirically defined permutations

- ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$

▶ finite-state models

- ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{*}{\Rightarrow} (f; f')\}$

▶ context-free models

- ▶ hand-crafted reordering rules

▶ a man-made model

- ▶ any combination thereof

Small to mild gains with respect to monotonic translation; huge gap in performance between “easy” and “difficult” language pairs.

# Approaches to reordering

Some attempts at modeling  $perm(\mathbf{f})$

- ▶ brute-force approach
  - + pruning based on distortion weights
- ▶ a priori defined permutations
  - ▶ define  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ define  $G$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{*}{\Rightarrow} (f; f')\}$
- ▶ empirically defined permutations
  - ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{*}{\Rightarrow} (f; f')\}$
- ▶ hand-crafted reordering rules
- ▶ any combination thereof

▶ try all permutations

▶ finite-state models

▶ context-free models

▶ finite-state models

▶ context-free models

▶ a man-made model

Small to mild gains with respect to monotonic translation; huge gap in performance between “easy” and “difficult” language pairs.

# Approaches to reordering

Some attempts at modeling  $perm(\mathbf{f})$

- ▶ brute-force approach
  - + pruning based on distortion weights
- ▶ a priori defined permutations
  - ▶ define  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ define  $G$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{\star}{\Rightarrow} (f; f')\}$
- ▶ empirically defined permutations
  - ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{\star}{\Rightarrow} (f; f')\}$
- ▶ **hand-crafted reordering rules**
- ▶ any combination thereof

▶ try all permutations

▶ finite-state models

▶ context-free models

▶ finite-state models

▶ context-free models

▶ a man-made model

Small to mild gains with respect to monotonic translation; huge gap in performance between “easy” and “difficult” language pairs.

# Approaches to reordering

Some attempts at modeling  $perm(\mathbf{f})$

- ▶ brute-force approach
  - + pruning based on distortion weights
- ▶ a priori defined permutations
  - ▶ define  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ define  $G$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{*}{\Rightarrow} (f; f')\}$
- ▶ empirically defined permutations
  - ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{*}{\Rightarrow} (f; f')\}$
- ▶ hand-crafted reordering rules
- ▶ any combination thereof

▶ try all permutations

▶ finite-state models

▶ context-free models

▶ finite-state models

▶ context-free models

▶ a man-made model

Small to mild gains with respect to monotonic translation; huge gap in performance between “easy” and “difficult” language pairs.

# Approaches to reordering

Some attempts at modeling  $perm(\mathbf{f})$

- ▶ brute-force approach
  - + pruning based on distortion weights
- ▶ a priori defined permutations
  - ▶ define  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ define  $G$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{\star}{\Rightarrow} (f; f')\}$
- ▶ empirically defined permutations
  - ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{\star}{\Rightarrow} (f; f')\}$
- ▶ hand-crafted reordering rules
- ▶ any combination thereof

▶ try all permutations

▶ finite-state models

▶ context-free models

▶ finite-state models

▶ context-free models

▶ a man-made model

Small to mild gains with respect to monotonic translation; huge gap in performance between “easy” and “difficult” language pairs.

# Approaches to reordering

Some attempts at modeling  $perm(\mathbf{f})$

- ▶ brute-force approach
  - + pruning based on distortion weights
- ▶ a priori defined permutations
  - ▶ define  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ define  $G$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{\star}{\Rightarrow} (f; f')\}$
- ▶ empirically defined permutations
  - ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \mathbf{f} \circ T$
  - ▶ learn/train  $T$ ,  $perm(\mathbf{f}) = \{\mathbf{f}', S \stackrel{\star}{\Rightarrow} (f; f')\}$
- ▶ hand-crafted reordering rules
- ▶ any combination thereof

▶ try all permutations

▶ finite-state models

▶ context-free models

▶ finite-state models

▶ context-free models

▶ a man-made model

Small to mild gains with respect to monotonic translation; huge gap in performance between “easy” and “difficult” language pairs.

# Why it works

PBT better than word based models

- ▶ **idioms, terms, multi-word units**

*pulling my leg, mène en bateau*

- ▶ “local” reordering decisions

*pulling my leg, mène en bateau* → *pulling my leg, mène en bateau*

- ▶ model “local” context and agreement

*pulling my leg, mène en bateau* → *pulling my leg, mène en bateau*

- ▶ allies simplicity, speed, and robustness

- ▶ matching large phrases yield high BLEU scores

# Why it works

PBT better than word based models

- ▶ idioms, terms, multi-word units

*pulling my leg, mène en bateau*

- ▶ “local” reordering decisions

*international conference, conférence internationale*

- ▶ model “local” context and agreement

- ▶ allies simplicity, speed, and robustness

- ▶ matching large phrases yield high BLEU scores

# Why it works

PBT better than word based models

- ▶ idioms, terms, multi-word units

*pulling my leg, mène en bateau*

- ▶ “local” reordering decisions

*international conference, conférence internationale*

- ▶ model “local” context and agreement

- ▶ allies simplicity, speed, and robustness

- ▶ matching large phrases yield high BLEU scores

# Why it works

PBT better than word based models

- ▶ idioms, terms, multi-word units

*pulling my leg, mène en bateau*

- ▶ “local” reordering decisions

*international conference, conférence internationale*

- ▶ model “local” context and agreement

*the international conference, la conférence internationale*

- ▶ allies simplicity, speed, and robustness

- ▶ matching large phrases yield high BLEU scores

# Why it works

PBT better than word based models

- ▶ idioms, terms, multi-word units

*pulling my leg, mène en bateau*

- ▶ “local” reordering decisions

*international conference, conférence internationale*

- ▶ **model “local” context and agreement**

*the international conference, la conférence internationale*

- ▶ allies simplicity, speed, and robustness

- ▶ matching large phrases yield high BLEU scores

# Why it works

PBT better than word based models

- ▶ idioms, terms, multi-word units  
*pulling my leg, mène en bateau*
- ▶ “local” reordering decisions  
*international conference, conférence internationale*
- ▶ model “local” context and agreement  
*the international conference, la conférence internationale*
- ▶ allies simplicity, speed, and robustness
- ▶ matching large phrases yield high BLEU scores

# Why it works

PBT better than word based models

- ▶ idioms, terms, multi-word units  
*pulling my leg, mène en bateau*
- ▶ “local” reordering decisions  
*international conference, conférence internationale*
- ▶ model “local” context and agreement  
*the international conference, la conférence internationale*
- ▶ **allies simplicity, speed, and robustness**
- ▶ matching large phrases yield high BLEU scores

# Why it works

PBT better than word based models

- ▶ idioms, terms, multi-word units  
*pulling my leg, mène en bateau*
- ▶ “local” reordering decisions  
*international conference, conférence internationale*
- ▶ model “local” context and agreement  
*the international conference, la conférence internationale*
- ▶ allies simplicity, speed, and robustness
- ▶ matching large phrases yield high BLEU scores

# Why it fails

PBT worst than syntax-based models ?

- ▶ **purely surfacist (no morphology, see [19] for a cure)**
- ▶ contiguous phrases miss important generalizations
- ▶ only “local” syntax on the target side ( $n$ -gram models)
- ▶ phrase weighting and selection is context-free
- ▶ no global reordering model

# Why it fails

PBT worst than syntax-based models ?

- ▶ purely surfacist (no morphology, see [19] for a cure)
- ▶ **contiguous phrases miss important generalizations**
- ▶ only “local” syntax on the target side ( $n$ -gram models)
- ▶ phrase weighting and selection is context-free
- ▶ no global reordering model

# Why it fails

PBT worst than syntax-based models ?

- ▶ purely surfacist (no morphology, see [19] for a cure)
- ▶ contiguous phrases miss important generalizations
- ▶ only “local” syntax on the target side (*n*-gram models)
- ▶ phrase weighting and selection is context-free
- ▶ no global reordering model

# Why it fails

PBT worst than syntax-based models ?

- ▶ purely surfacist (no morphology, see [19] for a cure)
- ▶ contiguous phrases miss important generalizations
- ▶ only “local” syntax on the target side ( $n$ -gram models)
- ▶ **phrase weighting and selection is context-free**
- ▶ no global reordering model

# Why it fails

PBT worst than syntax-based models ?

- ▶ purely surfacist (no morphology, see [19] for a cure)
- ▶ contiguous phrases miss important generalizations
- ▶ only “local” syntax on the target side ( $n$ -gram models)
- ▶ phrase weighting and selection is context-free
- ▶ **no global reordering model**

# Temporary conclusions

- ▶ **SMT's recent progress = simpler models + larger databases + metrics**
- ▶ + tuning + paying attentions to details
- ▶ acceptable translations for many pairs ▶ translations
- ▶ issue: modeling word order ... with acceptable robustness and speed  
⇒ towards more linguistically informed systems ?

# Temporary conclusions

- ▶ SMT's recent progress = simpler models + larger databases + metrics
- ▶ + tuning + paying attentions to details
- ▶ acceptable translations for many pairs translations
- ▶ issue: modeling word order ... with acceptable robustness and speed  
⇒ towards more linguistically informed systems ?

# Temporary conclusions

- ▶ SMT's recent progress = simpler models + larger databases + metrics
- ▶ + tuning + paying attentions to details
- ▶ **acceptable translations for many pairs** ▶ translations
- ▶ issue: modeling word order ... with acceptable robustness and speed  
⇒ towards more linguistically informed systems ?

# Temporary conclusions

- ▶ SMT's recent progress = simpler models + larger databases + metrics
- ▶ + tuning + paying attentions to details
- ▶ acceptable translations for many pairs
- ▶ **issue: modeling word order ... with acceptable robustness and speed**  
⇒ **towards more linguistically informed systems ?**

▶ translations

Questions ?

[We have opened post-doctoral positions]

## Exhaustive search

- ▶  $\mathbf{f}$  has a finite number of permutations
- ▶ hence represented by a finite-state automaton
- ▶ yet can't compute  $perm(\mathbf{f})$  with a finite-state device



# Heuristic search

- ▶ moves allowed within fixed boundaries
- ▶ small moves preferred over longer moves
- ▶ standard model:
  - ▶ distortion:  $d(i) = f(\text{start}(f_i) - \text{end}(f_{i-1}) - 1)$
  - ▶  $P(d(i) = k) \propto \exp(-\alpha k)$
  - ▶  $\forall i, d(i) < d_{max}$
- ▶ (costly) extension: lexicalized reordering weights [34]

▶ back



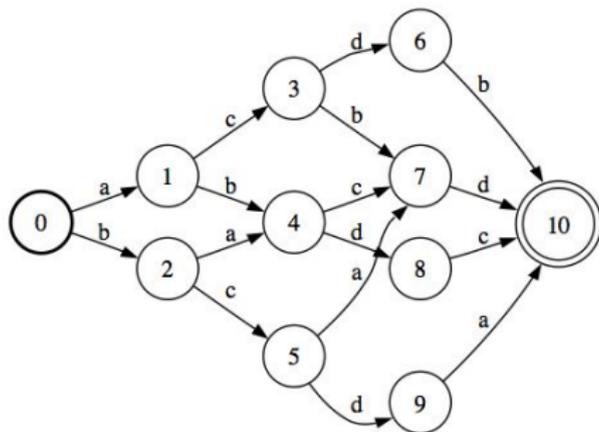
# IBM style constraints

- ▶ choose one the first  $k$  remaining tokens
- ▶ additional constraints:
  - ▶ moves take place within a fixed size window;
  - ▶ restrict the number of simultaneous gaps;

▶ back

# IBM style constraints

- ▶ choose one the first  $k$  remaining tokens
- ▶ additional constraints:
  - ▶ moves take place within a fixed size window;
  - ▶ restrict the number of simultaneous gaps;



The IBM permutations of  $a b c d$  for  $k = 2$

# A local approach

see [21] for details

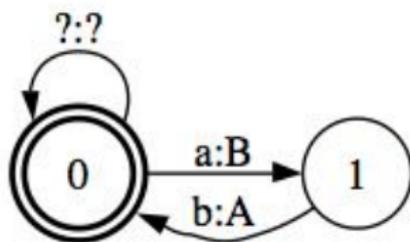
- ▶ allows permutations of neighbouring phrases
- ▶ within a bounded window

▶ back

# A local approach

see [21] for details

- ▶ allows permutations of neighbouring phrases
- ▶ within a bounded window



One state  $\forall a:A, b:B \in pt$ ,  $?:?$  is a copy loop

Exchange adjacent phrases

▶ back



# Inversion Transduction Grammars (ITGs)

A CF model for permutations

## Definition (from [37])

An Inversion Transduction Grammar (ITG) is a 5-uple  $G = (V, \Sigma, \Gamma, S, P)$ , where the context-free productions:

- ▶ terminals come in pairs  $a/b \in (\Sigma \cup \{\epsilon\}) \times (\Gamma \cup \{\epsilon\})$
- ▶ right-hand sides are explicitly oriented:
  - ▶  $A \rightarrow [BC]$ : left-to-right order in both derivations
  - ▶  $A \rightarrow \langle BC \rangle$ : left-to-right in one language, right-to-left in the other

▶ back

# ITG's permutations

## Bracketing grammar

Let  $G$  have productions  $X \rightarrow [XX] \mid \langle XX \rangle$ , and  $X \rightarrow e; e, \forall e;$

$$\text{perm}(w_1 \dots w_n) = \{v_1 \dots v_n \mid X \xRightarrow{*} w_1 \dots w_n; v_1 \dots v_n\}$$

▶ back









# ITG's permutations

## Bracketing grammar

Let  $G$  have productions  $X \rightarrow [XX] \mid \langle XX \rangle$ , and  $X \rightarrow e; e, \forall e;$

$$\text{perm}(w_1 \dots w_n) = \{v_1 \dots v_n \mid X \xrightarrow{*} w_1 \dots w_n; v_1 \dots v_n\}$$

## Complements

- ▶ a strict subset of all permutations
- ▶ combinatorily large  $O(K^n)$  [39], yet  $\ll n!$
- ▶ can be searched in polynomial time [39, 14]

▶ back

# Linguistic reordering

- ▶ use linguistically motivated transformations rules eg. [11]

## *Verb Initial Rule*

*In any verb phrase, find the head of the phrase, and move it into the initial position within the verb phrase*

**f**= *Ich werde Ihnen die entsprechenden Anmerkungen aushändigen*

**f'** = *Ich werde aushändigen ihnen die entsprechenden Anmerkungen*

**e**= *I will pass on to you the corresponding comments*

- ▶ deterministic process  $\Rightarrow$  transform dataset prior to learning
- ▶ requirements: a source parser + linguistic rules (for each pair)

▶ back

# Learning reordering rules

see eg. [38, 12]

## ▶ training procedure

- ▶ build symmetric alignments and extract phrases
- ▶ learn “within-phrase” reordering rules
- ▶ compose rules as a non-deterministic reordering transducer  $R$

$$R = \bigcirc_i (r_i \cup Id)$$

- ▶ decoding uses  $perm(\mathbf{f}) = \pi_1(tag(\mathbf{f}) \circ R)$

▶ back

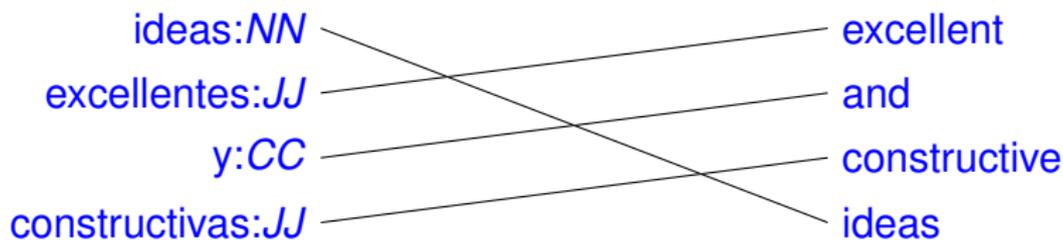
# Learning reordering rules

see eg. [38, 12]

- ▶ training procedure
  - ▶ build symmetric alignments and extract phrases
  - ▶ learn “within-phrase” reordering rules
  - ▶ compose rules as a non-deterministic reordering transducer  $R$

$$R = \bigcirc_i (r_i \cup Id)$$

- ▶ decoding uses  $perm(\mathbf{f}) = \pi_1(tag(\mathbf{f}) \circ R)$



# Learning reordering rules

see eg. [38, 12]

- ▶ training procedure
  - ▶ build symmetric alignments and extract phrases
  - ▶ learn “within-phrase” reordering rules
  - ▶ compose rules as a non-deterministic reordering transducer  $R$

$$R = \bigcirc_i (r_i \cup Id)$$

- ▶ decoding uses  $perm(\mathbf{f}) = \pi_1(tag(\mathbf{f}) \circ R)$

excelentes:JJ ————— excellent

y:CC ————— and

constructivas:JJ ————— constructive

ideas:NN ————— ideas

rule:  $NN JJ CC JJ \rightarrow JJ CC JJ NN$

# Learning reordering rules

see eg. [38, 12]

- ▶ training procedure
  - ▶ build symmetric alignments and extract phrases
  - ▶ learn “within-phrase” reordering rules
  - ▶ **compose rules as a non-deterministic reordering transducer  $R$**

$$R = \bigcirc_i (r_i \cup Id)$$

- ▶ decoding uses  $perm(\mathbf{f}) = \pi_1(\text{tag}(\mathbf{f}) \circ R)$

▶ back

# Learning reordering rules

see eg. [38, 12]

- ▶ training procedure
  - ▶ build symmetric alignments and extract phrases
  - ▶ learn “within-phrase” reordering rules
  - ▶ compose rules as a non-deterministic reordering transducer  $R$

$$R = \bigcirc_i (r_i \cup Id)$$

- ▶ decoding uses  $perm(\mathbf{f}) = \pi_1(\text{tag}(\mathbf{f}) \circ R)$

▶ back





# A hierarchical SMT system

Some innovations of [9]

- ▶ gappy phrases = rules of a synchronous CFG
  - ▶ usual phrases ( $e; f$ ) yield terminating rules  $X \rightarrow e; f$
  - ▶ gappy phrases ( $\alpha; \beta$ ) yield  $X \rightarrow \alpha; \beta$
  - ▶ “glue”  $S \rightarrow SX \mid X$
  - ▶ maximum likelihood estimates (+ smoothing)
- ▶ translation within parsing

$$\mathbf{e} = \operatorname{argmax}_{\mathbf{e} \in E} \lambda_1 \log P_{LM}(\mathbf{e}) + \lambda_2 \log P_G(\mathbf{f}; \mathbf{e}) + \dots$$

- ▶ Benefits
  - ▶ more (general) phrases
  - ▶ reordering model
  - ▶ performance [41]
- ▶ Issues
  - ▶ grammar size
  - ▶ search

# Translating with Stochastic Synchronous CFGs

- ▶ in theory:

$$\mathbf{e}^* = \operatorname{argmax}_{\mathbf{e} \in E} P_G(\mathbf{f} : \mathbf{e}) = \sum_{\{(t, t'), \text{yield}(t) = \mathbf{f} \wedge \text{yield}(t') = \mathbf{e}\}} P_G(t, t')$$

this problem is intractable (even in one dimension see [31])

- ▶ a two step approach

1. solve

$$t^* = \operatorname{argmax}_{\{t | \text{yield}(t) = \mathbf{f}\}} [P_G(t) = \sum_{t'} P_G(t, t')]$$

2. derive  $\mathbf{e}^*$  as

$$\mathbf{e}^* = \operatorname{argmax}_{\mathbf{e}} \sum_{\{t' | \text{yield}(t') = \mathbf{e}\}} P_G(t^*, t')$$

the first step is also NP-hard [30]

- ▶ a polynomial (?) approximation

$$\mathbf{e}^* = \text{yield}(t^*), \text{ where } (t, t^*) = \operatorname{argmax}_{\{t | \text{yield}(t) = \mathbf{f}\}, t'} P_G(t, t')$$

# References I

- [1] Necip Fazil Ayan and Bonnie J. Dorr. A maximum entropy approach to combining word alignments. In **Proceedings of the Human Language Technology Conference of the NAACL, Main Conference**, pages 96–103, New York City, USA, June 2006. Association for Computational Linguistics.
- [2] Srinivas Bangalore, Patrick Haffner, and Stephan Kanthak. Statistical machine translation through global lexical selection and sentence reconstruction. In **Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics**, pages 152–159, Prague, Czech Republic, 2007.
- [3] Srinivas Bangalore and Giuseppe Riccardi. Stochastic finite-state models for spoken language machine translation. **Machine Translation**, 17:165–184, 2002.
- [4] Phil Blunsom and Trevor Cohn. Discriminative word alignment with conditional random fields. In **Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics**, pages 65–72, Sydney, Australia, 2006.
- [5] Thorsten Brants, Ashok C. Popat, Peng Xu, Franz J. Och, and Jeffrey Dean. Large language models in machine translation. In **Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)**, pages 858–867, 2007.
- [6] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. **Computational Linguistics**, 19(2):263–311, 1993.
- [7] Francesco Casacuberta and Enrique Vidal. Machine translation with inferred stochastic finite-state transducers. **Computational Linguistics**, 30(3):205–225, 2004.
- [8] Daniel Cer, Dan Jurafsky, and Christopher D. Manning. Regularization and search for minimum error rate training. In **Proceedings of the Third Workshop on Statistical Machine Translation**, pages 26–34, Columbus, Ohio, 2008.
- [9] David Chiang. A hierarchical phrase-based model for statistical machine translation. In **Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)**, pages 263–270, Ann Arbor, Michigan, 2005.
- [10] Kenneth Church, Ted Hart, and Jianfeng Gao. Compressing trigram language models with Golomb coding. In **Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)**, pages 199–207, 2007.

# References II

- [11] Michael Collins, Philipp Koehn, and Ivona Kucerova. Clause restructuring for statistical machine translation. In **Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)**, pages 531–540, Ann Arbor, Michigan, 2005.
- [12] Josep Maria Crego and José B. Mari no. Improving statistical MT by coupling reordering and decoding. **Machine Translation**, 20(3):199–215, 2006.
- [13] Yonggang Deng and William Byrne. MTTK: An alignment toolkit for statistical machine translation. In **Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Demonstrations**, pages 265–268, New York City, USA, 2006.
- [14] Jason Eisner and Roy W. Tromble. Local search with very large-scale neighborhoods for optimal permutations in machine translation. In **Proceedings of the HLT-NAACL Workshop on Computationally Hard Problems and Joint Inference in Speech and Language Processing**, pages 57–75, New York, June 2006.
- [15] George Foster, Roland Kuhn, and Howard Johnson. Phrasetable smoothing for statistical machine translation. In **Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing**, pages 53–61, Sydney, Australia, 2006.
- [16] Howard Johnson, Joel Martin, George Foster, and Roland Kuhn. Improving translation quality by discarding most of the phrasetable. In **Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)**, pages 967–975, 2007.
- [17] Kevin Knight. Decoding complexity in word-replacement translation models. **Computational Linguistics**, 25(4):607–615, 1999.
- [18] Kevin Knight and Yussef Al-Onaizan. Translation with finite-state devices. In **Proceedings of the AMTA Conference**, volume 421–437, Langhorne, PA, 1998.
- [19] Philipp Koehn and Hieu Hoang. Factored translation models. In **Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)**, pages 868–876, 2007.
- [20] Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In **Proc. NAACL-HLT**, pages 127–133, Edmondton, Canada, 2003.

# References III

- [21] Shankar Kumar and William Byrne. Local phrase reordering models for statistical machine translation. In **Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing**, pages 161–168, Vancouver, British Columbia, Canada, 2005.
- [22] Shankar Kumar, Yonggang Deng, and William Byrne. A weighted finite state transducer translation template model for statistical machine translation. **Natural Language Engineering**, 12(1):35–75, 2006.
- [23] Adam Lopez. Tera-scale translation models via pattern matching. In **Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)**, pages 505–512, Manchester, UK, 2008.
- [24] Robert C. Moore. A discriminative framework for bilingual word alignment. In **Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing**, pages 81–88, Vancouver, British Columbia, Canada, 2005.
- [25] Robert C. Moore and Chris Quirk. Random restarts in minimum error rate training for statistical machine translation. In **Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)**, pages 585–592, Manchester, UK, 2008.
- [26] Dragos Stefan Munteanu and Daniel Marcu. Improving machine translation performance by exploiting non-parallel corpora. **Computational Linguistics**, 31(4):477–504, 2005.
- [27] Franz Josef Och. Minimum error rate training in statistical machine translation. In **Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics**, pages 160–167, Sapporo, Japan, 2003. Association for Computational Linguistics.
- [28] Franz-Joseph Och and Hermann Ney. Improved statistical alignment models. In **Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics**, pages 440–447, Hong Kong, 2000.
- [29] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. Technical Report RC22176 (W0109-022), IBM Research Division, Thomas J. Watson Research Center, 2001.
- [30] Giorgio Satta and Enoch Peserico. Some computational complexity results for synchronous context-free grammars. In **Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing**, pages 803–810, Vancouver, British Columbia, Canada, October 2005. Association for Computational Linguistics.

# References IV

- [31] Khalil Sima'an. Computational complexity of probabilistic disambiguation by means of tree-grammars. In **Proceedings of the 16th conference on Computational linguistics**, pages 1175–1180, Morristown, NJ, USA, 1996.
- [32] Nicolas Stroppa, Antal van den Bosch, and Andy Way. Exploiting source similarity for smt using context-informed features. In Andy Way and Barbara Gawronska, editors, **Proceedings of the 11th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI'07)**, pages 231–240, Skövde, Sweden, 2007.
- [33] David Talbot and Miles Osborne. Randomised language modelling for statistical machine translation. In **Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics**, pages 512–519, Prague, Czech Republic, June 2007. Association for Computational Linguistics.
- [34] Christoph Tillman. A unigram orientation model for statistical machine translation. In Daniel Marcu Susan Dumais and Salim Roukos, editors, **HLT-NAACL 2004: Short Papers**, pages 101–104, Boston, Massachusetts, USA, 2004.
- [35] Stephan Vogel. PESA: Phrase pair extraction as sentence splitting. In **Proceedings of the tenth Machine Translation Summit**, Phuket, Thailand, 2005.
- [36] Stephan Vogel, Hermann Ney, and Christoph Tillmann. Hmm-based word alignment in statistical translation. In **Proceedings of the 16th conference on Computational linguistics**, pages 836–841, Morristown, NJ, USA, 1996.
- [37] Dekai Wu. Stochastic inversion transduction grammar and bilingual parsing of parallel corpora. **Computational Linguistics**, 23(3):377–404, 1997.
- [38] Fei Xia and Michael McCord. Improving a statistical mt system with automatically learned rewrite patterns. In **Proceedings of Coling 2004**, pages 508–514, Geneva, Switzerland, Aug 23–Aug 27 2004. COLING.
- [39] Richard Zens and Hermann Ney. A comparative study on reordering constraints in statistical machine translation. In **Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics**, pages 144–151, Sapporo, Japan, 2003.
- [40] Richard Zens and Hermann Ney. Improvements in phrase-based statistical machine translation. In Susan Dumais, Daniel Marcu, and Salim Roukos, editors, **HLT-NAACL 2004: Main Proceedings**, pages 257–264, Boston, Massachusetts, USA, 2004.
- [41] Andreas Zollmann, Ashish Venugopal, Franz Och, and Jay Ponte. A systematic comparison of phrase-based, hierarchical and syntax-augmented statistical MT. In **Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)**, pages 1145–1152, Manchester, UK, 2008.

# This beautiful plant is unique

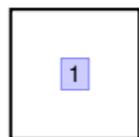
Courtesy of Ph. Langlais

▶ back

1

phrase table		
this	↔	ce
	↔	cette
beautiful	↔	belle
	↔	beau
plant	↔	plante
	↔	usine
is	↔	est
unique	↔	seule
	↔	unique
beautiful plant		
↓		
belle plante		
plante magnifique		

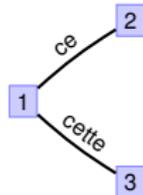
language model	
ce beau plante	:-(
cette belle usine	:-
belle usine est	:-)
...	



This beautiful plant is unique

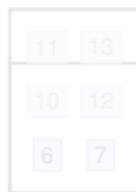
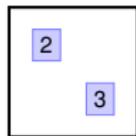
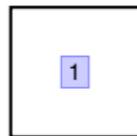
Courtesy of Ph. Langlais

▶ back



phrase table		
this	↔	ce
	↔	cette
beautiful	↔	belle
	↔	beau
plant	↔	plante
	↔	usine
is	↔	est
unique	↔	seule
	↔	unique
beautiful plant		
↓		
belle plante		
plante magnifique		

language model		
ce beau plante	:-	(
cette belle usine	:-	
belle usine est	:-	)
...		

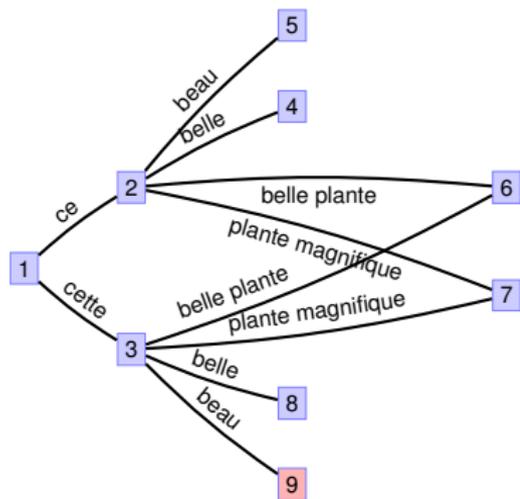




# This beautiful plant is unique

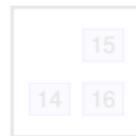
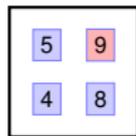
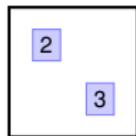
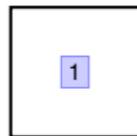
Courtesy of Ph. Langlais

▶ back



phrase table		
this	↔	ce
	↔	cette
beautiful	↔	belle
	↔	beau
plant	↔	plante
	↔	usine
is	↔	est
unique	↔	seule
	↔	unique
beautiful plant		
↓		
belle plante		
plante magnifique		

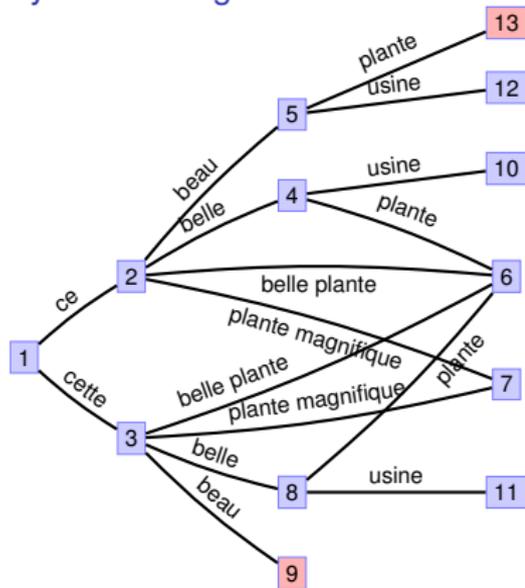
language model		
ce beau plante	:	(
cette belle usine	:	
belle usine est	:	)
...		



# This beautiful plant is unique

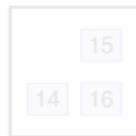
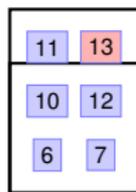
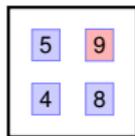
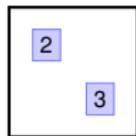
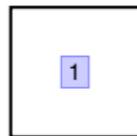
Courtesy of Ph. Langlais

▶ back



phrase table		
this	↔	ce
	↔	cette
beautiful	↔	belle
	↔	beau
plant	↔	plante
	↔	usine
is	↔	est
unique	↔	seule
	↔	unique
beautiful plant		
↓		
belle plante		
plante magnifique		

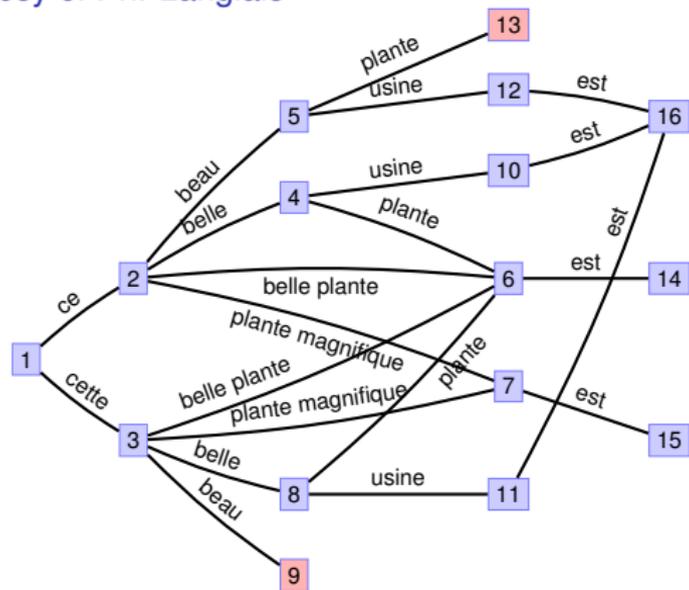
language model		
ce beau plante		:(
cette belle usine		:-
belle usine est		:-)
...		



# This beautiful plant is unique

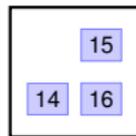
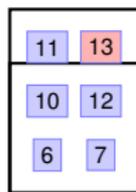
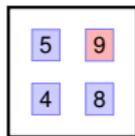
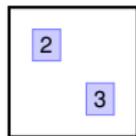
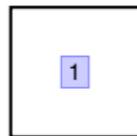
Courtesy of Ph. Langlais

▶ back



phrase table	
this	↔ ce
	↔ cette
beautiful	↔ belle
	↔ beau
plant	↔ plante
	↔ usine
is	↔ est
unique	↔ seule
	↔ unique
beautiful plant	
↓	
belle plante	
plante magnifique	

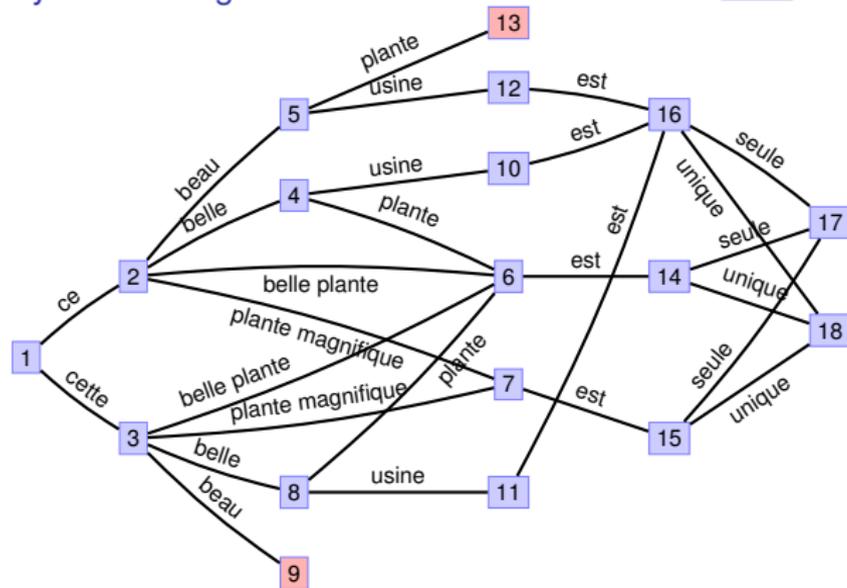
language model	
ce beau plante	:-)
cette belle usine	:-
belle usine est	:-)
...	



This beautiful plant is **unique**

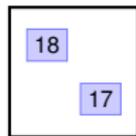
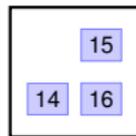
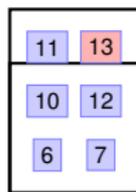
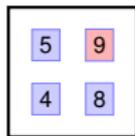
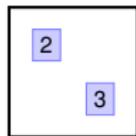
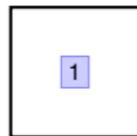
Courtesy of Ph. Langlais

▶ back



phrase table	
this	↔ ce
	↔ cette
beautiful	↔ belle
	↔ beau
plant	↔ plante
	↔ usine
is	↔ est
unique	↔ seule
	↔ unique
beautiful plant	
↓	
belle plante	
plante magnifique	

language model	
ce beau plante	:-)
cette belle usine	:-
belle usine est	:-)
...	





# A second step back

## Abstract SMT

1. get weighted local translation hypotheses from the PT
2. arrange them in a word graph
3. rescore permutations with a language model

## Two steps forward

- ▶ compute weights *on demand*, using all available information: SMT as EBMT [32], see also [35, 23]
- ▶ dispense with alignments in step 1, use complete sentence as contexts  
(but step 2 and 3 prove difficult [2])

# A second step back

## Abstract SMT

1. get weighted local translation hypotheses from the PT
2. arrange them in a word graph
3. rescore permutations with a language model

## Two steps forward

- ▶ compute weights *on demand*, using all available information: SMT as EBMT [32], see also [35, 23]
- ▶ dispense with alignments in step 1, use complete sentence as contexts  
(but step 2 and 3 prove difficult [2])

# Using Terascale Language Models

Some results from [5]

## Conventional back-off

$$P(w|h) = \begin{cases} \rho(hw) & \text{if } N(hw) > 0 \\ \alpha(h)P(w|\bar{h}) & \text{otherwise} \end{cases}$$

## "Stupid" (sic) Back-off

$$S(w|h) = \begin{cases} \frac{N(hw)}{\sum_{w'} N(hw')} & \text{if } N(hw) > 0 \\ \alpha S(W|\bar{h}) & \text{otherwise} \end{cases}$$

NB. "Stupid" Back-off does not even define a probability distribution

# Using Terascale Language Models

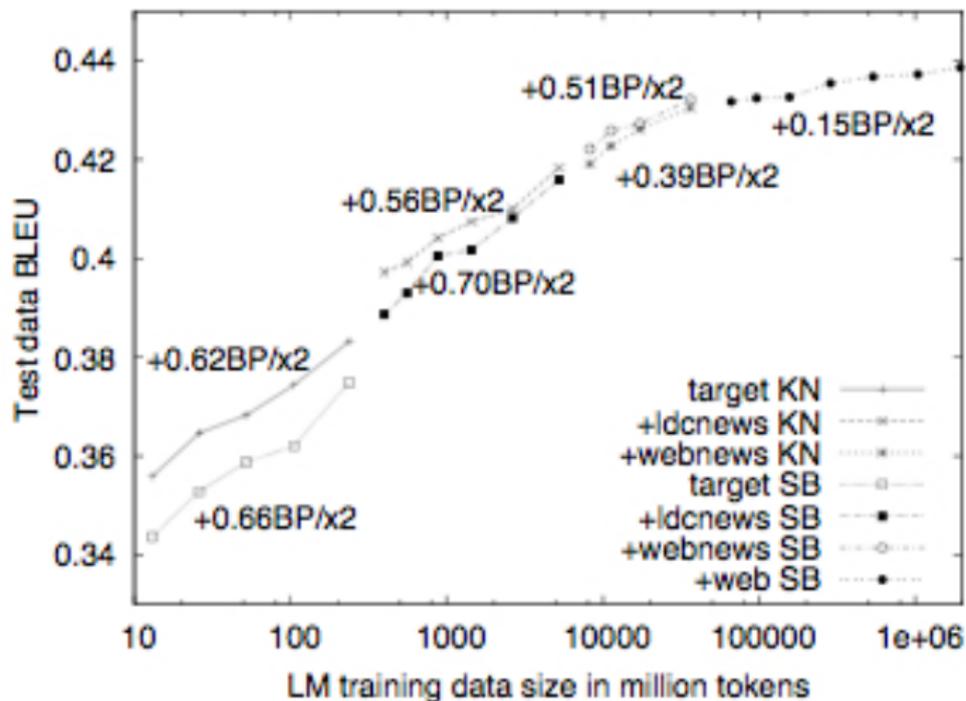
Some results from [5]

	<b>target</b>	<b>webnews</b>	<b>web</b>
# token	237 M	31G	1.8T
vocab size	200k	5M	16M
# ngrams	257M	21 G	300G
size (B)	2G	89G	1.8 T
time (SB)	20 min	8 hours	1 day
time (KN)	2.5 hours	2 days	-

▶ back

# Using Terascale Language Models

Some results from [5]



# A real world phrase-table

Based on the en-fr Europarl

## 467 (en → fr) translations for “European Commission”

```
European Commission ||| Commission européenne  
European Commission ||| Commission  
European Commission ||| la Commission européenne  
European Commission ||| Commission européenne ,  
European Commission ||| de la Commission européenne  
(...)
```

## 98 (fr → en) translations for “cultures”

```
cultures ||| agriculture  
cultures ||| arable  
cultures ||| crop production  
cultures ||| cultivation  
cultures ||| cultural content  
cultures ||| cultural history  
cultures ||| drug crops  
cultures ||| farming  
cultures ||| farms  
cultures ||| identities  
cultures ||| language  
cultures ||| plants  
(...)
```

# A real world phrase-table

Based on the en-fr Europarl

## 467 (en → fr) translations for “European Commission”

```
European Commission ||| Commission européenne  
European Commission ||| Commission  
European Commission ||| la Commission européenne  
European Commission ||| Commission européenne ,  
European Commission ||| de la Commission européenne  
(...)
```

## 98 (fr → en) translations for “cultures”

```
cultures ||| agriculture  
cultures ||| arable  
cultures ||| crop production  
cultures ||| cultivation  
cultures ||| cultural content  
cultures ||| cultural history  
cultures ||| drug crops  
cultures ||| farming  
cultures ||| farms  
cultures ||| identities  
cultures ||| language  
cultures ||| plants  
(...)
```

# A real world phrase-table

Based on the en-fr Europarl

## 672 translations for '!'

```
! ||| !!!  
! ||| !!  
! ||| ! |||  
! ||| : non !  
...  
! ||| , dit-on partout !  
! ||| , exigez que  
! ||| , exigez  
! ||| , il est primordial que la  
! ||| , il est primordial que  
...  
! ||| Messieurs , il est primordial que la  
! ||| Messieurs , il est primordial  
...
```

▶ back

mais là-dessus je voudrais marquer sinon un désaccord , du moins des nuances sur deux points .

but I would like to indicate otherwise a disagreement , at least the nuances on two points

From Europarl 2008

▶ back

n' y a -t-il pas ici deux poids , deux mesures ?

is there not here two weights , two measures ?

From Europarl 2008

▶ back

en réalité , les entrepreneurs sont plus souvent comparables à des joueurs qui espèrent toucher le *pactole* .

in reality , the entrepreneurs are more often comparable to players who are hoping to touch the *gold mine* .

From Europarl 2008

▶ back

les investisseurs plus vigilants *achètent* déjà en grand nombre , par exemple dans le *coin* de Banskó .

investors more vigilant *achètent* already in great numbers , for example in the *corner* of Banskó .

From NewsTest 2008

▶ back

*l' avocat des familles sinistrées Igor Veleba veut obtenir de l' hôpital de Motol un dédommagement de 12 millions de couronnes plus les dépens .*

*the lawyer of Igor Veleba affected families to obtain the hospital Motol compensation of 12 million kronor more expense .*

From NewsTest 2008

▶ back