Statistical Machine Translation: Phrase-based Models (Part I)

Jakub Waszczuk

Heinrich Heine Universität Düsseldorf

Winter Semester 2018/19

Outline

Outline

Alignment Evaluation

Alignment Applications

- Starting point for more refined phrase-based statistical translation systems
- Automatic extraction of bilingual lexica and terminology from corpora
- Transfer text analysis tools (morphologic analyzers, part-of-speech taggers, parsers) from a rich-resource language to a low-resource langauge
- ...

Alignment Evaluation

- We have a translation model, which makes predictions as to probable alignments
- We wish to tell how good those predictions are

Alignment Evaluation

How this is done?

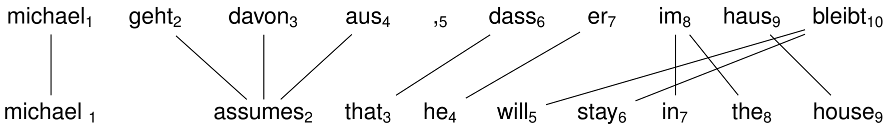
Compare:

- The predicted ($\approx$ Viterbi) alignments
- With gold standard alignments (made by hand)

Issue

We compare two objects with possibly different form:

- Predicted alignment *functions* ($\{1, \ldots, m\} \to \{0, \ldots, m\}$)
- Gold standard alignment *relations* ($\{1, \ldots, m\} \times \{0, \ldots, m\}$)



Alignment Evaluation

Bidirectional alignments

Given a bilingual EN-GE corpus, we can use EM to:

- Train a GE $\rightarrow$ EN translation model
- Train an EN $\rightarrow$ GE translation model

Having both translation models and a test sentence:

- Determine the best GE $\rightarrow$ EN alignment function A_1
- Determine the best EN $\rightarrow$ GE alignment function A_2

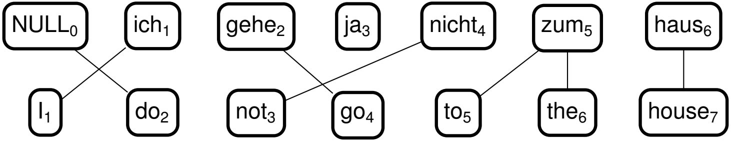
Issue 2

For a given sentence pair, which aligment – A_1 or A_2 – should we chose as the best one?

Alignment Evaluation

Functions as relations

Alignment functions can be represented as relations, for instance:



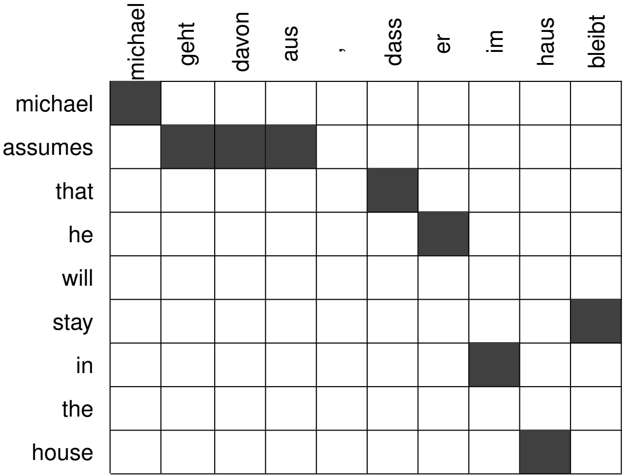
as:

$$\{(1,0),(2,0),(3,4),(4,2),(5,5),(6,6),(7,6)\}$$

As a result, we can perform standard set-theoretic operations on alignments (intersection, union, etc.)

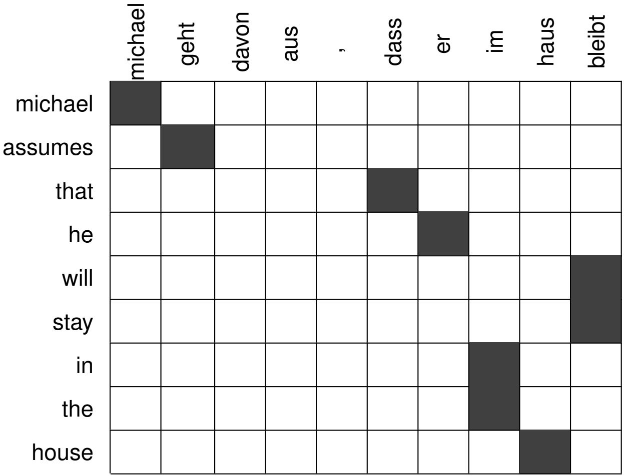
Alignment Evaluation

English to German



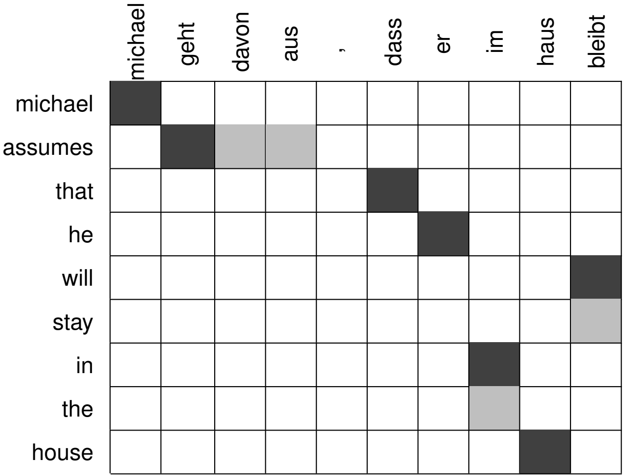
Alignment Evaluation

German to English



Alignment Evaluation

Intersection (black) / Union (black and gray)



Alignment Evaluation

Union

The set of all the alignment arcs present in either A_1 or A_2 (inversed):

$$A := A_1 \cup A_2^{-1} \tag{1}$$

Useful if we want to be sure to predict as many gold standard arcs as possible (*recall*)

Intersection

The set of all the alignment arcs present in both A_1 and A_2 (inversed):

$$A := A_1 \cap A_2^{-1} \tag{2}$$

Useful if we want to be sure to mostly predict only gold standard arcs (*precision*)

Golden mean

- In practice, a solution somewhere between the sum and the intersection is typically closest to the gold standard
- Various heuristics have been designed to deal with this problem

Alignment Evaluation

Gold standard

Manual alignment is not easy task (the notion of „correspondence“ between words is subjective), hence the gold standard often consists of:

- The set of *possible* arcs M
- The set of *sure* arcs $S \subseteq M$

Alignment error rate

$$AER(S, M, A) = 1 - \frac{|A \cap S| + |A \cap M|}{|S| + |A|} \quad (3)$$

AER returns a value between 0 (the best) and 1 (the worst).

Property

A perfect error rate of 0 is achieved when:

- Every sure arc is predicted ($S \subseteq A$)
- Every predicted arc is possible ($A \subseteq M$)

Outline

Phrase-based Translation

Motivation

- All IBM models based on lexical translation
- Context often required to correctly translate a word
- Non-local context may be needed

Er_1	$\textbf{macht}_2$	shon_3	wieder_4	$\textbf{blau}_5$
He_1	$\textbf{pulls}_2$	$\textbf{a}_3$	$\textbf{sickie}_4$	again_5

- But immediate context better than no context at all

Example

- *Viel Spass!* → *Lots of pleasure!*
- *Viel Spass!* → *Have fun!*

Phrase-based Translation

Phrase translation function

The main building block of phrase-based translation.

- R_F – the set of *phrases* in the input (foreign) language, $R_F \subset V_F^+$
- R_E – the set of *phrases* in the output (English) language, $R_E \subset V_E^*$
- **Phrase translation function** – given $f \in R_F$ and $e \in R_E$:

$$P(e \mid f) \in [0, 1] \tag{4}$$

- Since P is a conditional distribution, for each $f \in R_F$:

$$\sum_{e \in R_E} P(e \mid f) = 1 \tag{5}$$

Phrase-based Translation

Memory usage

- In practice, given $f \in R_F$, we only care about the possible translations $R_E(f) \subset R_E$
- Also, for each $e \in R_E :\ e \notin R_E(f)$:

$$P(e \mid f) = 0 \tag{6}$$

Example

Possible phrase translation tables for two German phrases:

natuerlich

e	$P(e\|f)$
of course	0.5
naturally	0.3
of course ,	0.15
, of course ,	0.05

Viel Spass !

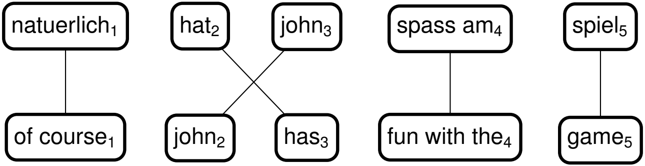
e	$P(e\|f)$
Have fun !	0.95
Lots of fun !	0.05

Phrase-based Translation

Translation process

- Split input sentence into phrases, each belonging to P_F
- Translate each phrase independently, according to phrase translation function P
- Reorder the resulting output phrases

Example



- Better than lexical translation (*am* → *with the*, makes sense in the context of *spass*)
- It's not really linguistically motivated (*fun with the* is not a linguistic phrase)

Phrase-based Translation

Advantages

In comparison with word-based (IBM) models:

- Translating word groups helps in resolving translation ambiguities (words not the best atomic units for translation)
- Possible to ,,memorize" translations of long phrases, even entire sentences
- Conceptually much simpler – don't need the complex notions of fertility, insertion, deletion (not allowing arbitrary adding and dropping of words makes more sense)

Phrase-based Translation

Determining possible phrases

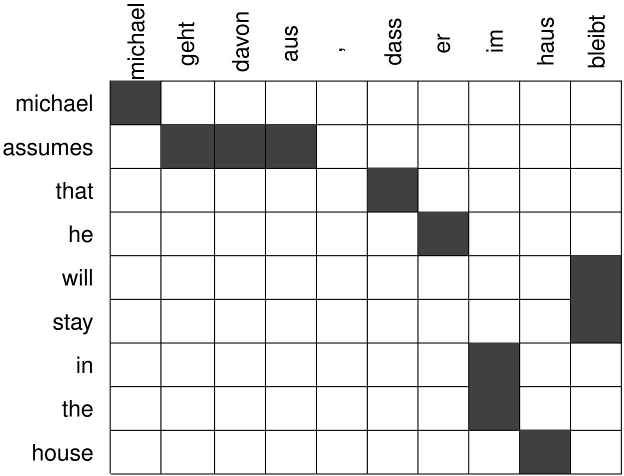
- Word-based translation: virtually any $f \in V_F$ can be translated to any $e \in V_E$
- Phrase-based translation: allowing to translate any $f \in R_F$ to any $e \in R_E$ infeasible
- The first step is, therefore, to determine the possible translation phrase pairs

Outline

1. Alignment Evaluation
2. Phrase-based Translation
3. Extracting Phrase Pairs

Extracting Phrase Pairs

Input



Extracting Phrase Pairs

Goal

Given sentence pair $(\boldsymbol{f}, \boldsymbol{e})$ and alignment relation A, determine all phrase pairs $(\bar{f}, \bar{e})$ consistent with A.

Consistency

We say that a phrase pair $(\bar{f}, \bar{e})$ is *consistent* with alignment A if:

- $\bar{e}$ is a contiguous fragment in $\boldsymbol{e}$
- $\bar{f}$ is a contiguous fragment in $\boldsymbol{f}$
- for each $e_i \in \bar{e}$ and each corresponding alignment point $(i, j) \in A$: $f_j \in \bar{f}$
- for each $f_j \in \bar{f}$ and each corresponding alignment point $(i, j) \in A$: $e_i \in \bar{e}$
- there exist $e_i \in \bar{e}$ and $f_j \in \bar{f}$ such that $(i, j) \in A$

Extracting Phrase Pairs

Goal

Given sentence pair $(\boldsymbol{f}, \boldsymbol{e})$ and alignment relation A, determine all phrase pairs $(\bar{f}, \bar{e})$ consistent with A.

Consistency

We say that a phrase pair $(\bar{f}, \bar{e})$ is *consistent* with alignment A if:

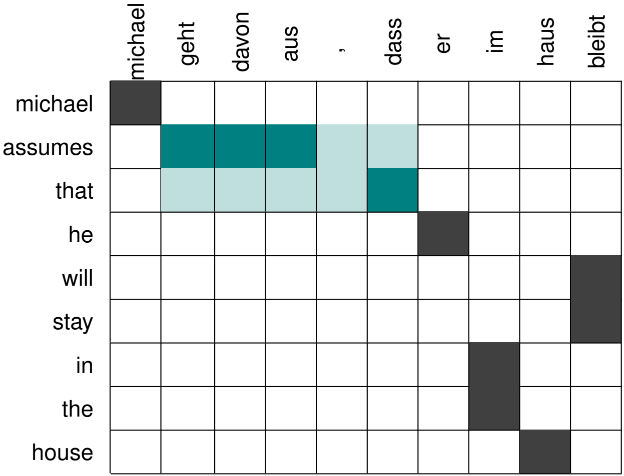
- $\bar{e}$ is a contiguous fragment in $\boldsymbol{e}$
- $\bar{f}$ is a contiguous fragment in $\boldsymbol{f}$
- for each $e_i \in \bar{e}$ and each corresponding alignment point $(i, j) \in A$: $f_j \in \bar{f}$
- for each $f_j \in \bar{f}$ and each corresponding alignment point $(i, j) \in A$: $e_i \in \bar{e}$
- there exist $e_i \in \bar{e}$ and $f_j \in \bar{f}$ such that $(i, j) \in A$

Graphically:

- $(\bar{f}, \bar{e})$ must correspond to a non-empty (at least one marked cell) rectangle R
- for each column and row that intersects R, all its marked cells must be in R

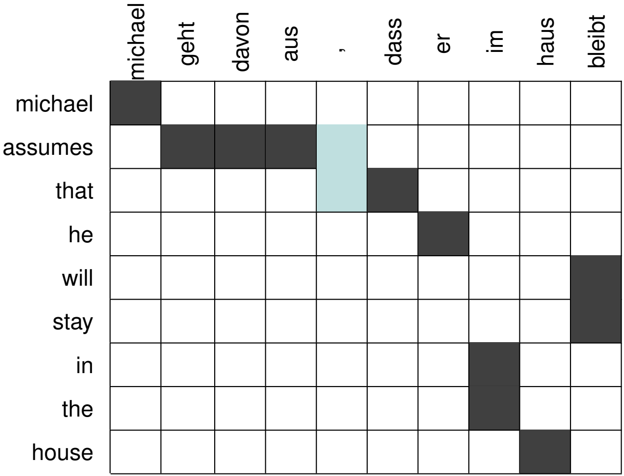
Extracting Phrase Pairs

Consistency example



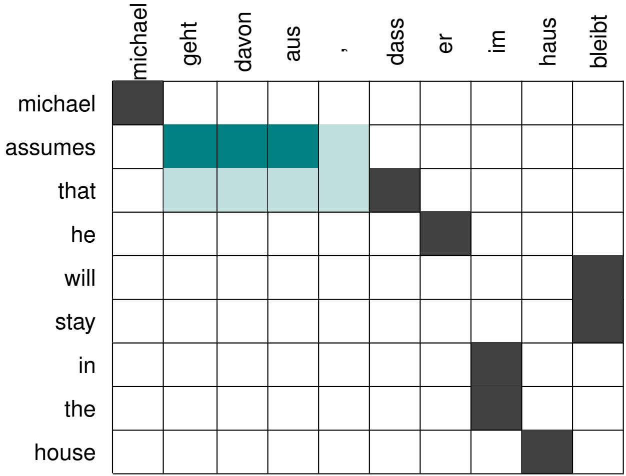
Extracting Phrase Pairs

Consistency example



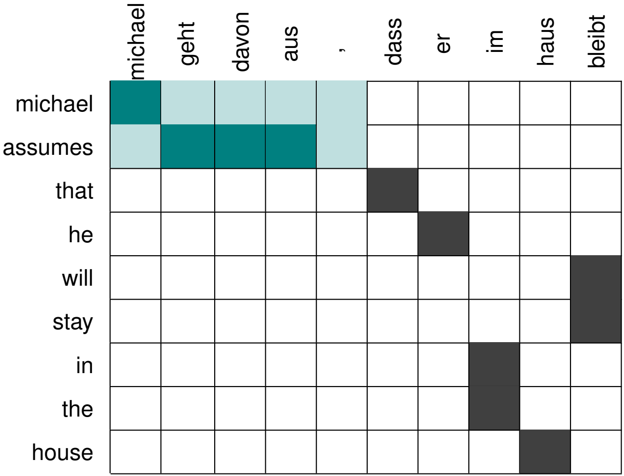
Extracting Phrase Pairs

Consistency example



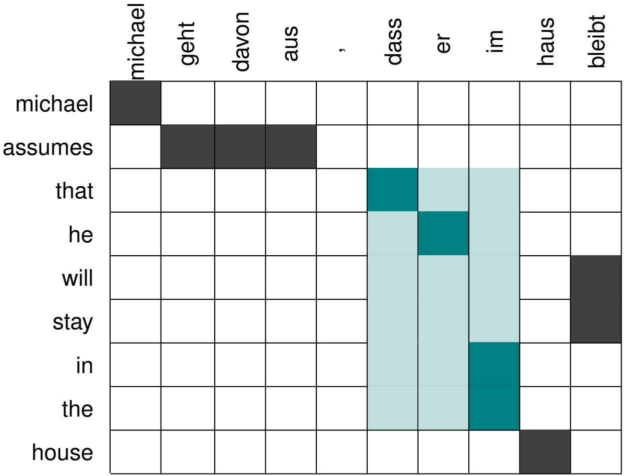
Extracting Phrase Pairs

Consistency example



Extracting Phrase Pairs

Consistency example



Extracting Phrase Pairs

Algorithm (naive)

Given sentence pair $(\boldsymbol{e}, \boldsymbol{f})$ and alignment relation A:

- $F := \emptyset$
- for j, j' : $1 \leq j \leq j' \leq m$:
  - let $\bar{e} = (e_j, \ldots, e_j')$
  - for i, i' : $1 \leq i \leq i' \leq n$:
    - let $\bar{f} = (f_i, \ldots, f_i')$
    - if $(\bar{e}, \bar{f})$ consistent with A: $F := F \cup (\bar{e}, \bar{f})$

Extracting Phrase Pairs

Algorithm (improved)

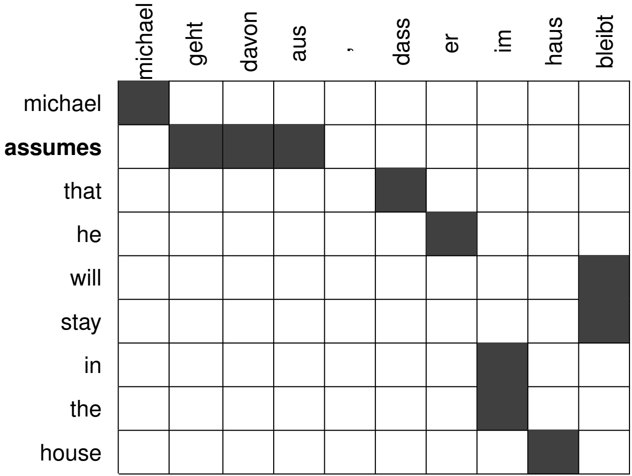
Given sentence pair $(\boldsymbol{e}, \boldsymbol{f})$ and alignment relation A:

- $F := \emptyset$
- for $j, j' : 1 \leq j \leq j' \leq m$:
  - let $\bar{e} = (e_j, \ldots, e_j')$
  - determine the minimal matching phrase $\bar{f} = (f_i, \ldots, f_i')$
  - add $(\bar{e}, \bar{f})$, as well as all its extensions covering the neighboring non-aligned words, to F

See the complementary material for more information.

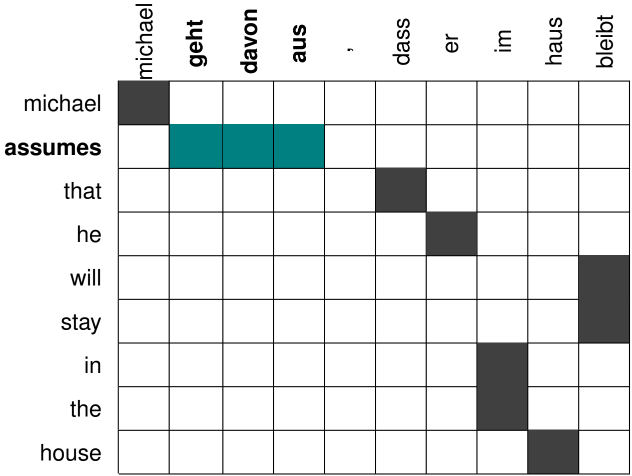
Extracting Phrase Pairs

Phrase extraction example



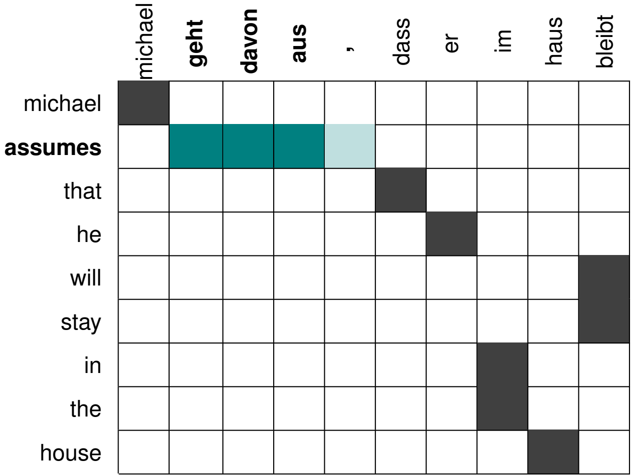
Extracting Phrase Pairs

Phrase extraction example



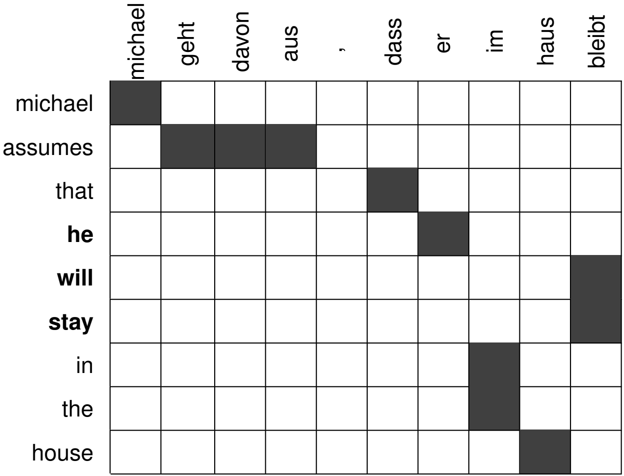
Extracting Phrase Pairs

Phrase extraction example



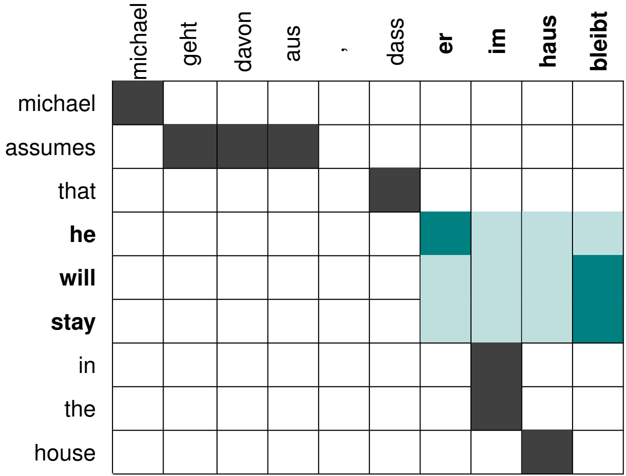
Extracting Phrase Pairs

Phrase extraction example



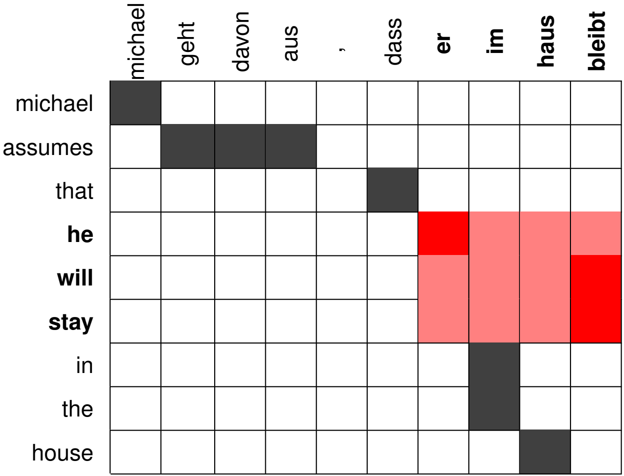
Extracting Phrase Pairs

Phrase extraction example



Extracting Phrase Pairs

Phrase extraction example



Next time

Next time

Further info on phrase-based translation:

- Reordering
- Translation probability
- Parameter estimation