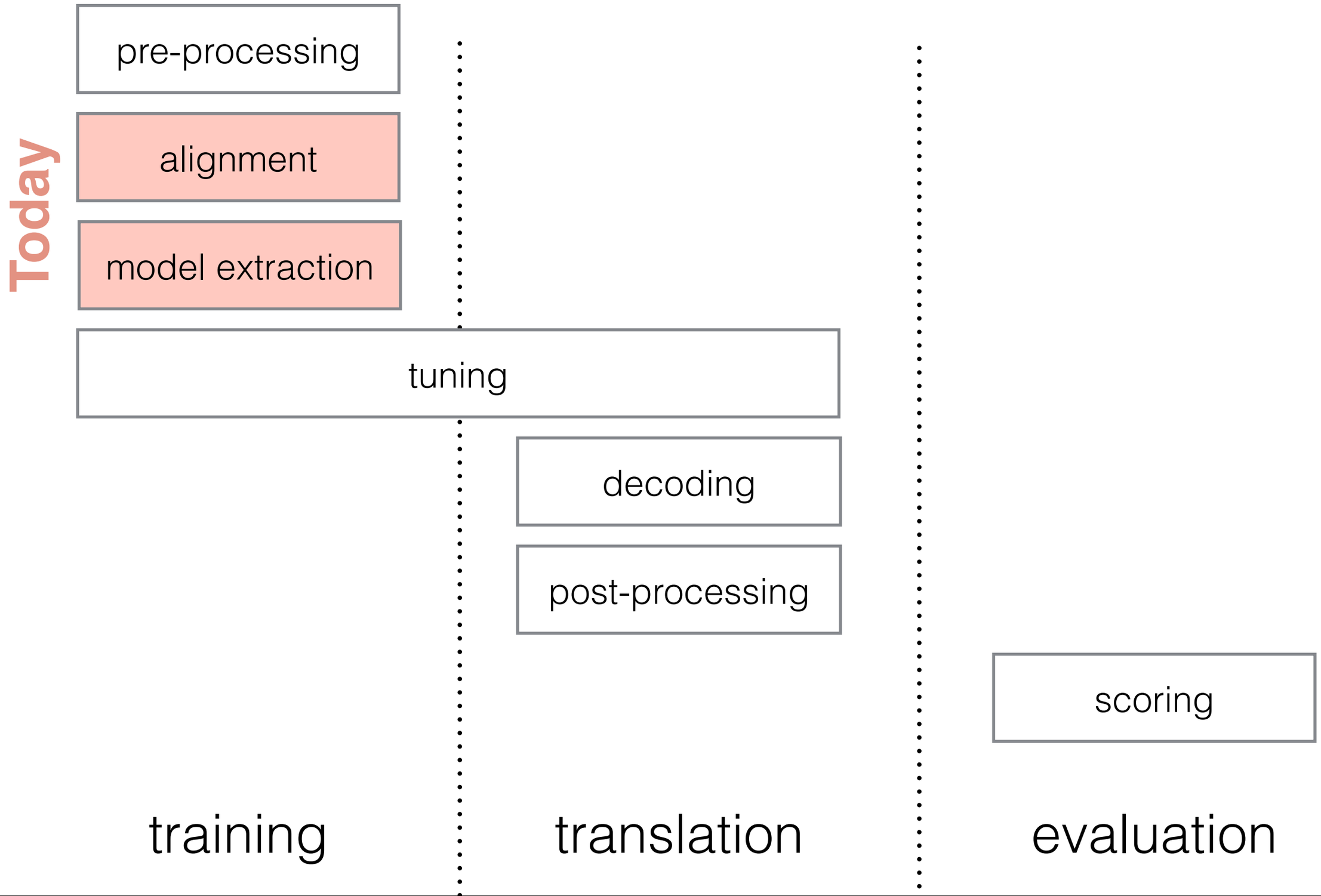
Homework 1

Rank	Handle	#0 Scalar (% N)	#1 Alignment (1-AER)
1	query	77.77	0.63
2	ksk	2.00	0.37
3	default	100.00	0.32
4	sharon	93.00	-inf
5	alopez	89.00	-inf
6	BersaKAIN	56.01	-inf
7	HarryTheHB	5.00	-inf
8	jay	-1.00	-inf
9	adithya.r	-inf	-inf
	debmalyachkrbrty	-inf	-inf
	nx.niuxiang	-inf	-inf

Big Picture


Today
pre-processing
alignment
model extraction
tuning
decoding
post-processing
scoring
training
translation
evaluation

Lexicalized alignment

Brown et al. (1993)

- Model 1: source words generate target words in any order
- Model 2: *absolute* distortion model for reordering
- Model 3: fertility of source words (0+), sentence-length based NULLs
- Model 4: lexicalized, *relative* distortion model
- Model 5: removes mathematical deficiency
- HMM Alignment: relative distortion model

Model 3

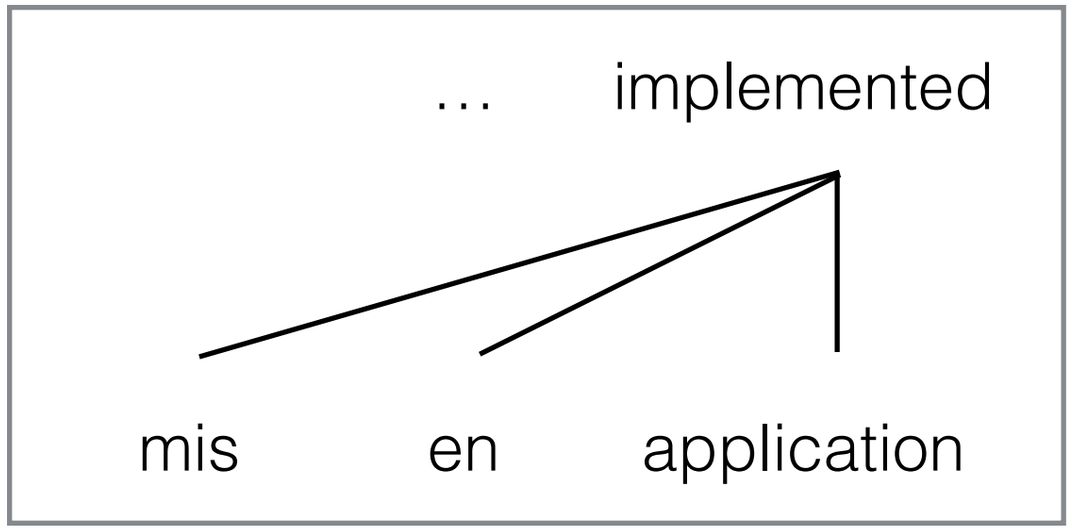
	parameter	*example*
fertility step	$n(\phi \mid e)$	*n(2 | zum) = 0.7*
NULL insertion step	p_1	p_1 *= 0.2*
lexical translation step	$t(f \mid e)$	*t(the | zum) = 0.4*
distortion step	$d(j \mid i, l, m)$	*d(6 | 5) = 0.8*

- Adds fertility
- Initialized with Model 2 Viterbi alignment, then hill-climbs to Model 3 Viterbi approximation through random perturbations of the alignment

Today

- Producing phrase tables
- With direct models of phrasal alignment
- With the IBM models and extraction heuristics

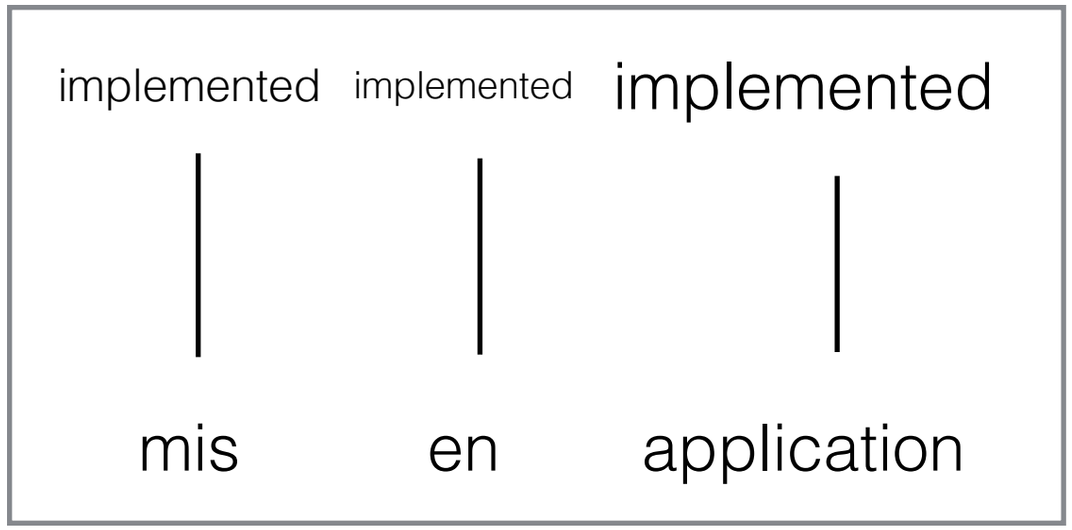
Lexical Alignment



P(*mis en application* | *implemented*)
= t(*mis* | *implemented*)
x t(*en* | *implemented*)
x t(*application* | *implemented*)

- As we saw last week, these translation probabilities are all generated independently

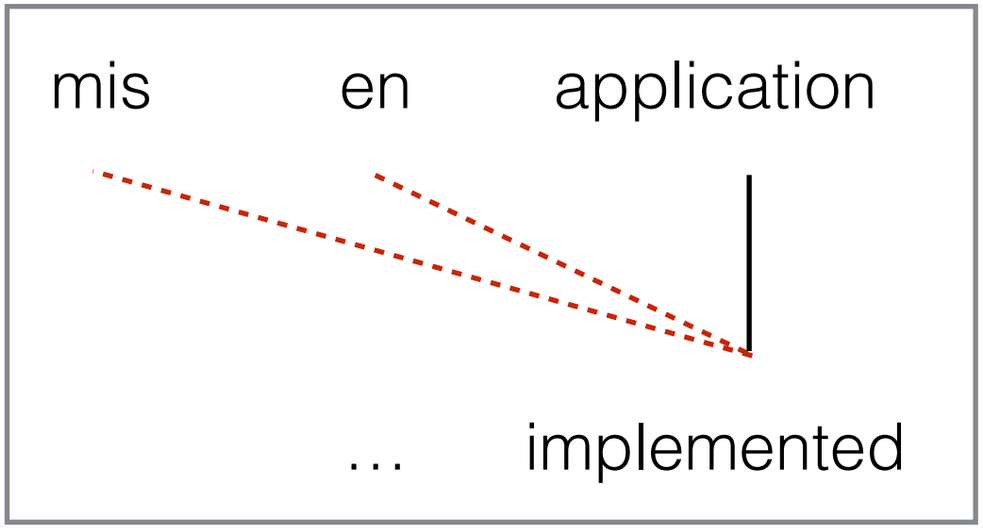
Lexical Alignment



P(*mis en application* | *implemented*)
= ϕ(3 | *implemented*)
x t(*mis* | *implemented*)
x t(*en* | *implemented*)
x t(*application* | *implemented*)

- As we saw last week, these translation probabilities are all generated independently
- Even with Model 3's fertility, the translations are still independent!

Other problems



- *Asymmetry*: the alignment is impossible if modeled this way
- Words are not atomic units of meaning

manger *le* *morceau*
“eat the piece”
spill the beans

Modeling failures

- Some drawbacks of word based alignments
  - ~~All reorderings have the same probability~~ **MODEL 2**
  - Alignments are independent
  - No notion of multiword alignments
  - Alignments are asymmetric
  - No morphology
  - No syntax

Modeling failures

- Some drawbacks of word based alignments
  - ~~All reorderings have the same probability~~ **MODEL 2**
  - ~~Alignments are independent~~ **HMM MODEL**
  - No notion of multiword alignments
  - Alignments are asymmetric
  - No morphology
  - No syntax

Modeling failures

- Some drawbacks of word based alignments
  - ~~All reorderings have the same probability~~ **MODEL 2**
  - ~~Alignments are independent~~ **HMM MODEL**
  - ~~No notion of multiword alignments~~ **MODEL 3**
  - Alignments are asymmetric
  - No morphology
  - No syntax

Phrasal alignment

- All models are deficient
- But why not model phrases directly?
  - (spoiler) decoding is phrase-based, why not alignment?
  - e.g., treat phrase pairs as atomic, with a single alignment between them

A joint phrase-based model

Marcu & Wong (2002)

- Model

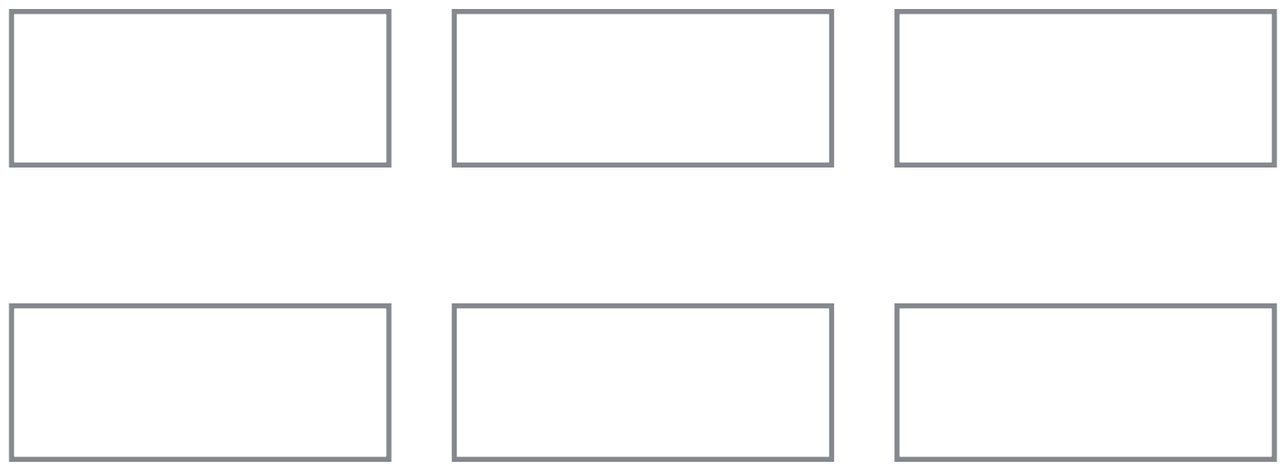
> Choose a number *l* of phrase pairs *(e, f)*
> Choose a 1-1 alignment
> Sample each of the phrase pairs from a big joint distribution

A joint phrase-based model

Marcu & Wong (2002)

- Model

Choose a number *l* of phrase pairs *(e, f)*
Choose a 1-1 alignment
Sample each of the phrase pairs from a big joint distribution



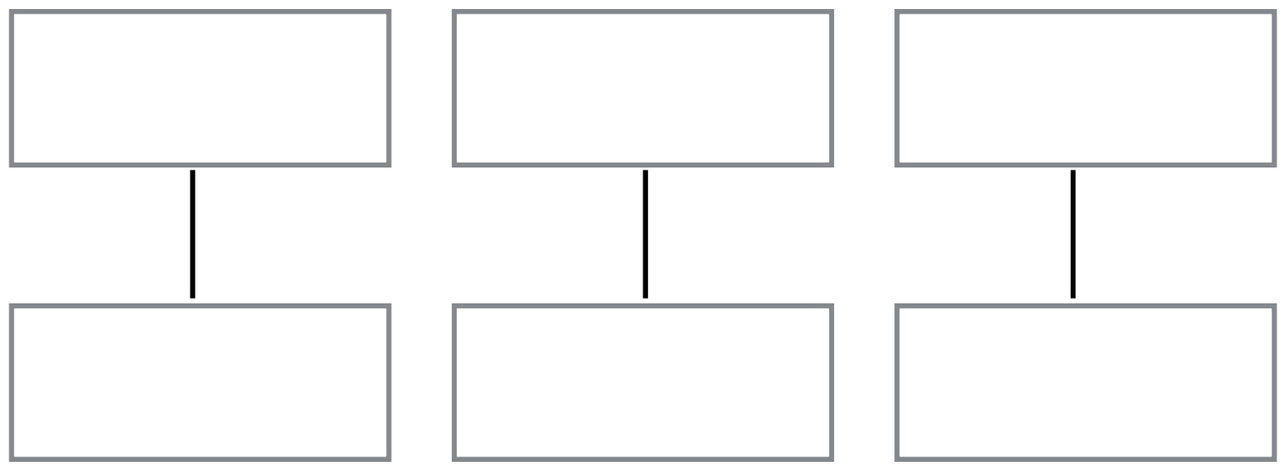
p(l = 3)

A joint phrase-based model

Marcu & Wong (2002)

- Model

Choose a number *l* of phrase pairs *(e, f)*
Choose a 1-1 alignment
Sample each of the phrase pairs from a big joint distribution



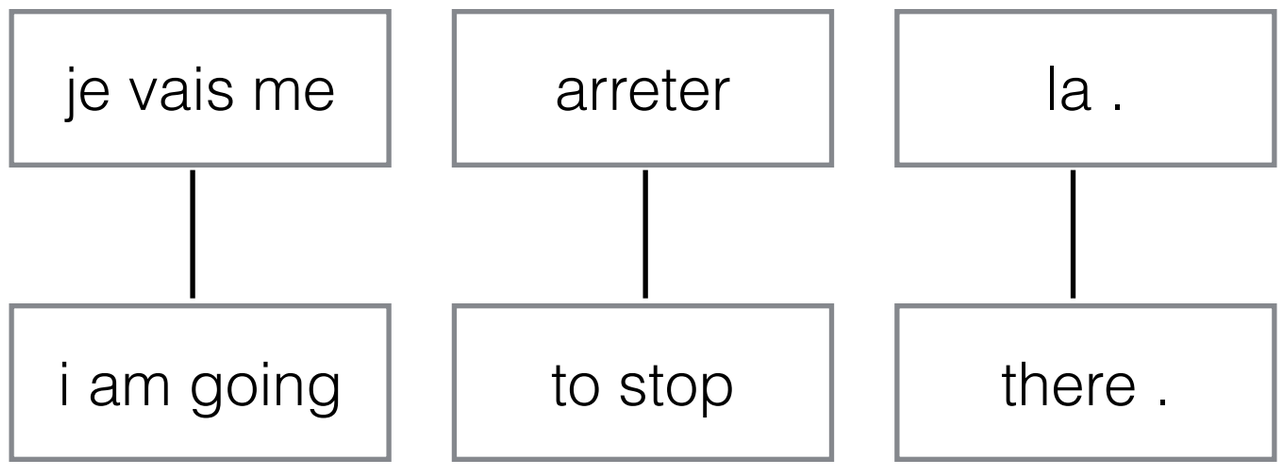
p(l = 3)
x p(a)

A joint phrase-based model

Marcu & Wong (2002)

- Model

Choose a number *l* of phrase pairs *(e, f)*
Choose a 1-1 alignment
Sample each of the phrase pairs from a big joint distribution



p(l = 3)
x p(a)
x p(je vais me, I am going)
x p(arreter, to stop)
x P(la ., there .)

EM Learning

- Imagine EM for this model:
  - Compute expectations (soft counts, weighted by the alignments) over phrase pairs using current estimate of t(e,f)
  - Sum expected counts and normalize to get new estimates for t(e,f)

EM learning

- Basically, a phrase-based analogue of the EM algorithm for Model 2

initialize parameters *t* and *q* to something
repeat until convergence
    for every sentence
        for every target position *j*
            for every source position *i*
                count(f_j, e_i) += P(a_i = j | e_i, f_j)
                count(e_i) += P(a_i = j | e_i, f_j)
                count(*j, i, l, m*) += P(a_i = j | e_i, f_j)
                count(*i, l, m*) += P(a_i = j | e_i, f_j)
    t(f | e) = count(f, e) / count(e)
    q(j | i, l, m) = count(j, i, l, m) / count(i, l, m)

Discuss

5 minutes, with a neighbor or two

$$\mathcal{L}(t|(E,F)) = \sum_{e,f} \prod_{i,j} t(e_i, f_j)$$

Discuss

5 minutes, with a neighbor or two

- Find the maximum likelihood solution

$$\mathcal{L}(t|(E,F)) = \sum_{e,f} \prod_{i,j} t(e_i, f_j)$$

for the corpus

and the program has been implemented

le programme a été mis en application

Solution

$$\mathcal{L}(t|(E,F)) = \sum_{e,f} \prod_{i,j} t(e_i, f_j)$$

e	f	t(e,f)

Solution

$$\mathcal{L}(t|(E,F)) = \sum_{e,f} \prod_{i,j} t(e_i, f_j)$$

e	f	t(e,f)
and the program has been implemented	le programme a été mis en application	1.0

Solution

$$\mathcal{L}(t|(E,F)) = \sum_{e,f} \prod_{i,j} t(e_i, f_j)$$

e	f	t(e,f)
and the program has been implemented	le programme a été mis en application	1.0

Solution

$$\mathcal{L}(t|(E,F)) = \sum_{e,f} \prod_{i,j} t(e_i, f_j)$$

e	f	t(e,f)
and the program has been implemented	le programme a été mis en application	1.0

- EM has a degenerate solution, which is to memorize the training data

Other problems

- Given sentences of length e and f, how many possible phrase pairs are there?

Other problems

- Given sentences of length e and f, how many possible phrase pairs are there?
  - $O(e^2f^2) \approx O(n^4)$

Other problems

- Given sentences of length e and f, how many possible phrase pairs are there?
  - $O(e^2f^2) \approx O(n^4)$
- One solution: cap phrase length (e.g., 6)

Other problems

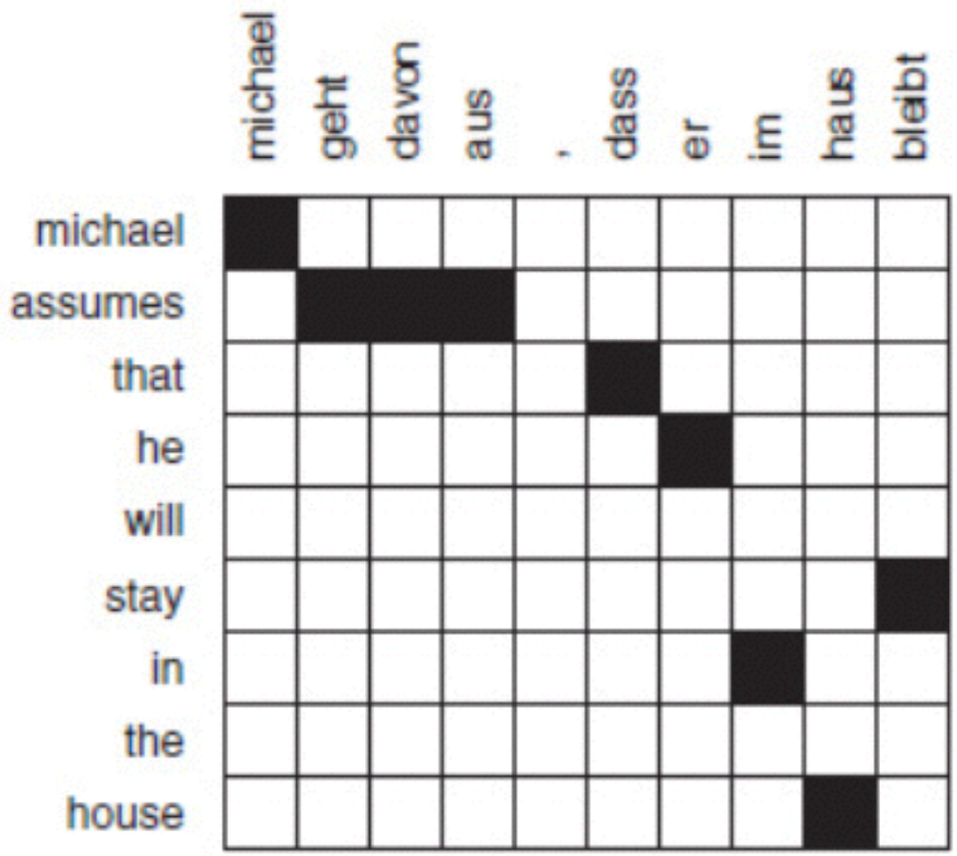
- Given sentences of length e and f, how many possible phrase pairs are there?
  - $O(e^2f^2) \approx O(n^4)$
- One solution: cap phrase length (e.g., 6)
- Another solution: Bayesian framework with priors that favor small phrases (DeNero et al., 2008)

Deeper problems

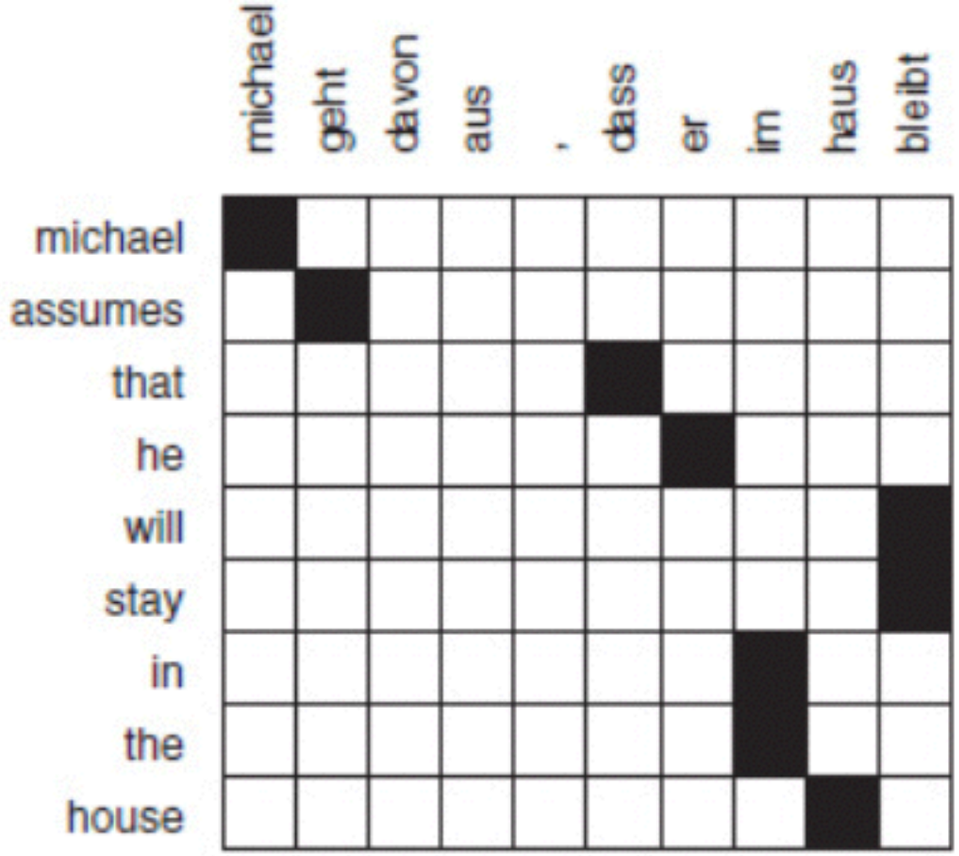
- Complexity (DeNero & Klein, 2008)
  - Computing the most likely alignment is NP hard
  - Computing expectations is #P-complete
- The segmentation is an extra hidden variable in addition to the alignment
- Solutions require arbitrary limits to the model (capped phrase length) or sampling approaches (which are slow and don’t scale)

Heuristic phrase extraction

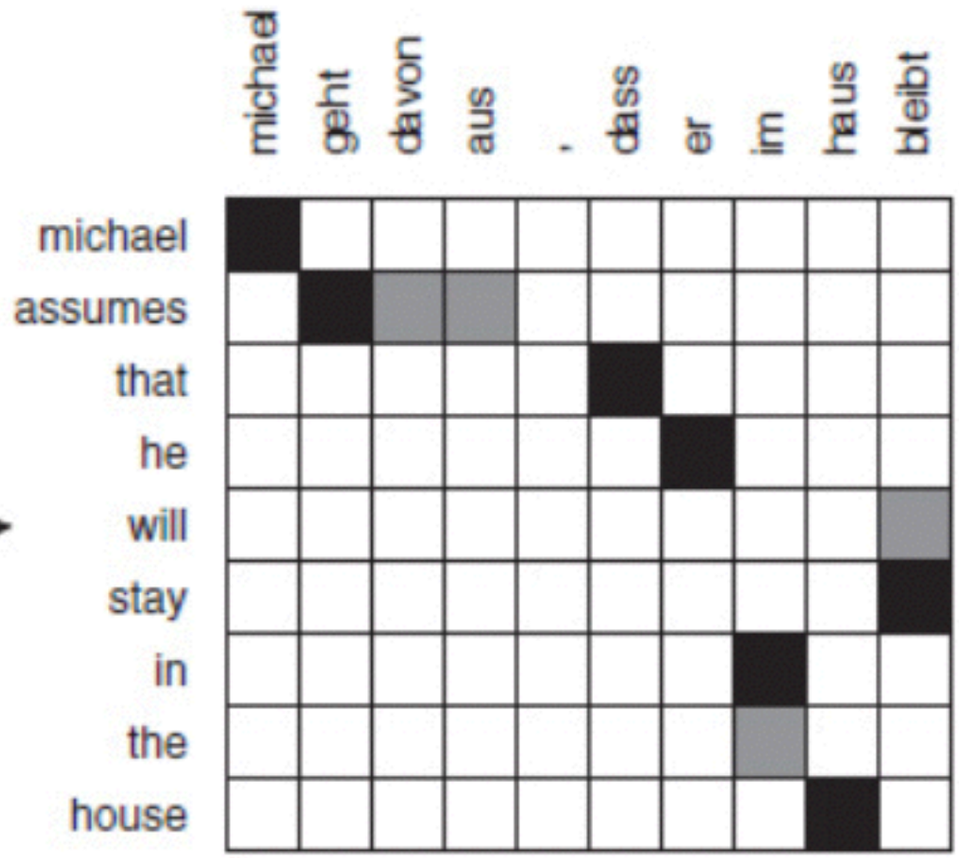
- New approach using word alignment models
  - Compute Viterbi (most probable) alignments
  - Combine models of (e | f) and (f | e)
  - Extract phrases consistent with alignments



English to German



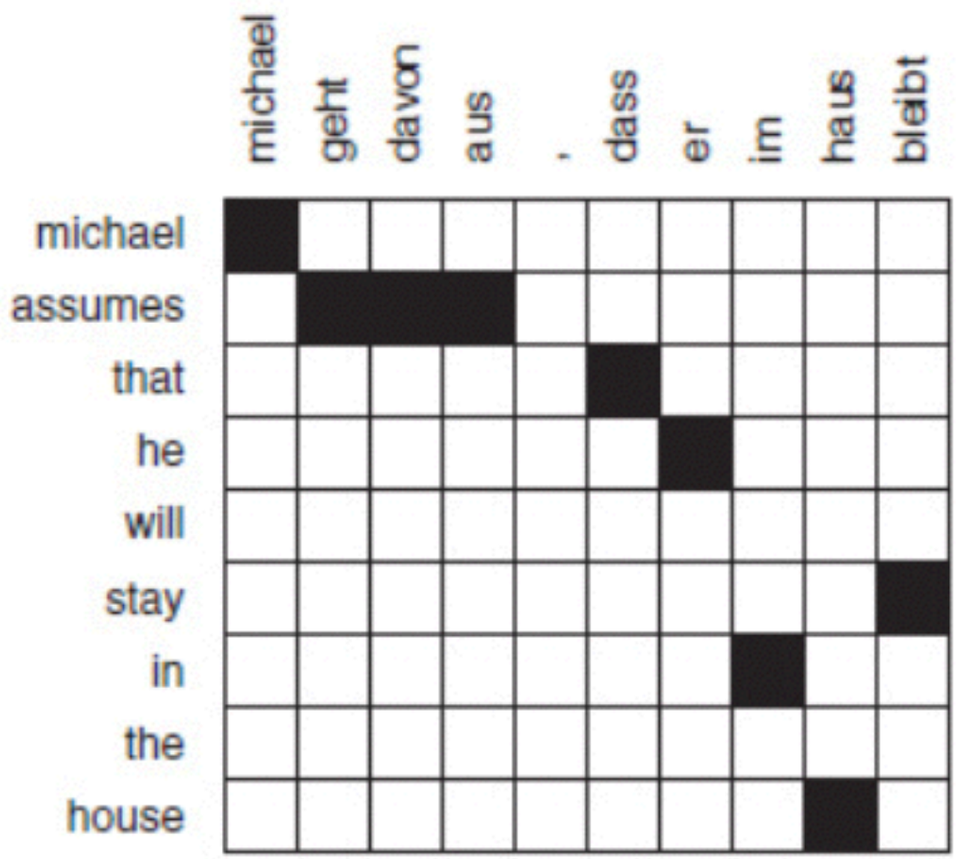
German to English



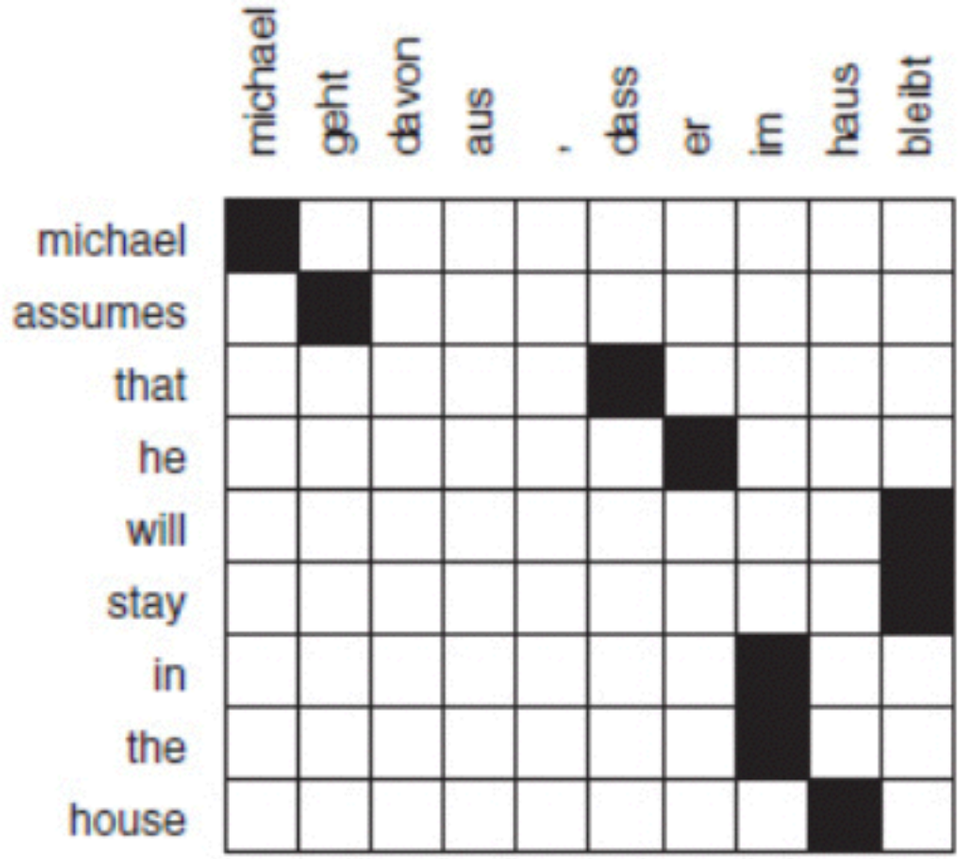
Intersection / Union

Learn separate models

- Common practice: 5 iterations of Model 1, 5 iterations of HMM, 3 of Model 3, and 3 of Model 4



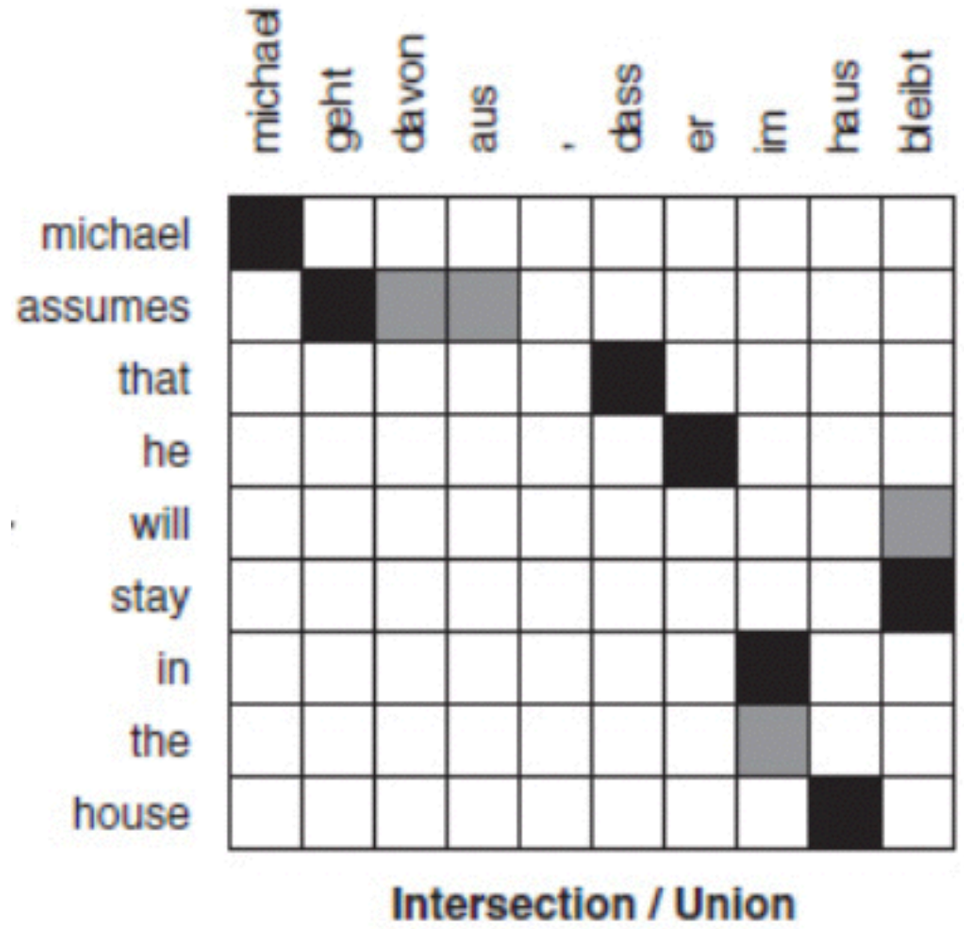
English to German



German to English

Combine them

- Combination is heuristic: start with intersection (black), add unaligned neighboring points from union



Phrase extraction

- Extract phrases *consistent with* the merged alignments
- Blocks whose words' alignment points are wholly contained in the block

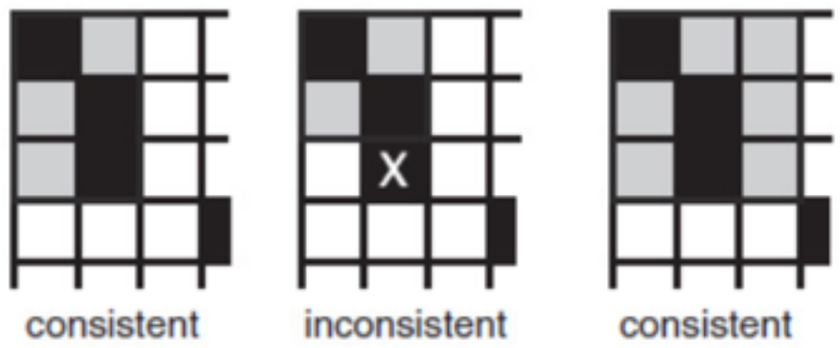


Figure 5.4 Definition of phrase pairs being consistent with a word alignment: All words have to align to each other. This is true in the first example, violated in the second example (one alignment point in the second column is outside the phrase pair), and true in the third example (note that it includes an unaligned word on the right).

Phrase extraction

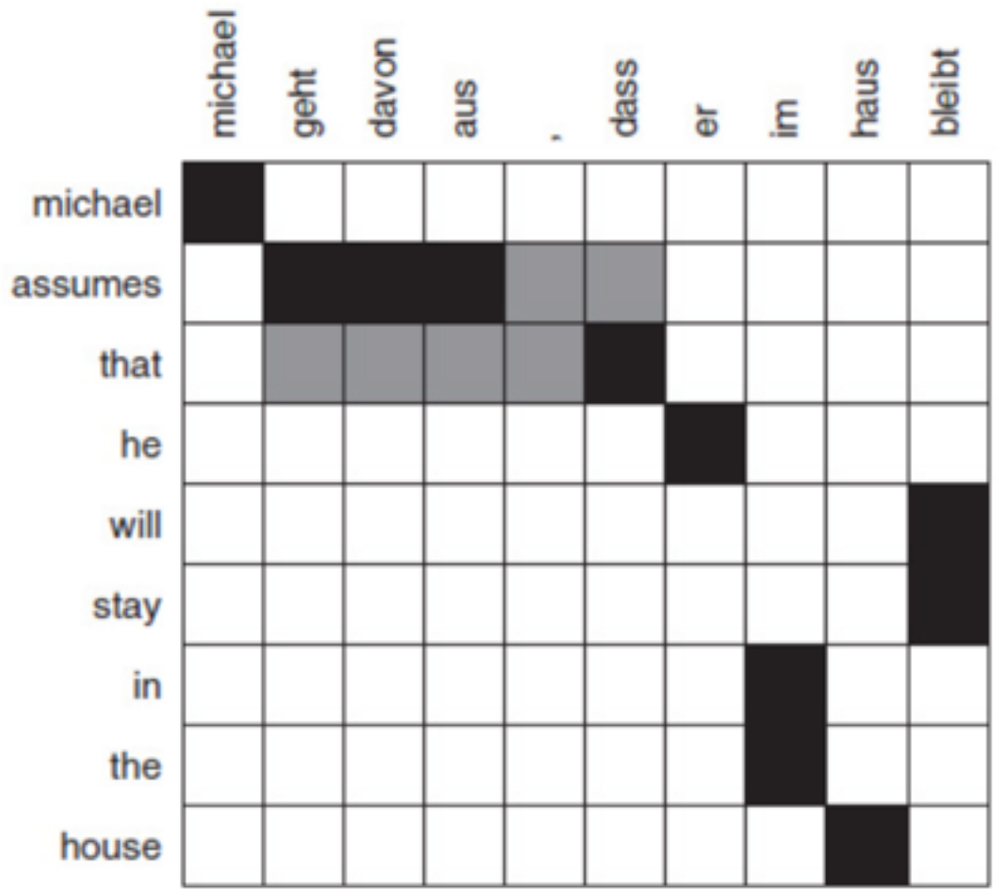


Figure 5.3 Extracting a phrase from a word alignment: The English phrase *assumes that* and the German phrase *geht davon aus, dass* are aligned, because their words are aligned to each other.

Scores

- You could imagine counting up all phrases and normalizing to produce a single joint distribution

$$t(e, f) = \frac{\textbf{count}(e, f)}{\sum_{e', f'} \textbf{count}(e', f')}$$

- This isn't actually common practice, but those details will come later

Summary

- Modeling phrase alignments — while intuitive — introduces some surprising complexity that currently preclude their use

- The best approach is the cobbled-together pipeline of word alignments, merging, and heuristic extraction

- This approach scales to very large datasets

- The end result is a phrase table