Machine Translation as Sequence Modelling

Philipp Koehn

23 July 2016

Sequence Model

- Input sentence

Eu quero ouvir uma apresentação muito interessante.

- Output

I want to listen to a very interesting presentation.

- Idea: produce output one word at a time

N-Gram Model

- Input sentence

  Eu quero ouvir uma apresentação muito interessante.

- Output
  - $p(\text{I})$
  - $p(\text{want}|\text{I})$
  - $p(\text{to}|\text{I want})$

- We learned how to do this today

- Major flaw: Output is not conditioned on input

Conditioning on Input

- Input sentence

Eu quero ouvir uma apresentação muito interessante.

- Output
  - p(I|Eu quero ouvir uma apresentação muito interessante.)
  - p(want|I, Eu quero ouvir uma apresentação muito interessante.)
  - p(to|I want, Eu quero ouvir uma apresentação muito interessante.)

- Conditioning on entire source sentence too sparse to estimate (unlikely that we have seen input sentence before)

1-1 Alignment to Input

Input	Eu	quero	ouvir	uma	...
	|	|	|	|	
Output	I	want	hear	a	
Model	$p(\text{I}\|\text{Eu})$	$p(\text{want}\|\text{quero})$	$p(\text{hear}\|\text{ouvir})$	$p(\text{a}\|\text{uma})$	

- We are slowly getting somewhere
- Open problems
  - we need to move beyond 1-1 alignments
  - where do we get the probabilities from?

ibm model 1

Lexical Translation

- How to translate a word → look up in dictionary

  Haus — house, building, home, household, shell.

- Multiple translations
  - some more frequent than others
  - for instance: house, and building most common
  - special cases: Haus of a snail is its shell

- Note: In all lectures, we translate from a foreign language into English

Collect Statistics

Look at a parallel corpus (German text along with English translation)

Translation of *Haus*	Count
house	8,000
building	1,600
home	200
household	150
shell	50

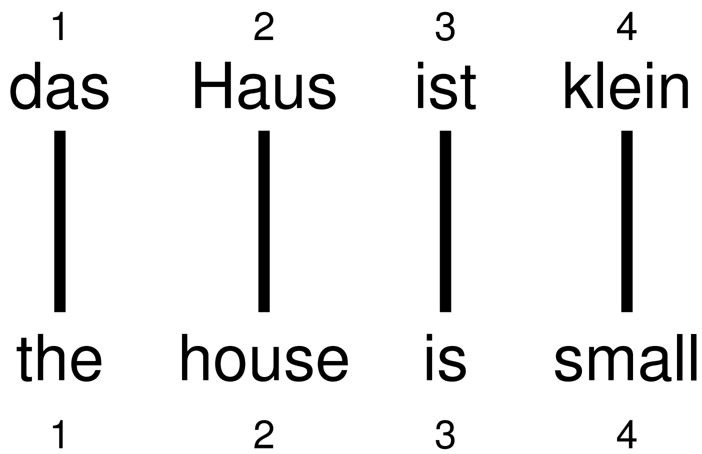
Estimate Translation Probabilities

Maximum likelihood estimation

$$p_f(e) = \begin{cases} 0.8 & \text{if } e = \text{house}, \\ 0.16 & \text{if } e = \text{building}, \\ 0.02 & \text{if } e = \text{home}, \\ 0.015 & \text{if } e = \text{household}, \\ 0.005 & \text{if } e = \text{shell}. \end{cases}$$

Alignment

- In a parallel text (or when we translate), we align words in one language with the words in the other



- Word positions are numbered 1–4

Alignment Function

- Formalizing alignment with an alignment function

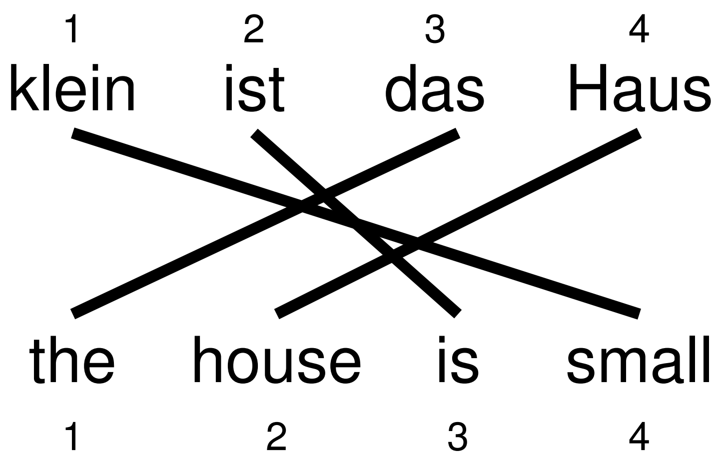
- Mapping an English target word at position i to a German source word at position j with a function $a : i \rightarrow j$

- Example

$$a : \{1 \rightarrow 1, 2 \rightarrow 2, 3 \rightarrow 3, 4 \rightarrow 4\}$$

Reordering

Words may be reordered during translation



$$a : \{1 \rightarrow 3, 2 \rightarrow 4, 3 \rightarrow 2, 4 \rightarrow 1\}$$

IBM Model 1

- Generative model: break up translation process into smaller steps
  - IBM Model 1 only uses lexical translation

- Translation probability
  - for a foreign sentence $\mathbf{f} = (f_1, ..., f_{l_f})$ of length l_f
  - to an English sentence $\mathbf{e} = (e_1, ..., e_{l_e})$ of length l_e
  - with an alignment of each English word e_j to a foreign word f_i according to the alignment function $a : j \rightarrow i$

$$p(\mathbf{e}, a|\mathbf{f}) = \frac{\epsilon}{(l_f + 1)^{l_e}} \prod_{j=1}^{l_e} t(e_j|f_{a(j)})$$

  - parameter ϵ is a normalization constant

Example

das

e	$t(e|f)$
the	0.7
that	0.15
which	0.075
who	0.05
this	0.025

Haus

e	$t(e|f)$
house	0.8
building	0.16
home	0.02
household	0.015
shell	0.005

ist

e	$t(e|f)$
is	0.8
’s	0.16
exists	0.02
has	0.015
are	0.005

klein

e	$t(e|f)$
small	0.4
little	0.4
short	0.1
minor	0.06
petty	0.04

$$\begin{aligned} p(e,a|f) &= \frac{\epsilon}{4^3} \times t(\text{the}|\text{das}) \times t(\text{house}|\text{Haus}) \times t(\text{is}|\text{ist}) \times t(\text{small}|\text{klein}) \\ &= \frac{\epsilon}{4^3} \times 0.7 \times 0.8 \times 0.8 \times 0.4 \\ &= 0.0028\epsilon \end{aligned}$$

Learning Lexical Translation Models

- We would like to estimate the lexical translation probabilities $t(e|f)$ from a parallel corpus

- ... but we do not have the alignments

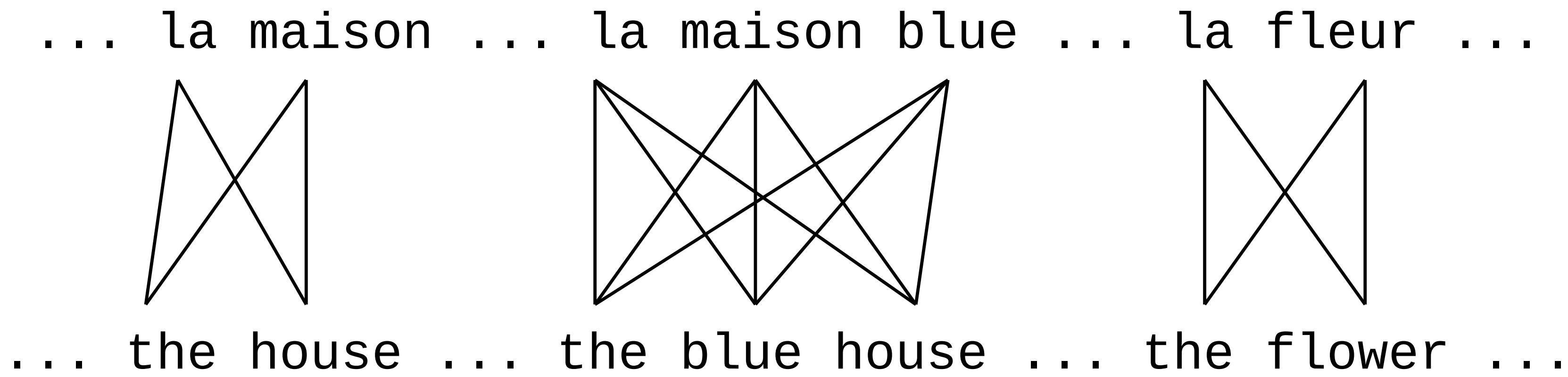
- Chicken and egg problem
  - if we had the *alignments,*
    $\rightarrow$ we could estimate the *parameters* of our generative model
  - if we had the *parameters,*
    $\rightarrow$ we could estimate the *alignments*

EM Algorithm

- Incomplete data
  - if we had *complete data*, would could estimate *model*
  - if we had *model*, we could fill in the *gaps in the data*

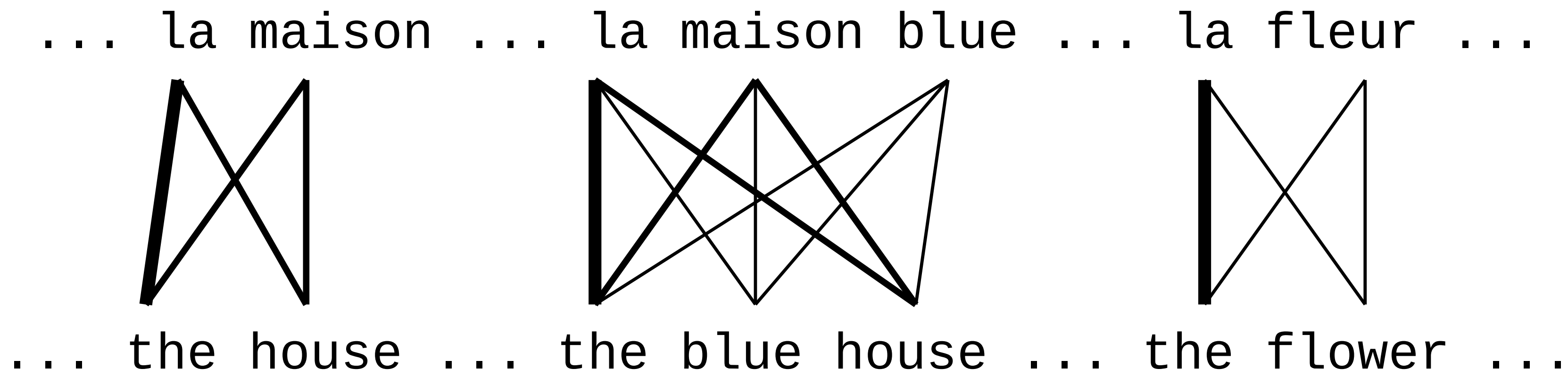
- Expectation Maximization (EM) in a nutshell
  1. initialize model parameters (e.g. uniform)
  2. assign probabilities to the missing data
  3. estimate model parameters from completed data
  4. iterate steps 2–3 until convergence

EM Algorithm



- Initial step: all alignments equally likely
- Model learns that, e.g., la is often aligned with the

EM Algorithm

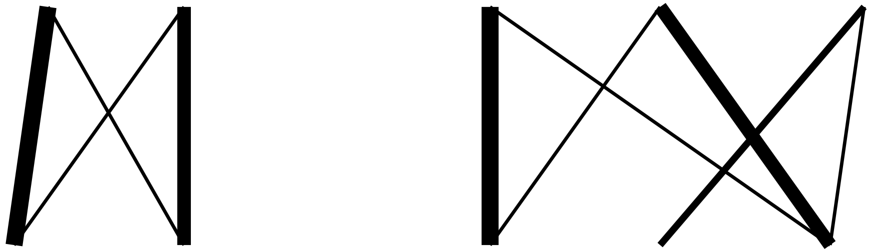


- After one iteration

- Alignments, e.g., between la and the are more likely

EM Algorithm

`... la maison ... la maison bleu ... la fleur ...`

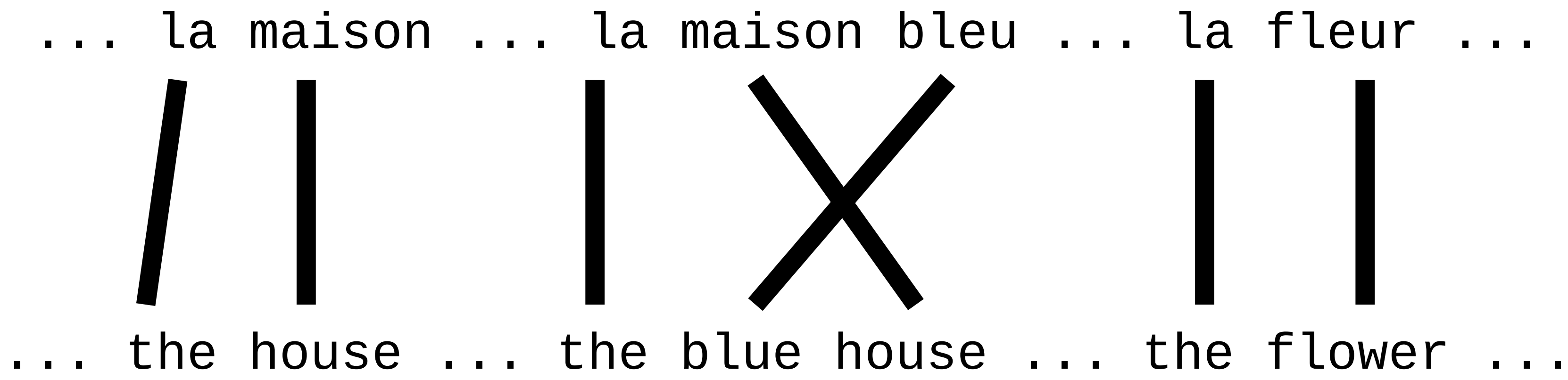


`... the house ... the blue house ... the flower ...`

- After another iteration

- It becomes apparent that alignments, e.g., between fleur and flower are more likely (pigeon hole principle)

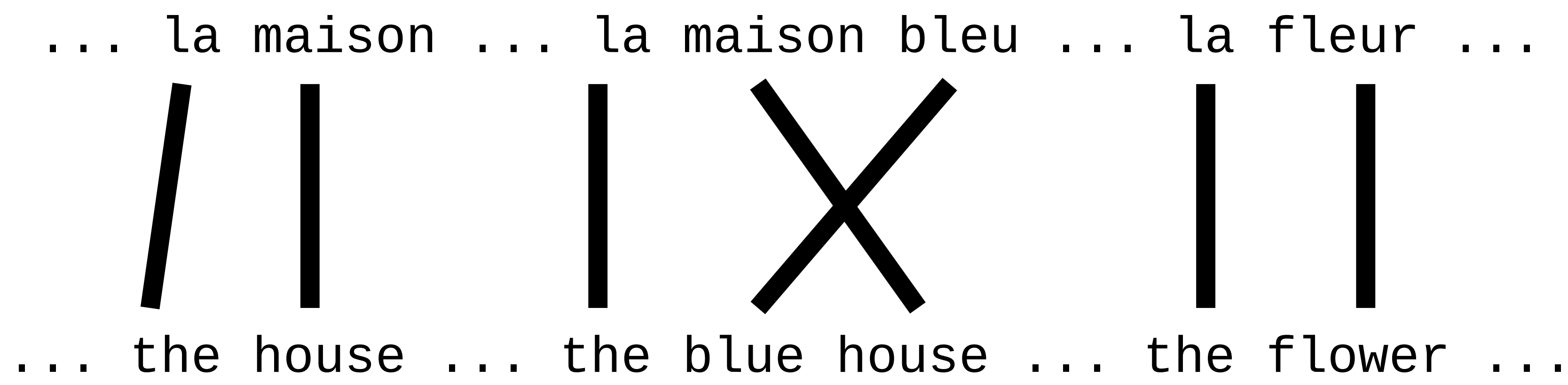
EM Algorithm



- Convergence

- Inherent hidden structure revealed by EM

EM Algorithm




```
p(la|the) = 0.453
p(le|the) = 0.334
p(maison|house) = 0.876
p(bleu|blue) = 0.563
...
```

- Parameter estimation from the aligned corpus

IBM Model 1 and EM

- EM Algorithm consists of two steps

- Expectation-Step: Apply model to the data
  - parts of the model are hidden (here: alignments)
  - using the model, assign probabilities to possible values

- Maximization-Step: Estimate model from data
  - take assign values as fact
  - collect counts (weighted by probabilities)
  - estimate model from counts

- Iterate these steps until convergence

IBM Model 1 and EM

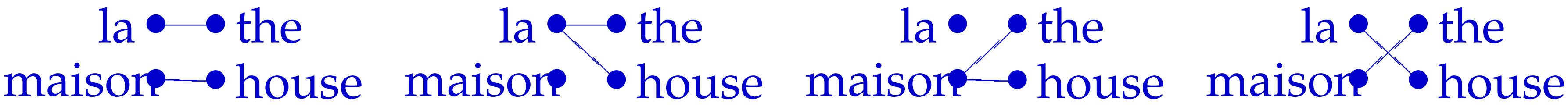
- We need to be able to compute:
  - Expectation-Step: probability of alignments
  - Maximization-Step: count collection

IBM Model 1 and EM

- **Probabilities**

$$p(\text{the}|\text{la}) = 0.7 \qquad p(\text{house}|\text{la}) = 0.05$$
$$p(\text{the}|\text{maison}) = 0.1 \qquad p(\text{house}|\text{maison}) = 0.8$$

- **Alignments**



$$p(\mathbf{e}, a|\mathbf{f}) = 0.56 \qquad p(\mathbf{e}, a|\mathbf{f}) = 0.035 \qquad p(\mathbf{e}, a|\mathbf{f}) = 0.08 \qquad p(\mathbf{e}, a|\mathbf{f}) = 0.005$$

$$p(a|\mathbf{e}, \mathbf{f}) = 0.824 \qquad p(a|\mathbf{e}, \mathbf{f}) = 0.052 \qquad p(a|\mathbf{e}, \mathbf{f}) = 0.118 \qquad p(a|\mathbf{e}, \mathbf{f}) = 0.007$$

- **Counts**

$$c(\text{the}|\text{la}) = 0.824 + 0.052 \qquad c(\text{house}|\text{la}) = 0.052 + 0.007$$
$$c(\text{the}|\text{maison}) = 0.118 + 0.007 \qquad c(\text{house}|\text{maison}) = 0.824 + 0.118$$

hmm model

Modeling Alignment

- IBM Model 1 uses alignments to identify conditioning context

- But: does not model alignment itself

- Is it better to start translating the 1st input word or 10th input word?

HMM Model

- Condition word movements on previous word

- HMM alignment model:

$$p(a(j)|a(j-1), l_f)$$

Decoding

- Input sentence

Eu quero ouvir uma apresentação muito interessante.

- Translation

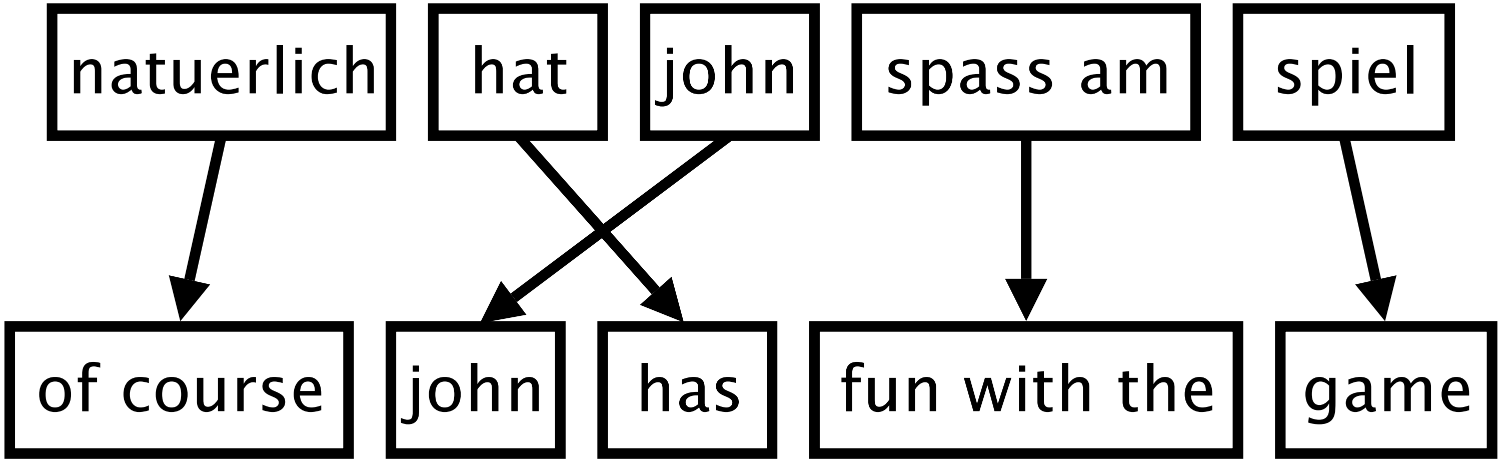
Input	Eu	quero		ouvir	uma
	|	/	\	|	|
Output	I	want	to	hear	a
Translation	$p(\text{I}\|\text{Eu})$	$p(\text{want}\|\text{quero})$	$p(\text{to}\|\text{quero})$	$p(\text{hear}\|\text{ouvir})$	$p(\text{a}\|\text{uma})$
Alignment	$p(1\|0,7)$	$p(2\|1,7)$	$p(2\|2,7)$	$p(3\|2,7)$	$p(4\|3,7)$
Language Model	$p(\text{I}\|\text{START})$	$p(\text{want}\|\text{I})$	$p(\text{to}\|\text{want})$	$p(\text{hear}\|\text{to})$	$p(\text{a}\|\text{hear})$

phrase-based model

Motivation

- Word-Based Models translate *words* as atomic units
- Phrase-Based Models translate *phrases* as atomic units
- Advantages:
  - many-to-many translation can handle non-compositional phrases
  - use of local context in translation
  - the more data, the longer phrases can be learned
- ”Standard Model”, used by Google Translate and others

Phrase-Based Model



- Foreign input is segmented in phrases
- Each phrase is translated into English
- Phrases are reordered

Phrase Translation Table

- Main knowledge source: table with phrase translations and their probabilities
- Example: phrase translations for natuerlich

Translation	Probability $\phi(\bar{e}|\bar{f})$
of course	0.5
naturally	0.3
of course ,	0.15
, of course ,	0.05

Real Example

- Phrase translations for den Vorschlag learned from the Europarl corpus:

English	$\phi(\bar{e}\|\bar{f})$	English	$\phi(\bar{e}\|\bar{f})$
the proposal	0.6227	the suggestions	0.0114
's proposal	0.1068	the proposed	0.0114
a proposal	0.0341	the motion	0.0091
the idea	0.0250	the idea of	0.0091
this proposal	0.0227	the proposal ,	0.0068
proposal	0.0205	its proposal	0.0068
of the proposal	0.0159	it	0.0068
the proposals	0.0159	...	...

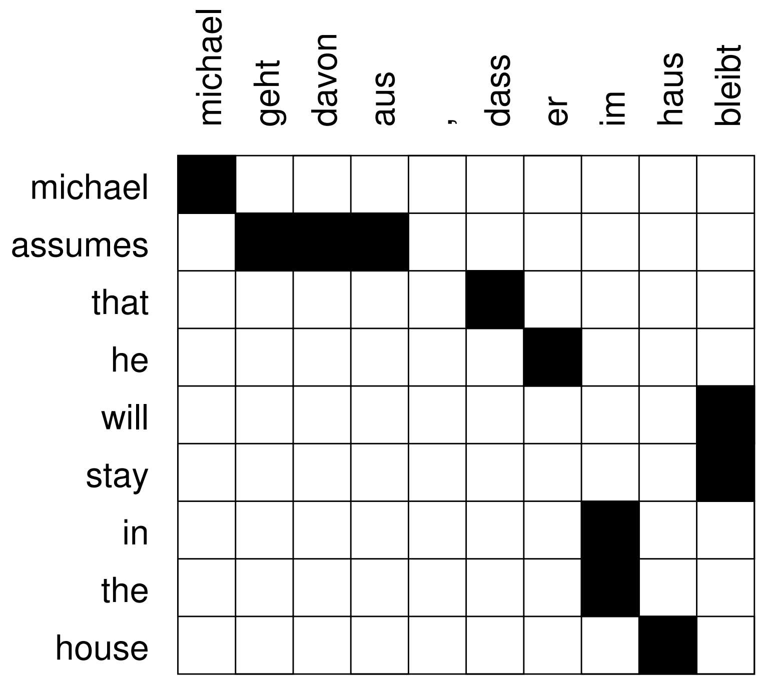
  - lexical variation (proposal vs suggestions)
  - morphological variation (proposal vs proposals)
  - included function words (the, a, ...)
  - noise (it)

Learning a Phrase Translation Table

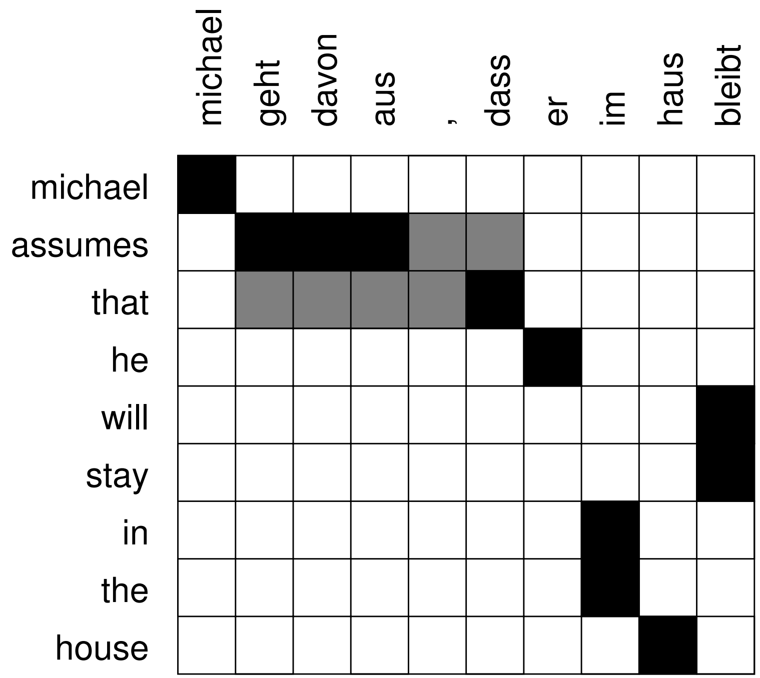
- Task: learn the model from a parallel corpus

- Three stages:
  - word alignment: using IBM models or other method
  - extraction of phrase pairs
  - scoring phrase pairs

Word Alignment



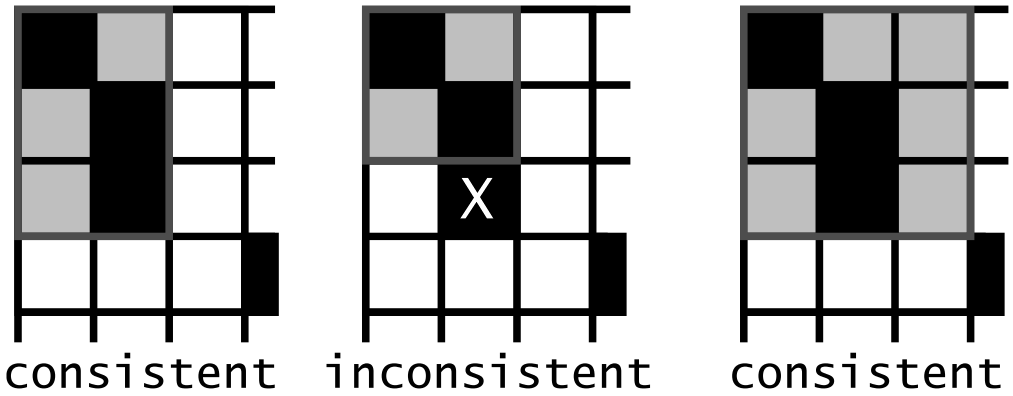
Extracting Phrase Pairs



extract phrase pair consistent with word alignment:

assumes that / geht davon aus , dass

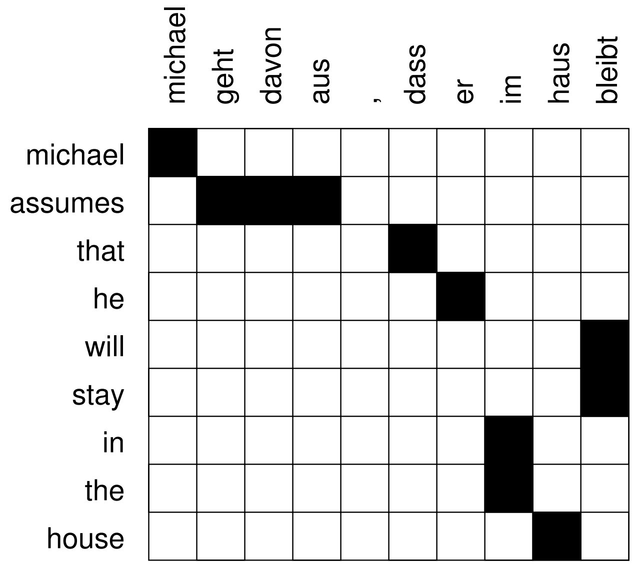
Consistent



ok	**violated**	**ok**
	one alignment point outside	unaligned word is fine

All words of the phrase pair have to align to each other.

Phrase Pair Extraction

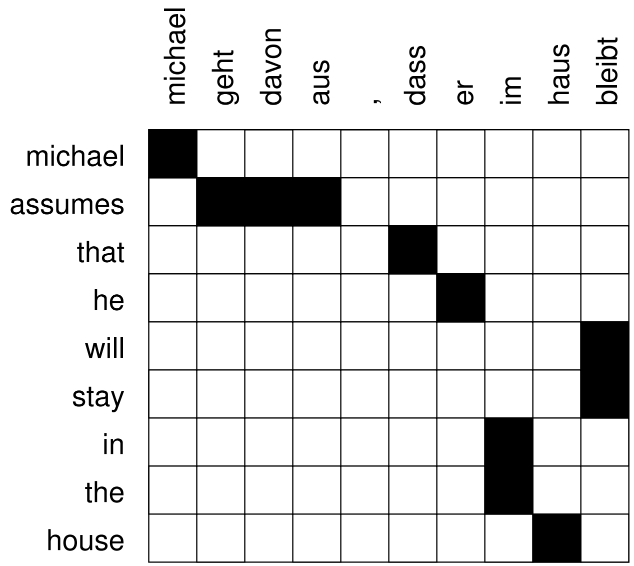


Smallest phrase pairs:

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

unaligned words (here: German comma) lead to multiple translations

Larger Phrase Pairs



michael assumes — michael geht davon aus / michael geht davon aus ,
assumes that — geht davon aus , dass ; assumes that he — geht davon aus , dass er
that he — dass er / , dass er ; in the house — im haus
michael assumes that — michael geht davon aus , dass
michael assumes that he — michael geht davon aus , dass er
michael assumes that he will stay in the house — michael geht davon aus , dass er im haus bleibt
assumes that he will stay in the house — geht davon aus , dass er im haus bleibt
that he will stay in the house — dass er im haus bleibt ; dass er im haus bleibt ,
he will stay in the house — er im haus bleibt ; will stay in the house — im haus bleibt

Scoring Phrase Translations

- Phrase pair extraction: collect all phrase pairs from the data
- Phrase pair scoring: assign probabilities to phrase translations
- Score by relative frequency:

$$\phi(\bar{f}|\bar{e}) = \frac{\text{count}(\bar{e}, \bar{f})}{\sum_{\bar{f}_i} \text{count}(\bar{e}, \bar{f}_i)}$$

Decoding

- We have a mathematical model for translation

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

- Task of decoding: find the translation $\mathbf{e}_{\text{best}}$ with highest probability

$$\mathbf{e}_{\text{best}} = \text{argmax}_{\mathbf{e}}\ p(\mathbf{e}|\mathbf{f})$$

- Two types of error
  - the most probable translation is bad $\rightarrow$ fix the model
  - search does not find the most probably translation $\rightarrow$ fix the search

- Decoding is evaluated by search error, not quality of translations (although these are often correlated)

Translation Process

- Task: translate this sentence from German into English

er	**geht**	**ja**	**nicht**	**nach**	**hause**

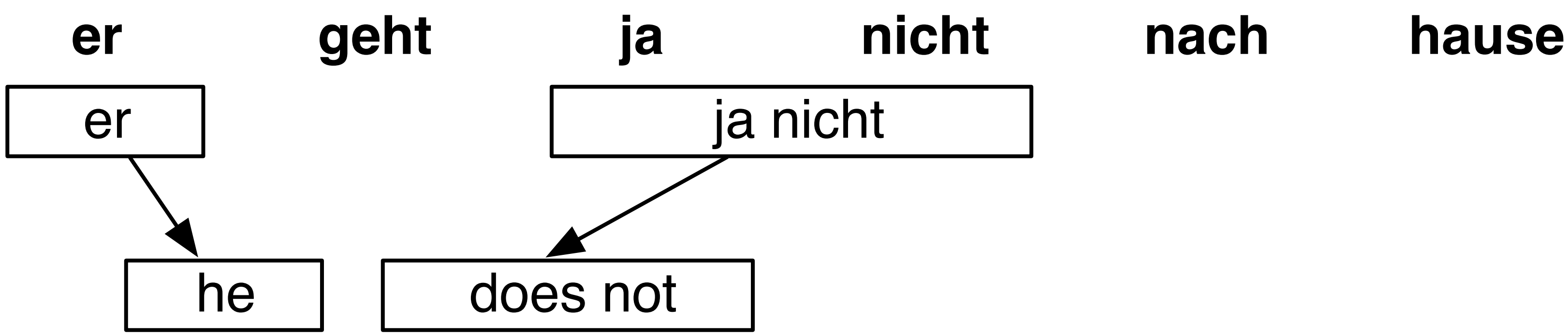
Translation Process

- Task: translate this sentence from German into English

- Pick phrase in input, translate

Translation Process

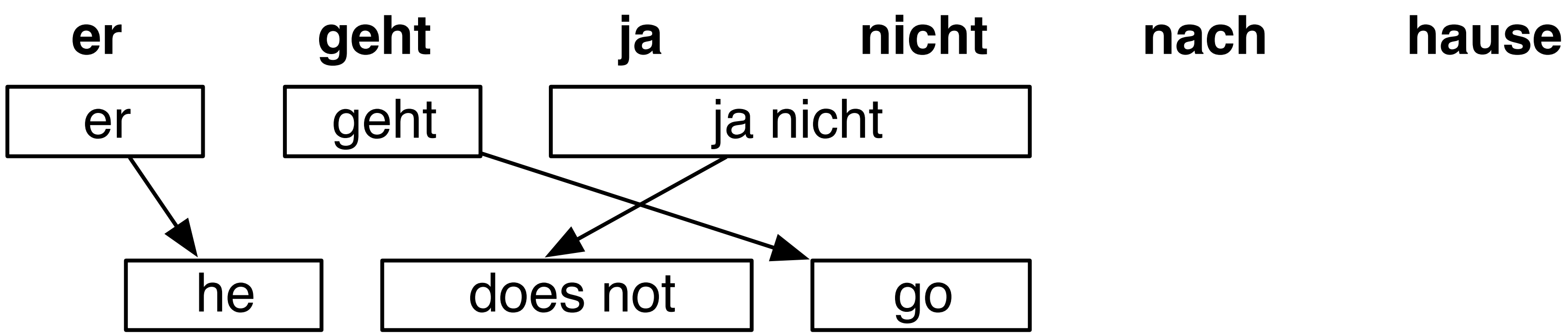
- Task: translate this sentence from German into English



- Pick phrase in input, translate
  - it is allowed to pick words out of sequence reordering
  - phrases may have multiple words: many-to-many translation

Translation Process

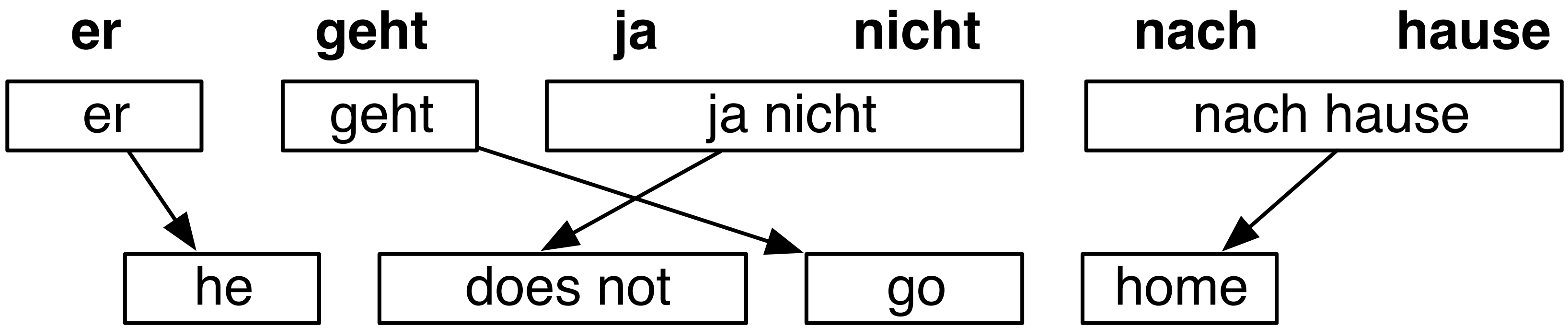
- Task: translate this sentence from German into English



- Pick phrase in input, translate

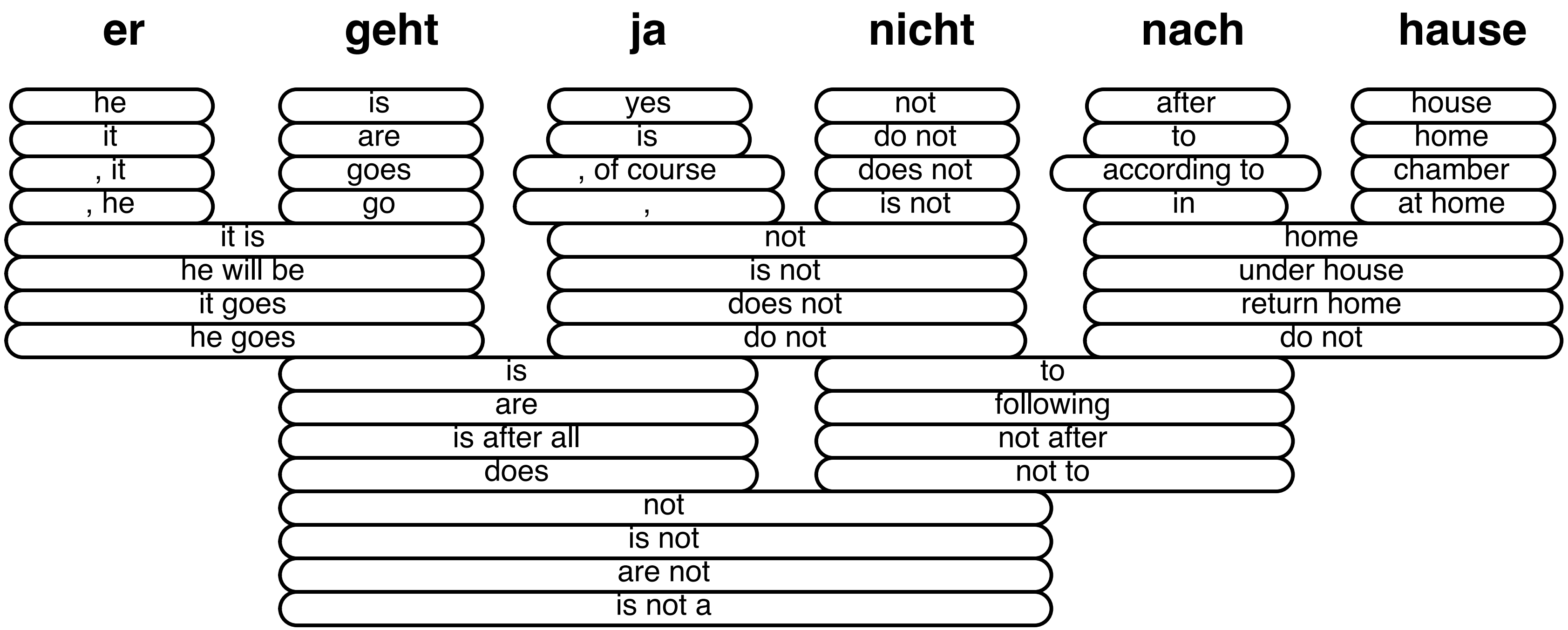
Translation Process

- Task: translate this sentence from German into English



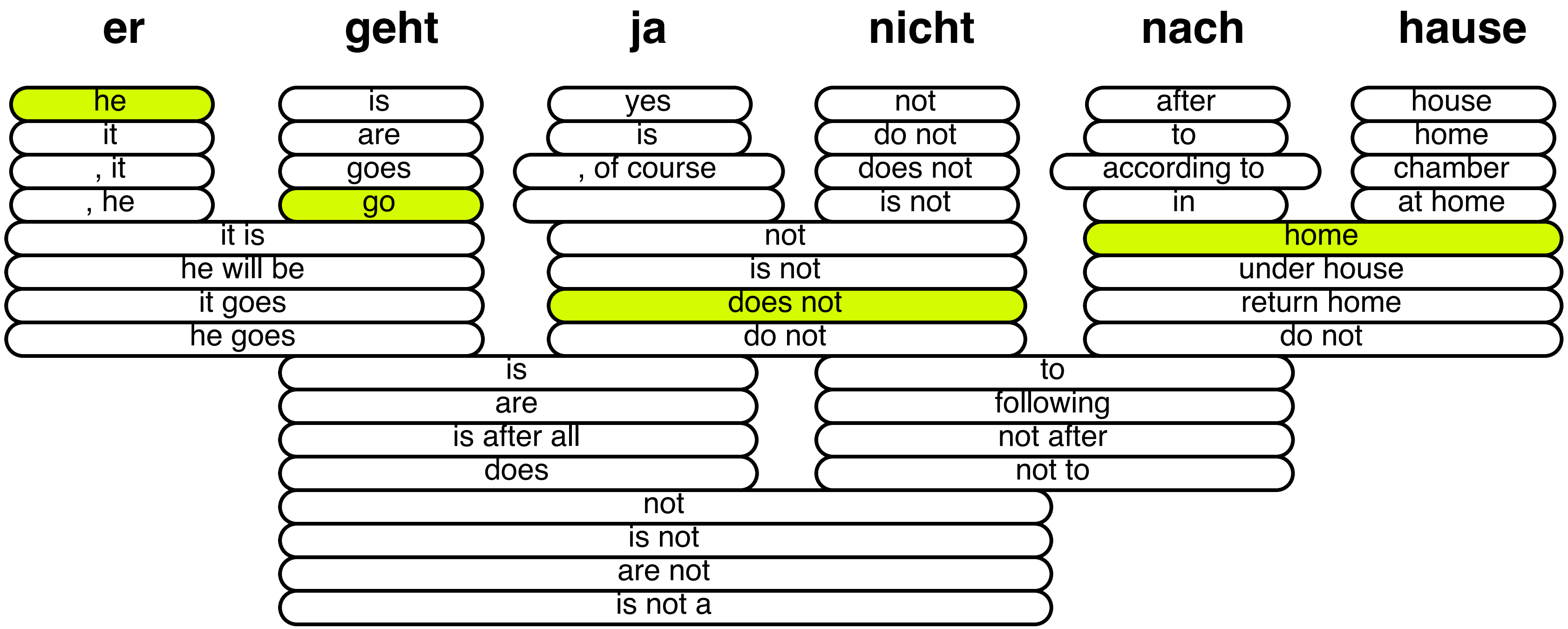
- Pick phrase in input, translate

Translation Options



- Many translation options to choose from
  - in Europarl phrase table: 2727 matching phrase pairs for this sentence
  - by pruning to the top 20 per phrase, 202 translation options remain

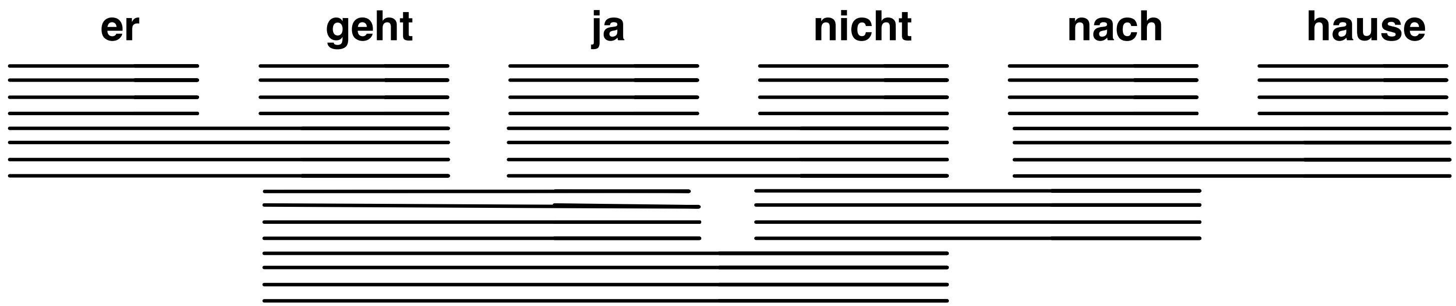
Translation Options



- The machine translation decoder does not know the right answer
  - picking the right translation options
  - arranging them in the right order

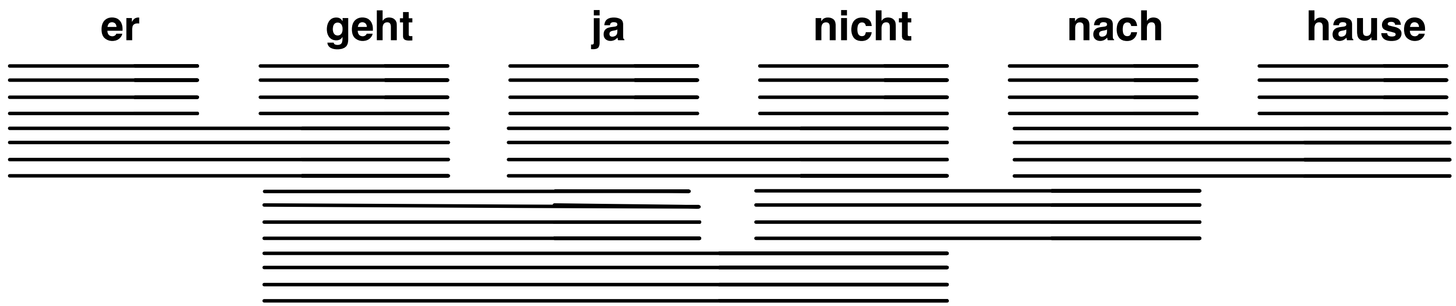
$\rightarrow$ Search problem solved by heuristic beam search

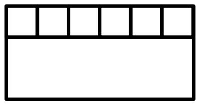
Decoding: Precompute Translation Options



consult phrase translation table for all input phrases

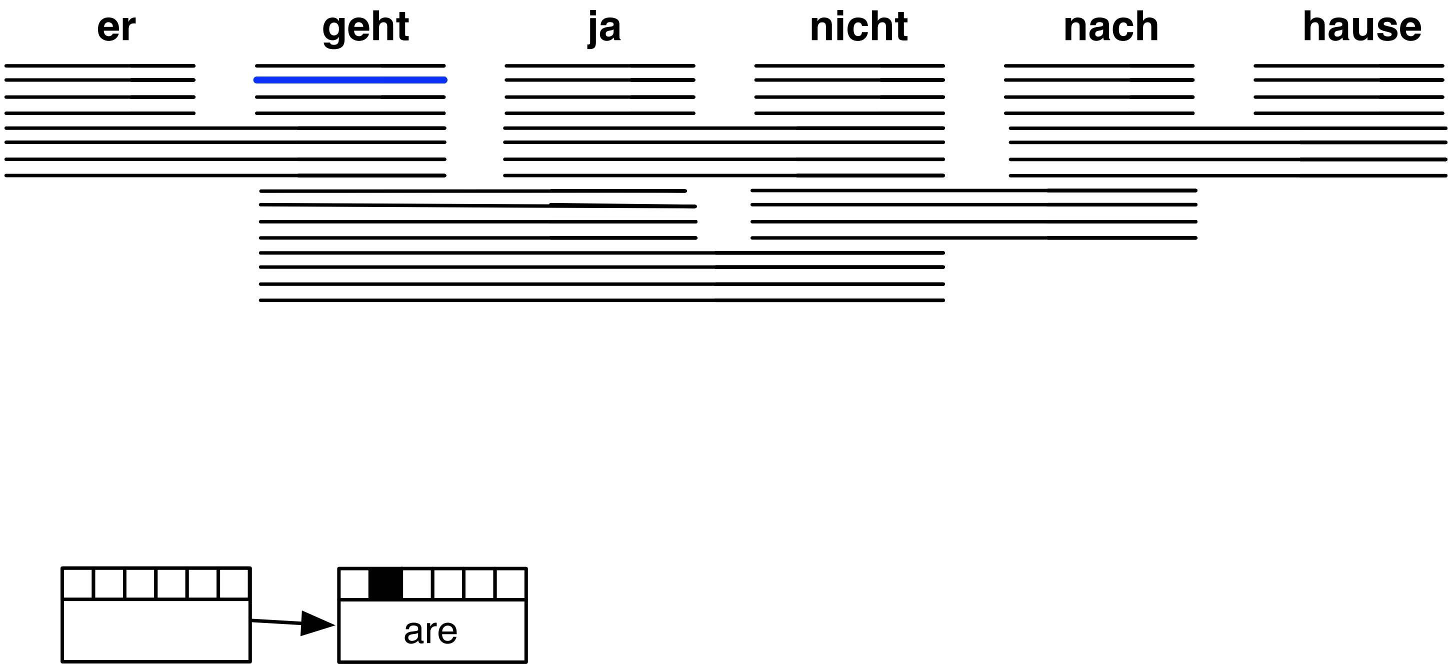
Decoding: Start with Initial Hypothesis





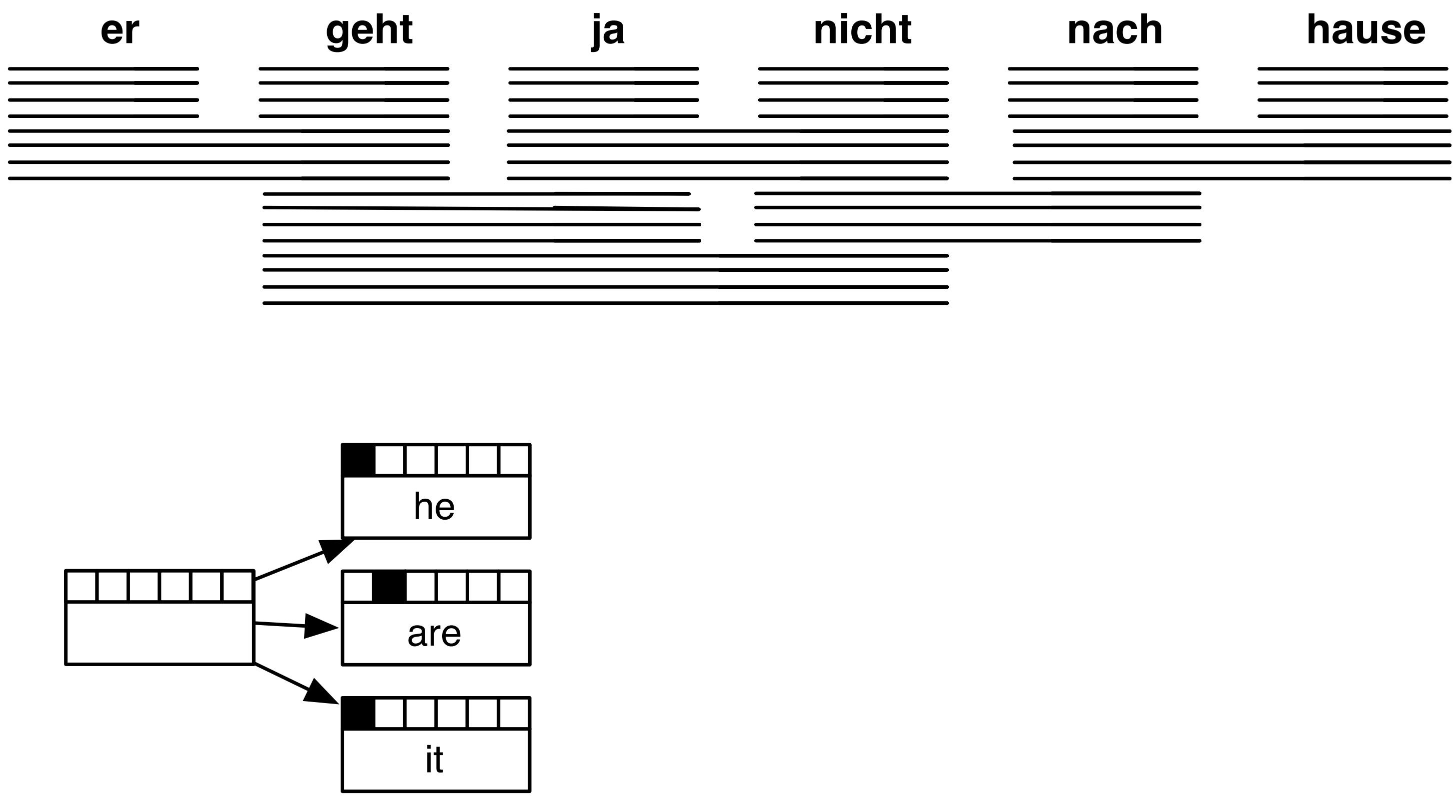
initial hypothesis: no input words covered, no output produced

Decoding: Hypothesis Expansion



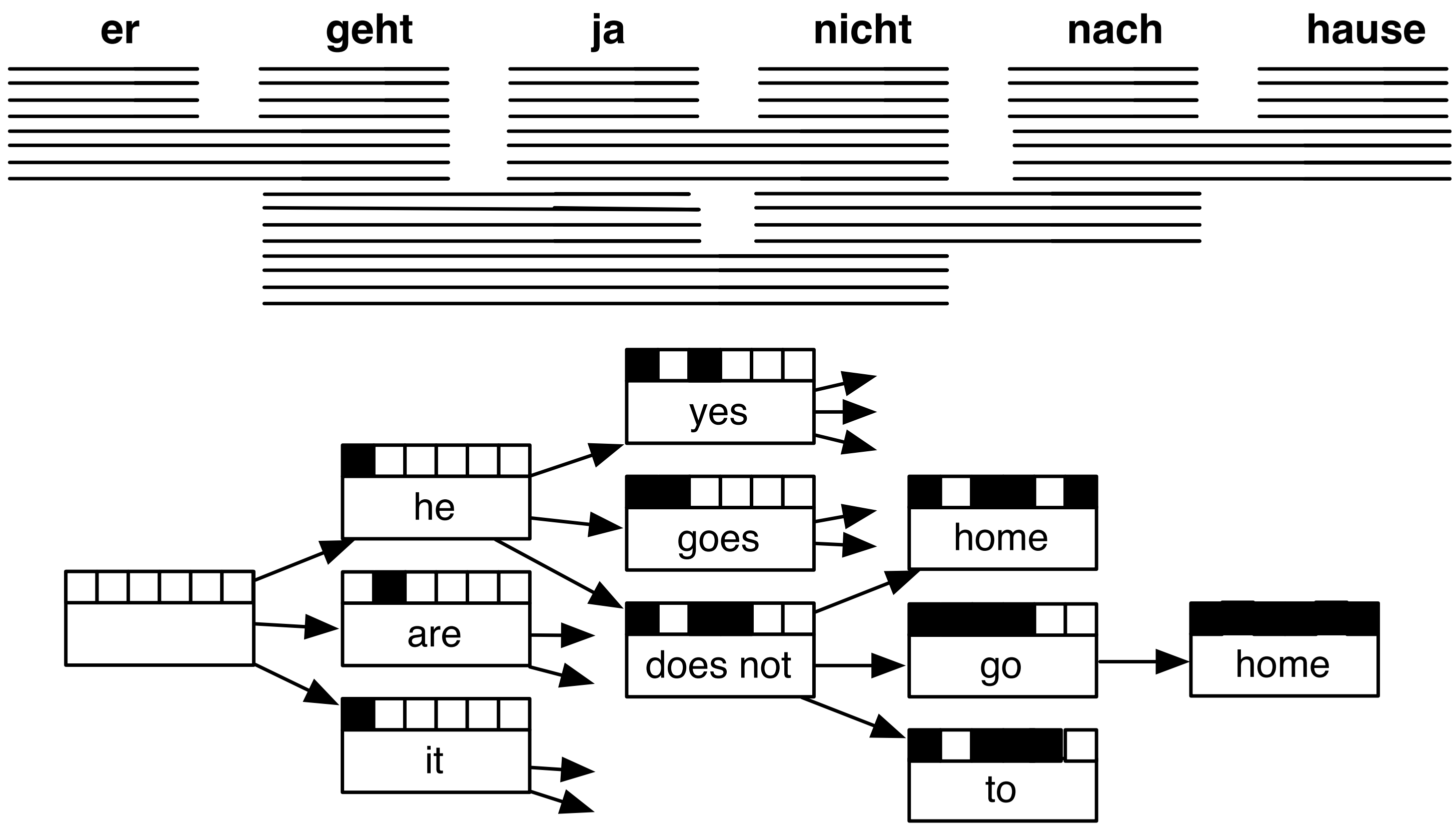
pick any translation option, create new hypothesis

Decoding: Hypothesis Expansion



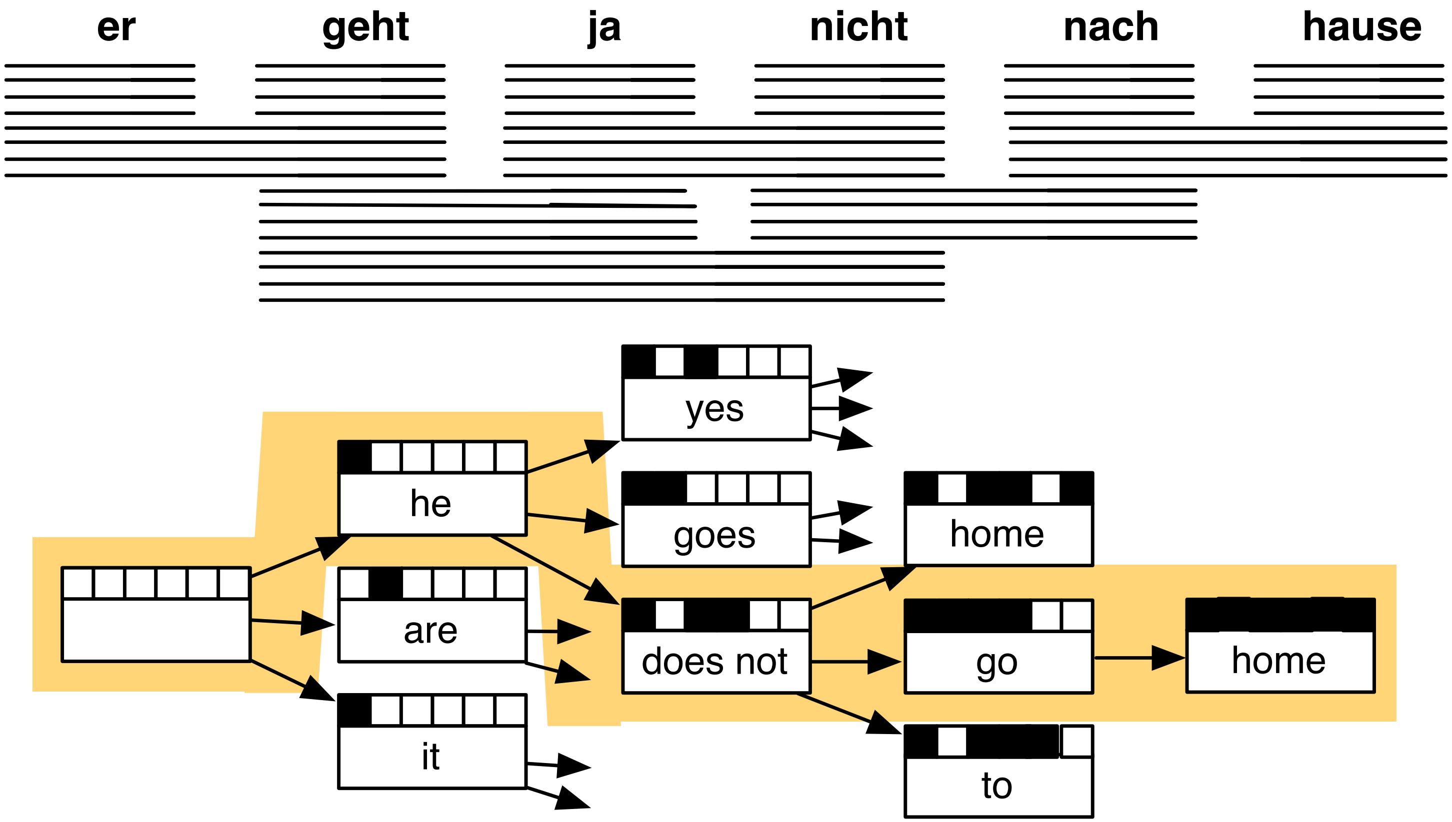
create hypotheses for all other translation options

Decoding: Hypothesis Expansion



also create hypotheses from created partial hypothesis

Decoding: Find Best Path



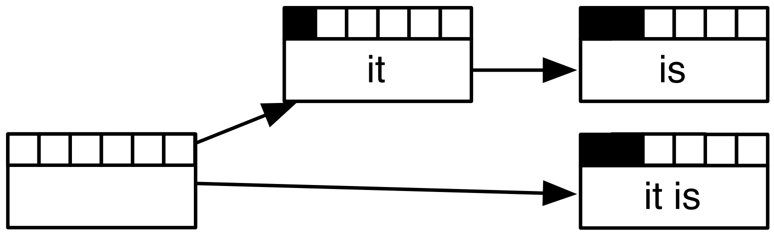
backtrack from highest scoring complete hypothesis

Computational Complexity

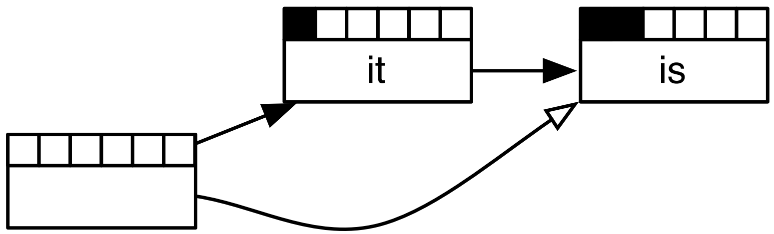
- The suggested process creates exponential number of hypothesis
- Machine translation decoding is NP-complete
- Reduction of search space:
  - recombination (risk-free)
  - pruning (risky)

Recombination

- Two hypothesis paths lead to two matching hypotheses
  - same number of foreign words translated
  - same English words in the output
  - different scores

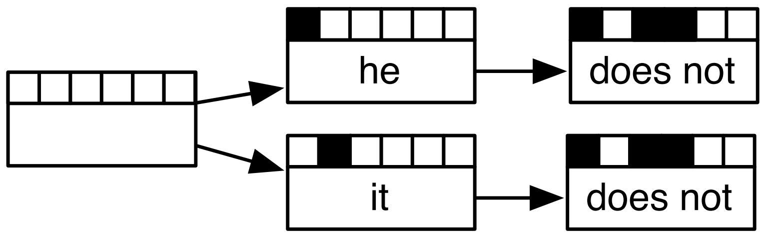


- Worse hypothesis is dropped

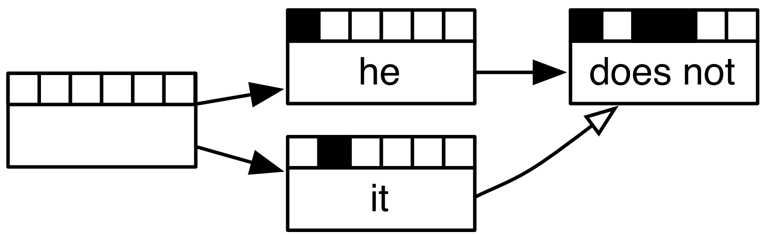


Recombination

- Two hypothesis paths lead to hypotheses indistinguishable in subsequent search
  - same number of foreign words translated
  - same last two English words in output (assuming trigram language model)
  - same last foreign word translated
  - different scores



- Worse hypothesis is dropped

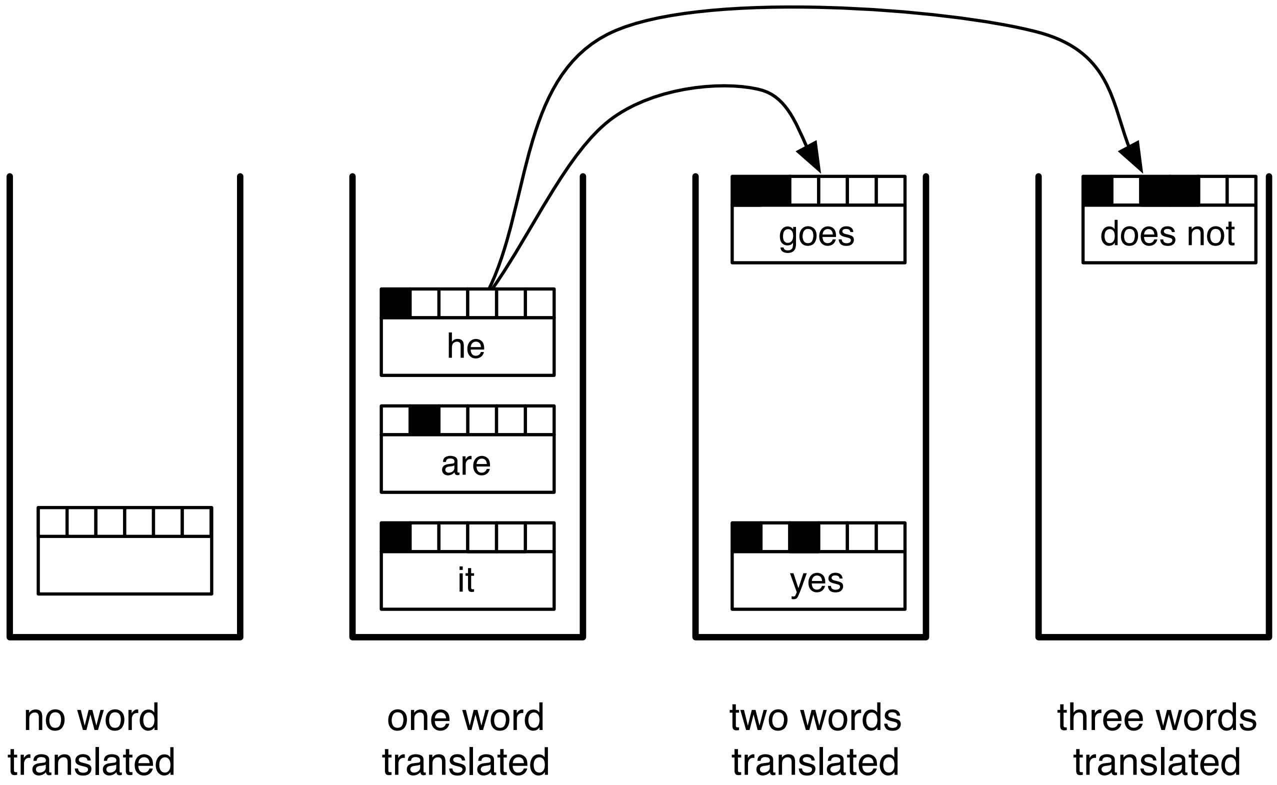


Pruning

- Recombination reduces search space, but not enough
  (we still have a NP complete problem on our hands)

- Pruning: remove bad hypotheses early
  - put comparable hypothesis into stacks
    (hypotheses that have translated same number of input words)
  - limit number of hypotheses in each stack

Stacks



- Hypothesis expansion in a stack decoder
  - translation option is applied to hypothesis
  - new hypothesis is dropped into a stack further down

Stack Decoding Algorithm


```
place empty hypothesis into stack 0
for all stacks 0...n − 1 do
    for all hypotheses in stack do
        for all translation options do
            if applicable then
                create new hypothesis
                place in stack
                recombine with existing hypothesis if possible
                prune stack if too big
            end if
        end for
    end for
end for
```

Pruning

- Pruning strategies
  - histogram pruning: keep at most k hypotheses in each stack
  - stack pruning: keep hypothesis with score $\alpha \times$ best score ($\alpha < 1$)

- Computational time complexity of decoding with histogram pruning

$$O(\text{max stack size} \times \text{translation options} \times \text{sentence length})$$

- Number of translation options is linear with sentence length, hence:

$$O(\text{max stack size} \times \text{sentence length}^2)$$

- Quadratic complexity

operation sequence model

A Critique: Phrase Segmentation is Arbitrary

- If multiple segmentations possible - why chose one over the other?

  $\boxed{\text{spass am}}$ $\boxed{\text{spiel}}$ vs. $\boxed{\text{spass}}$ $\boxed{\text{am spiel}}$

- When choose larger phrase pairs or multiple shorter phrase pairs?

  $\boxed{\text{spass am}}$ $\boxed{\text{spiel}}$ vs. $\boxed{\text{spass}}$ $\boxed{\text{am}}$ $\boxed{\text{spiel}}$ vs. $\boxed{\text{spass am spiel}}$

- None of this has been properly addressed

A Critique: Strong Independence Assumptions

- Lexical context considered only within phrase pairs

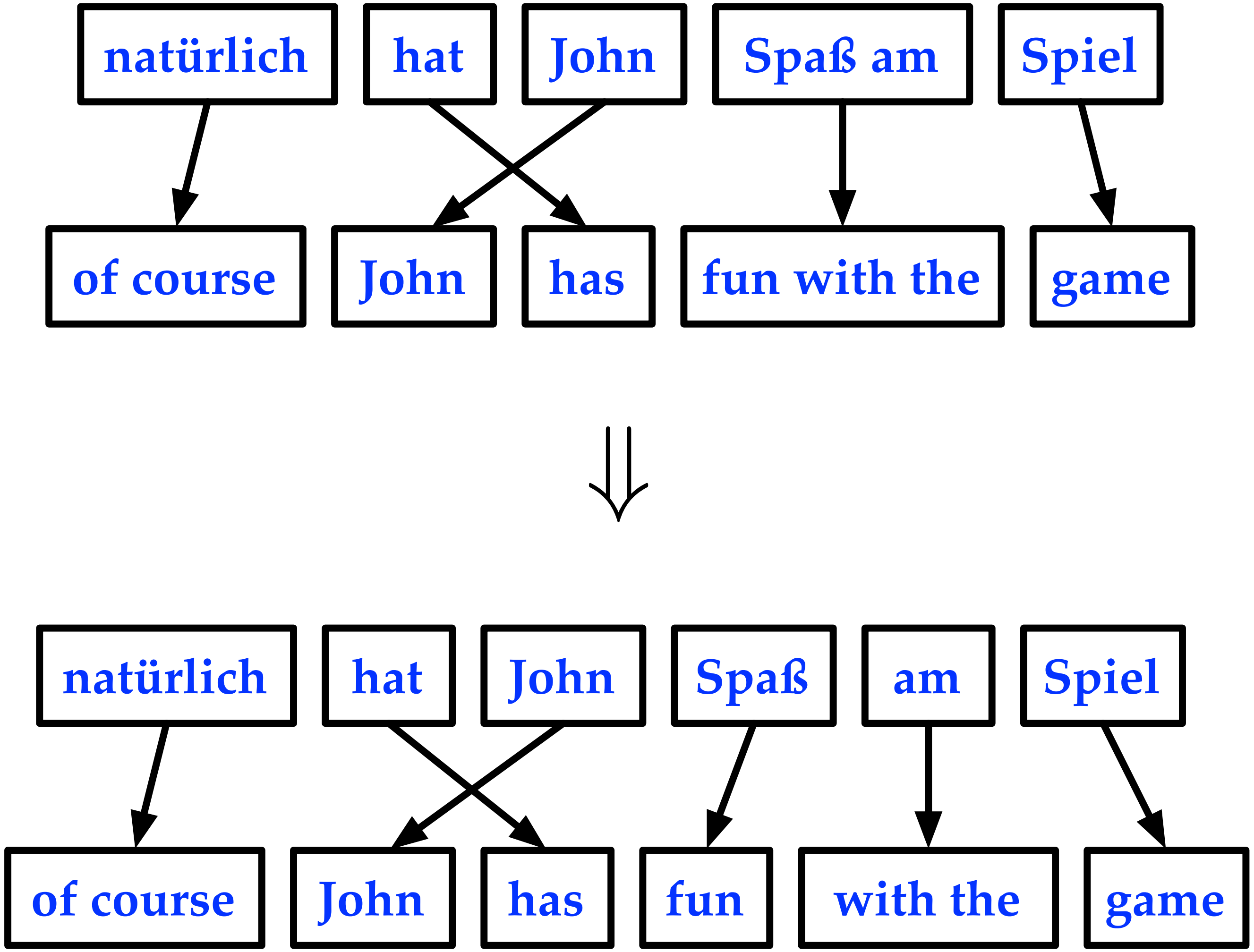
$\boxed{\text{spass am}} \rightarrow \boxed{\text{fun with}}$

- No context considered between phrase pairs

$? \boxed{\text{spass am}} ? \rightarrow ? \boxed{\text{fun with}} ?$

- Some phrasal context considered in lexicalized reordering model
  ... but not based on the identity of neighboring phrases

Segmentation? Minimal Phrase Pairs



Independence?
Consider Sequence of Operations

o_1	Generate(natürlich, of course)	natürlich ↓ of course
o_2 o_3	Insert Gap Generate (John, John)	natürlich ↓ $\boxed{}$ John of course John
o_4 o_5	Jump Back (1) Generate (hat, has)	natürlich hat ↓ John of course John has
o_6	Jump Forward	natürlich hat John ↓ of course John has
o_7	Generate(natürlich, of course)	natürlich hat John Spaß ↓ of course John has fun
o_8 o_9	Generate(am, with) GenerateTargetOnly(the)	natürlich hat John Spaß am ↓ of course John has fun with the
o_{10}	Generate(Spiel, game)	natürlich hat John Spaß am Spiel ↓ of course John has fun with the game

Operation Sequence Model

- Operations
  - generate (phrase translation)
  - generate target only
  - generate source only
  - insert gap
  - jump back
  - jump forward

- N-gram sequence model over operations, e.g., 5-gram model:

$$p(o_1)\ p(o_2|o_1)\ p(o_3|o_1,o_2)\ ...\ p(o_{10}|o_6,o_7,o_8,o_9)$$

In Practice

- Operation Sequence Model used as additional feature function

- Significant improvements over phrase-based baseline

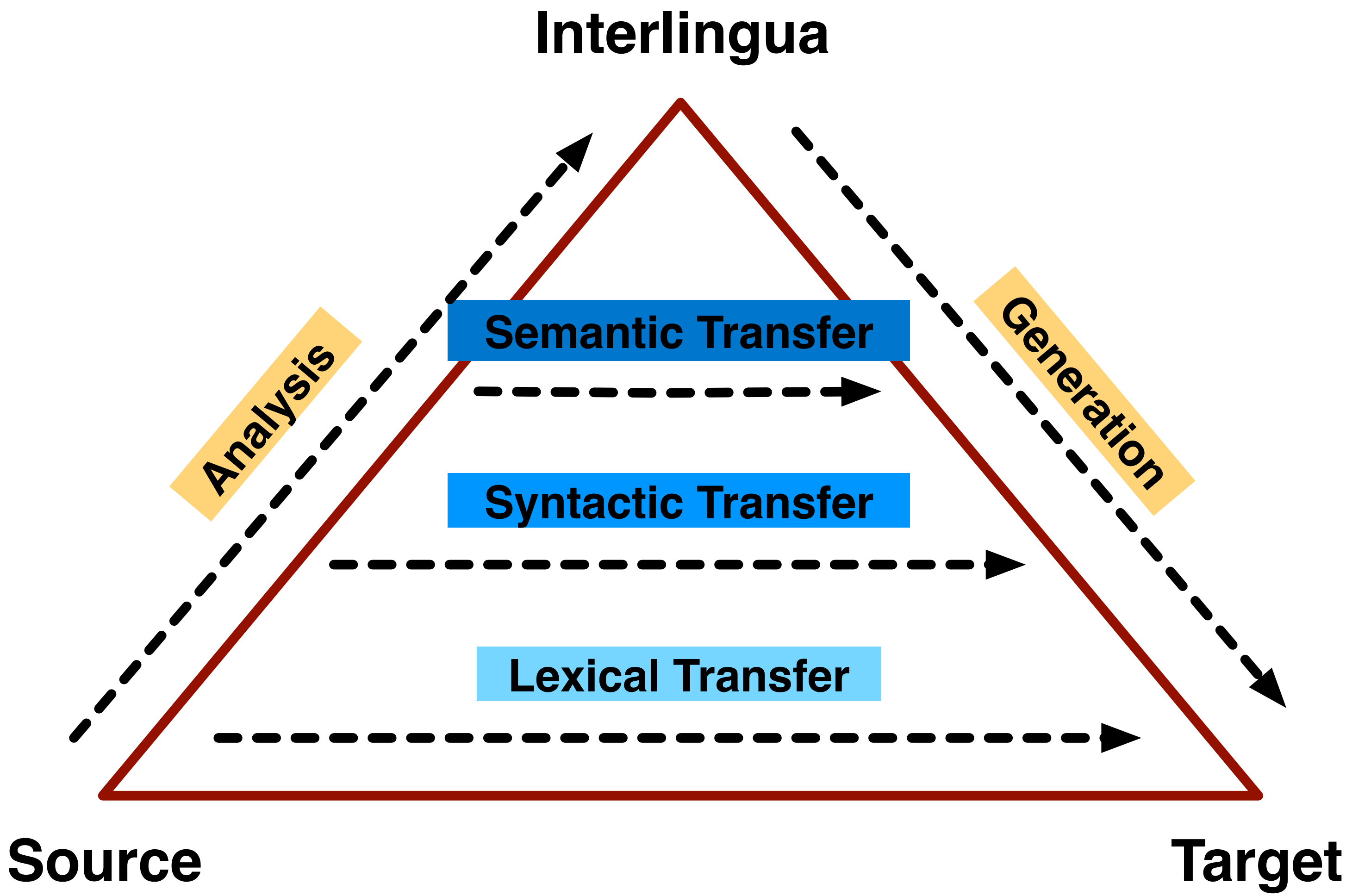
→ State-of-the-art systems include such a model

syntax

Sequence Model — Really?

- Different languages have different word order
- Language is recursive $\rightarrow$ tree formalisms
- Need to translate *meaning,* not *words*

A Vision

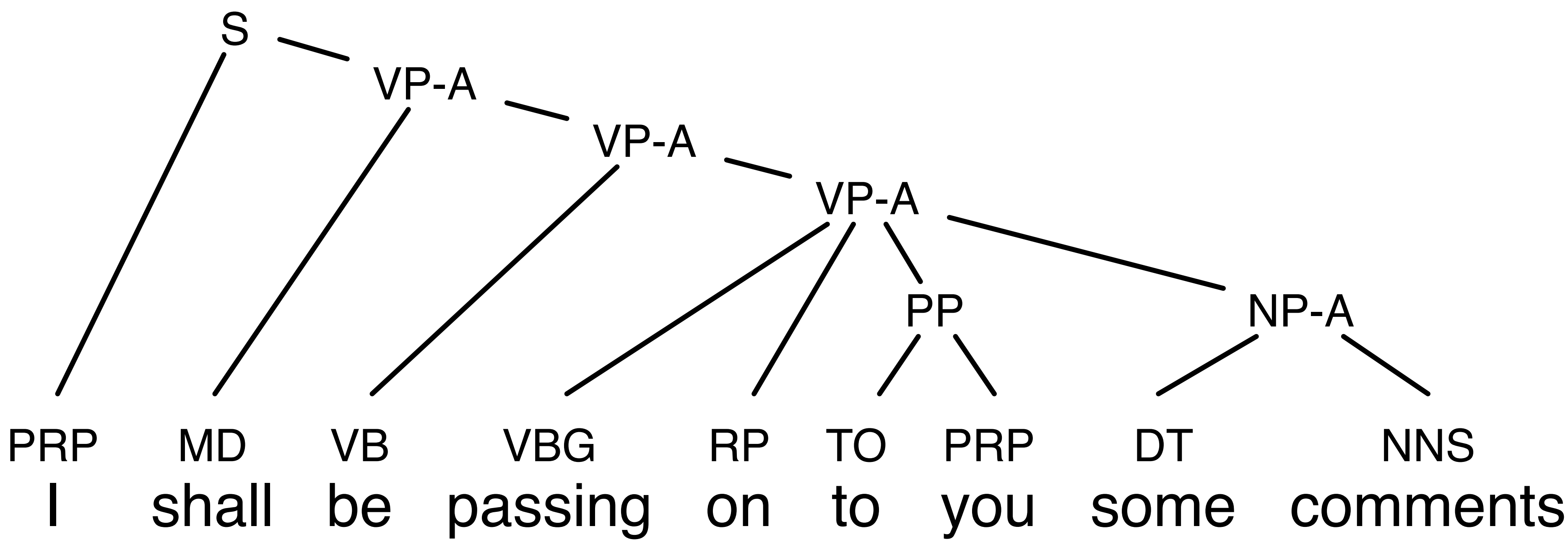

Interlingua
Analysis
Generation
Semantic Transfer
Syntactic Transfer
Lexical Transfer
Source
Target

Phrase Structure Grammar

- Phrase structure
  - noun phrases: the big man, a house, ...
  - prepositional phrases: at 5 o'clock, in Edinburgh, ...
  - verb phrases: going out of business, eat chicken, ...
  - adjective phrases, ...

- Context-free Grammars (CFG)
  - non-terminal symbols: phrase structure labels, part-of-speech tags
  - terminal symbols: words
  - production rules: NT $\rightarrow$ [NT,T]+
    example: NP $\rightarrow$ DET NN

Phrase Structure Grammar



Phrase structure grammar tree for an English sentence
(as produced Collins' parser)

Synchronous Phrase Structure Grammar

- English rule

$$\text{NP} \rightarrow \text{DET JJ NN}$$

- French rule

$$\text{NP} \rightarrow \text{DET NN JJ}$$

- Synchronous rule (indices indicate alignment):

$$\text{NP} \rightarrow \text{DET}_1 \ \text{NN}_2 \ \text{JJ}_3 \ \big| \ \text{DET}_1 \ \text{JJ}_3 \ \text{NN}_2$$

Synchronous Grammar Rules

- Nonterminal rules

$$\text{NP} \rightarrow \text{DET}_1\ \text{NN}_2\ \text{JJ}_3 \mid \text{DET}_1\ \text{JJ}_3\ \text{NN}_2$$

- Terminal rules

$$\text{N} \rightarrow \text{maison} \mid \text{house}$$

$$\text{NP} \rightarrow \text{la maison bleue} \mid \text{the blue house}$$

- Mixed rules

$$\text{NP} \rightarrow \text{la maison JJ}_1 \mid \text{the JJ}_1\ \text{house}$$

Tree-Based Translation Model

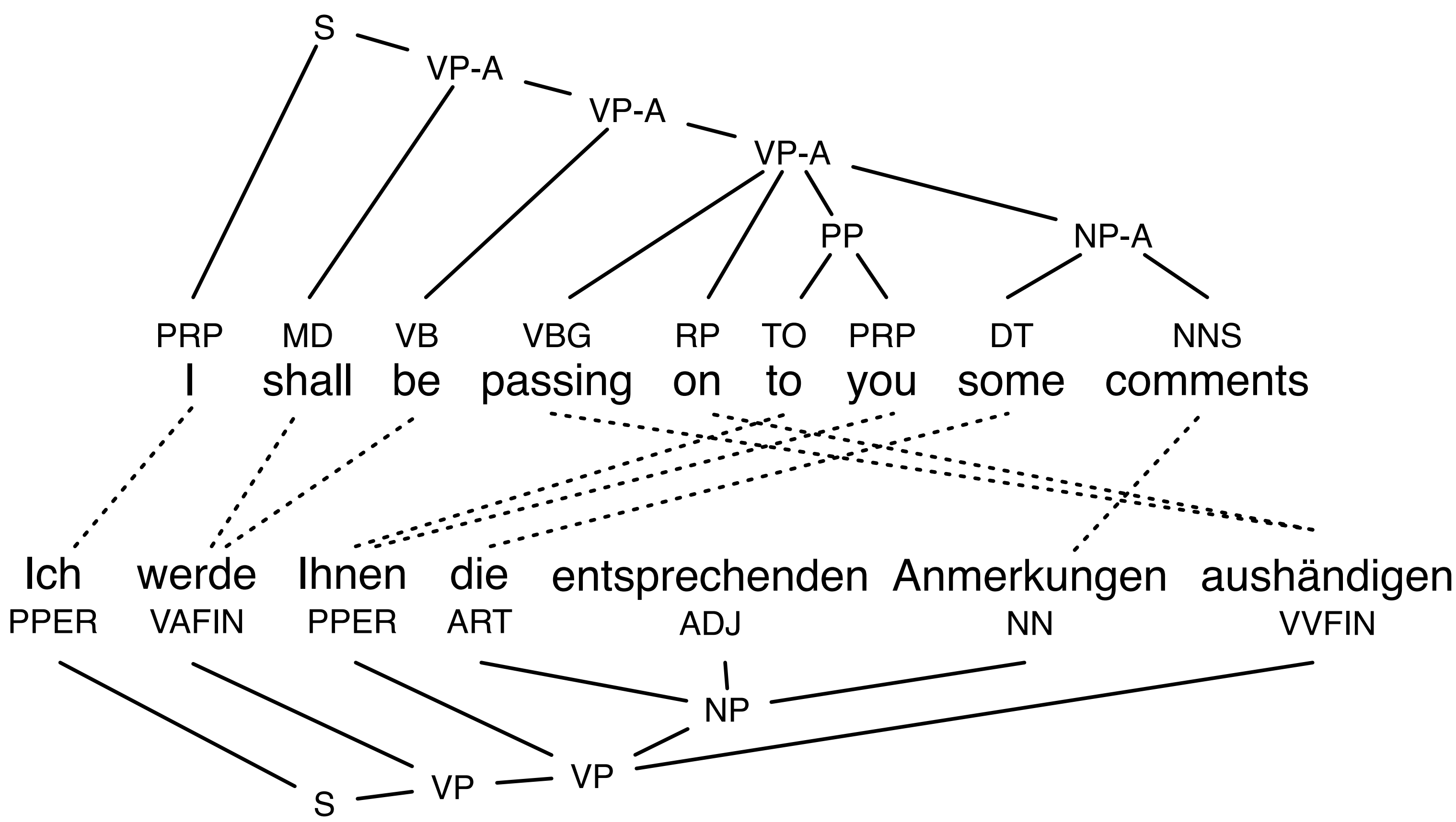
- Translation by parsing
  - synchronous grammar has to parse entire input sentence
  - output tree is generated at the same time
  - process is broken up into a number of rule applications

- Translation probability

$$\text{SCORE}(\text{TREE}, \text{E}, \text{F}) = \prod_i \text{RULE}_i$$

- Many ways to assign probabilities to rules

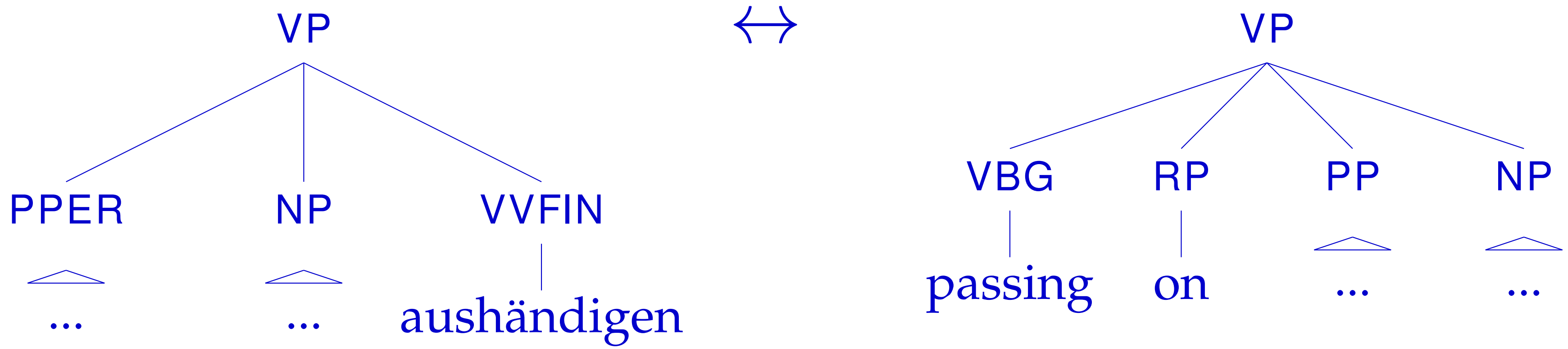
Aligned Tree Pair



Phrase structure grammar trees with word alignment
(German–English sentence pair.)

Reordering Rule

- Subtree alignment

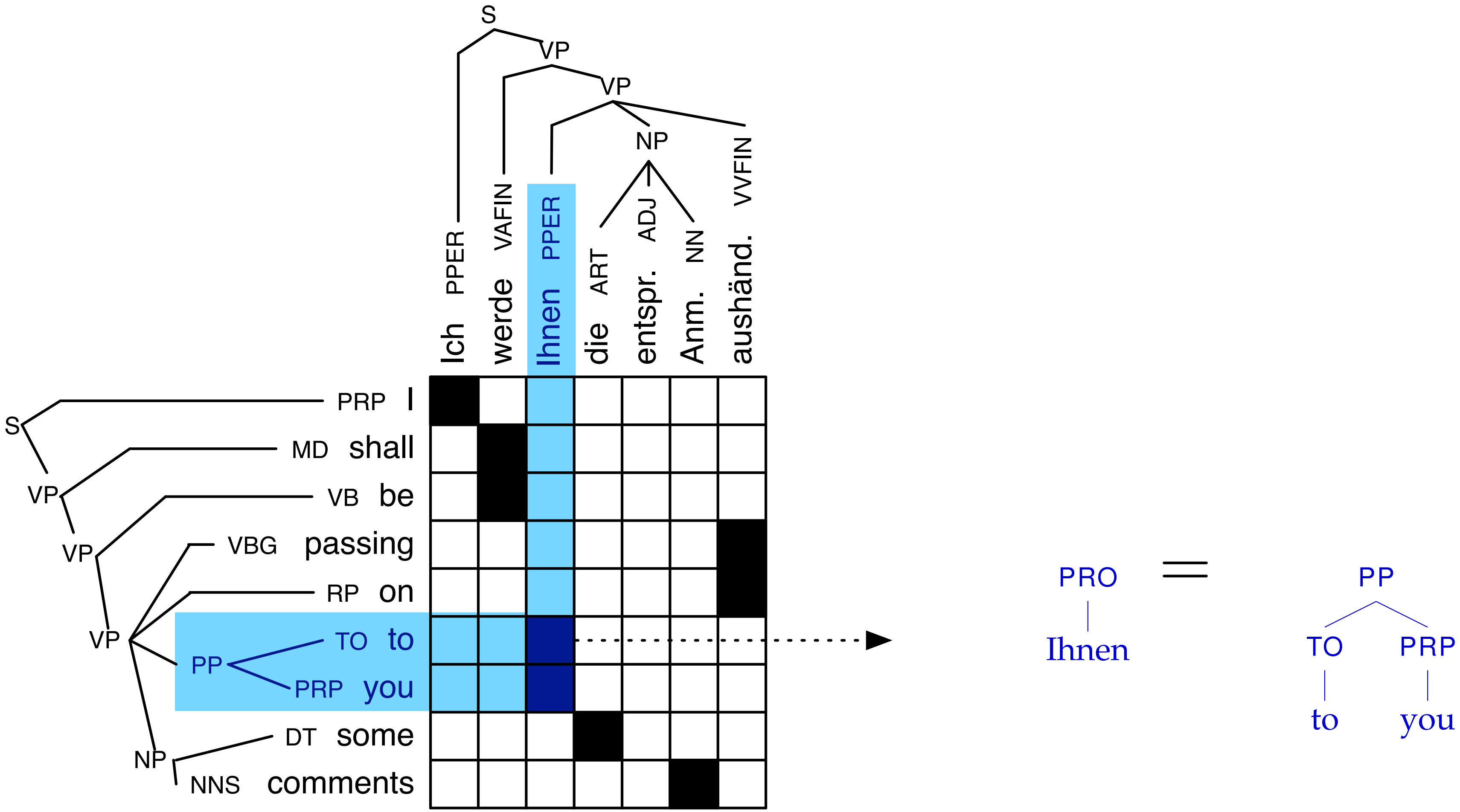


- Synchronous grammar rule

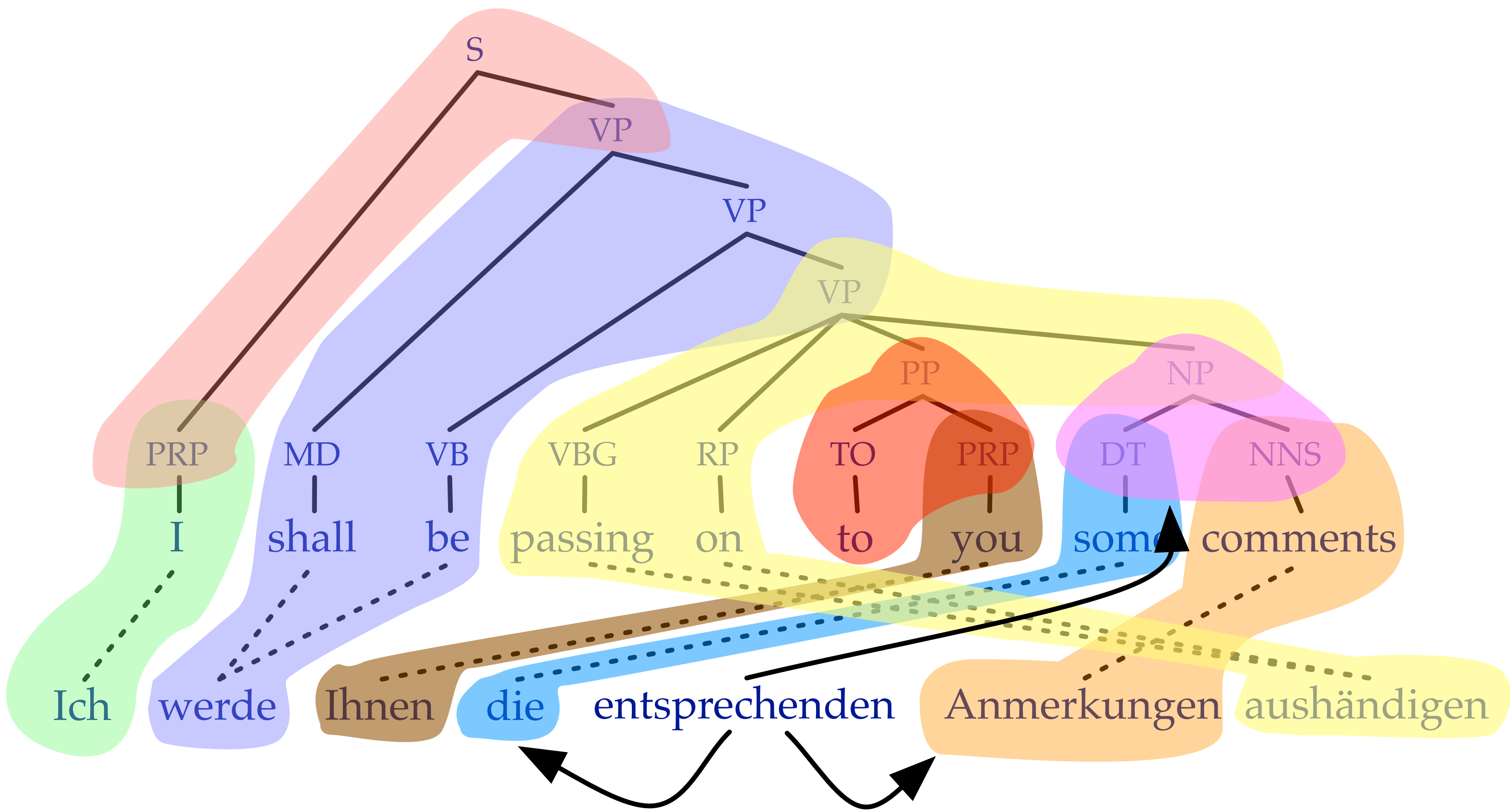
$$\text{VP} \rightarrow \text{PPER}_1 \ \text{NP}_2 \ \text{aushändigen} \mid \text{passing on PP}_1 \ \text{NP}_2$$

- Note:
  - one word aushändigen mapped to two words passing on ok
  - but: fully non-terminal rule not possible (one-to-one mapping constraint for nonterminals)

Learning Syntactic Translation Rules

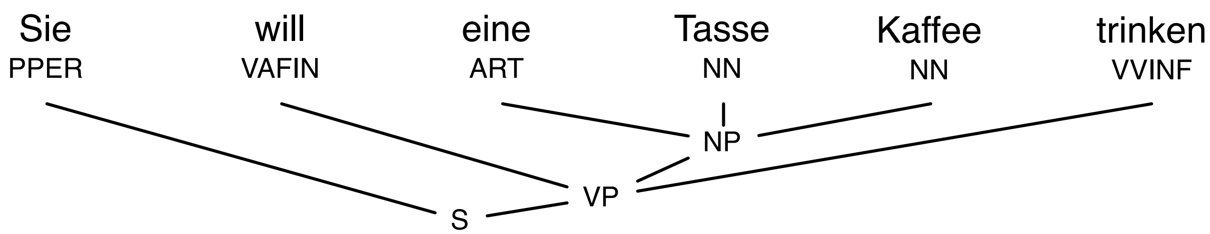


Minimal Rule Extraction



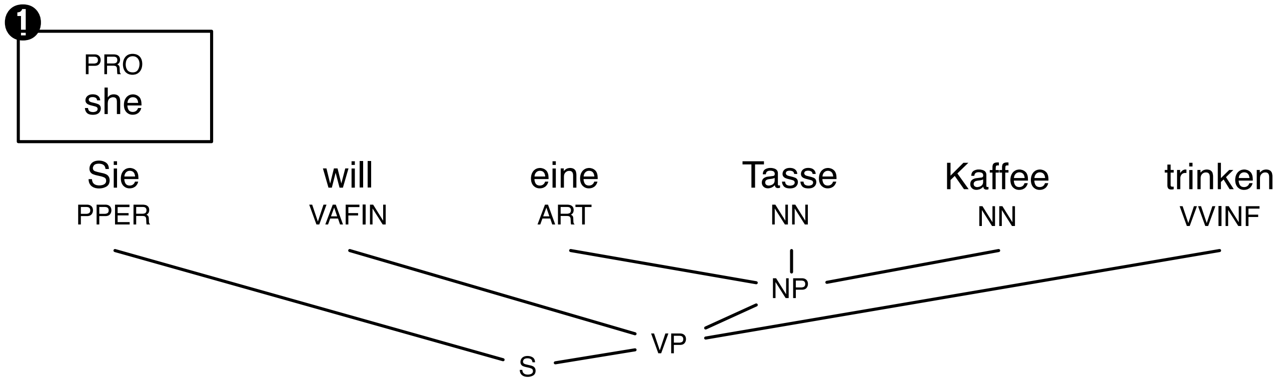
Align each node in the parse tree

Syntax Decoding



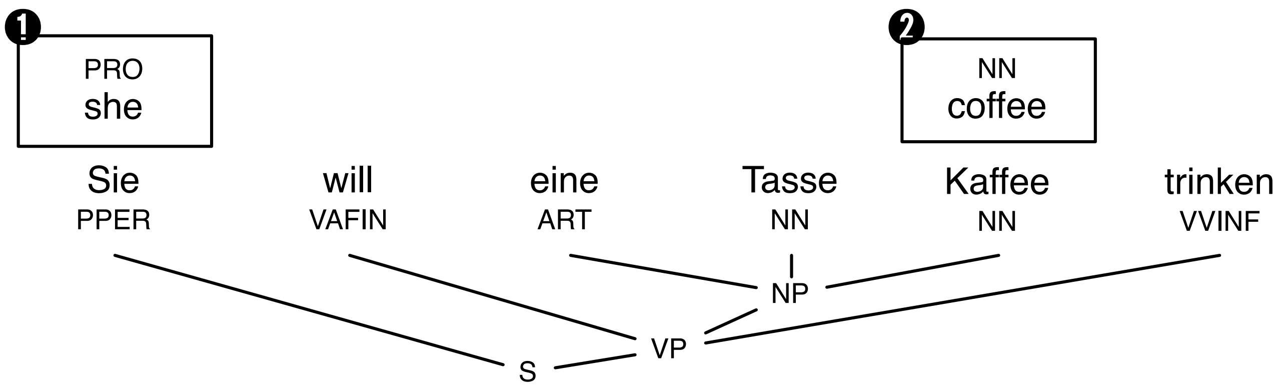
German input sentence with tree

Syntax Decoding



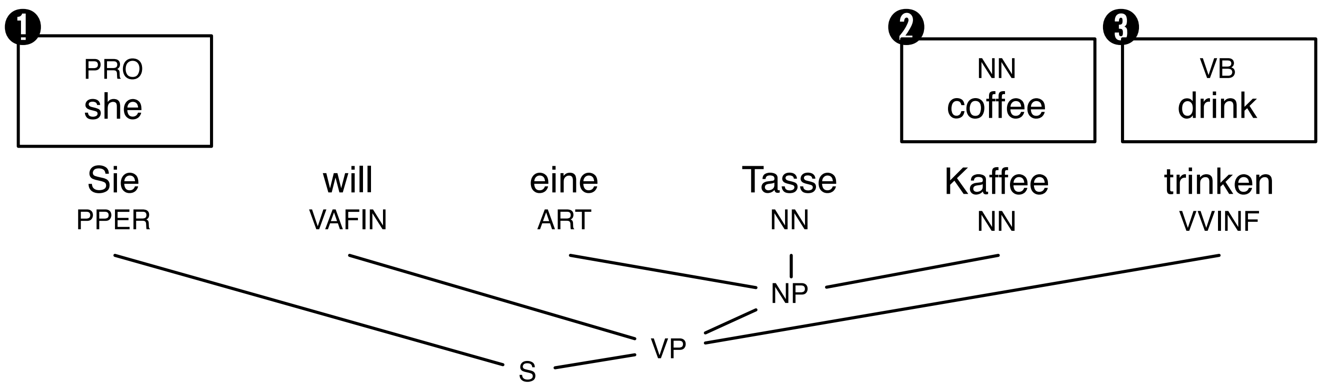
Purely lexical rule: filling a span with a translation (a constituent in the chart)

Syntax Decoding



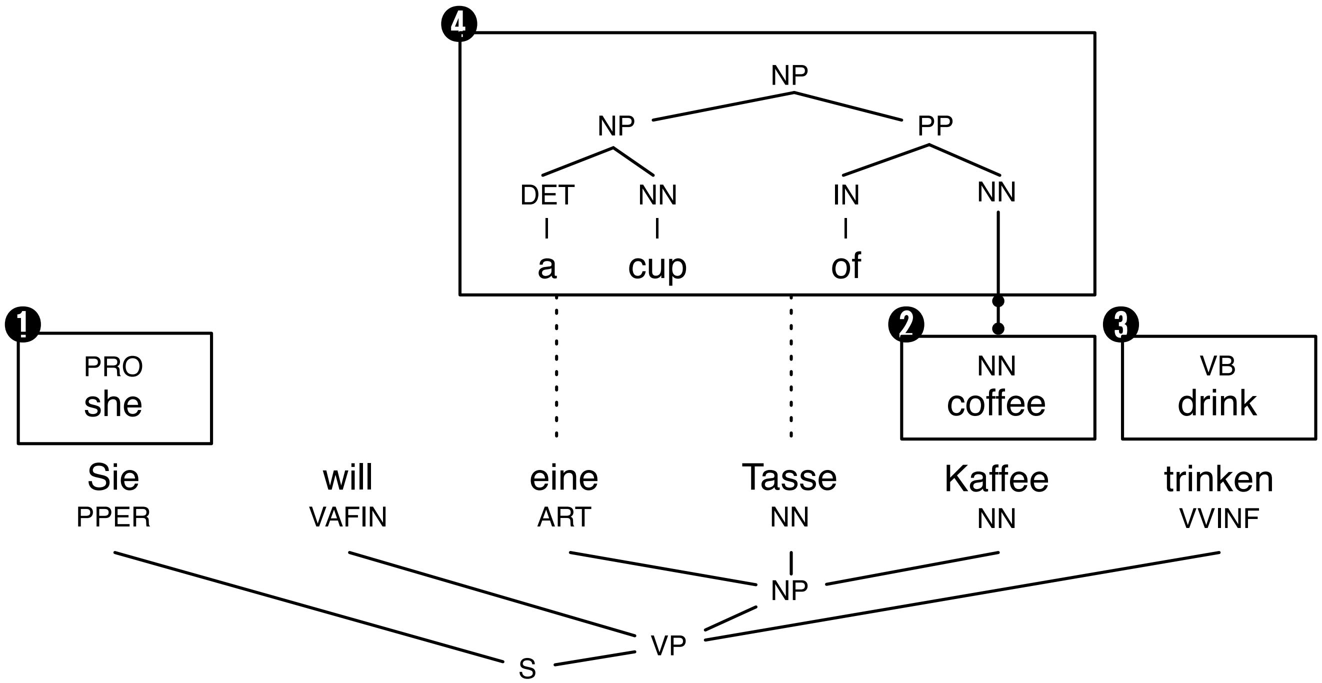
Purely lexical rule: filling a span with a translation (a constituent in the chart)

Syntax Decoding



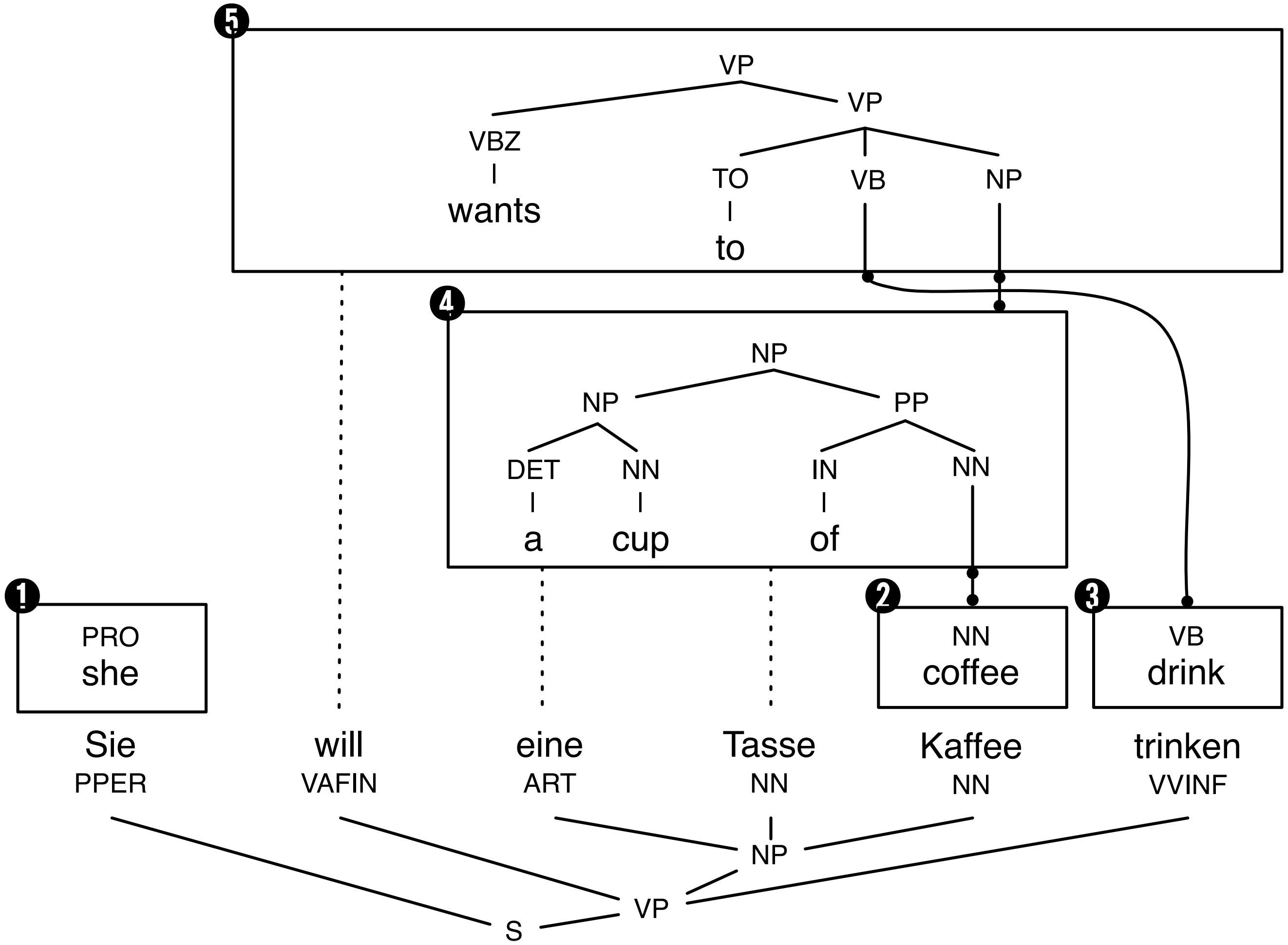
Purely lexical rule: filling a span with a translation (a constituent in the chart)

Syntax Decoding



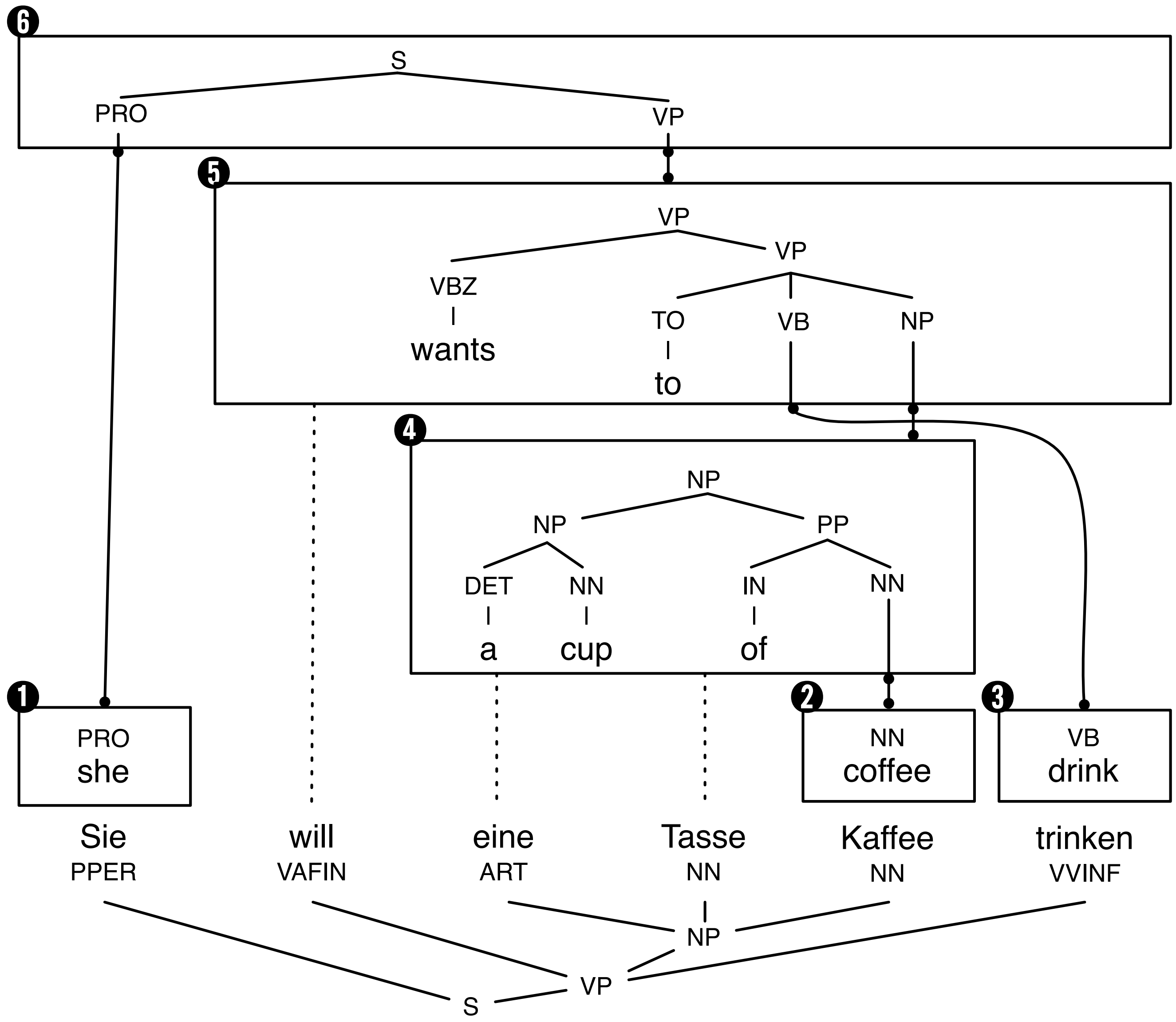
Complex rule: matching underlying constituent spans, and covering words

Syntax Decoding



Complex rule with reordering

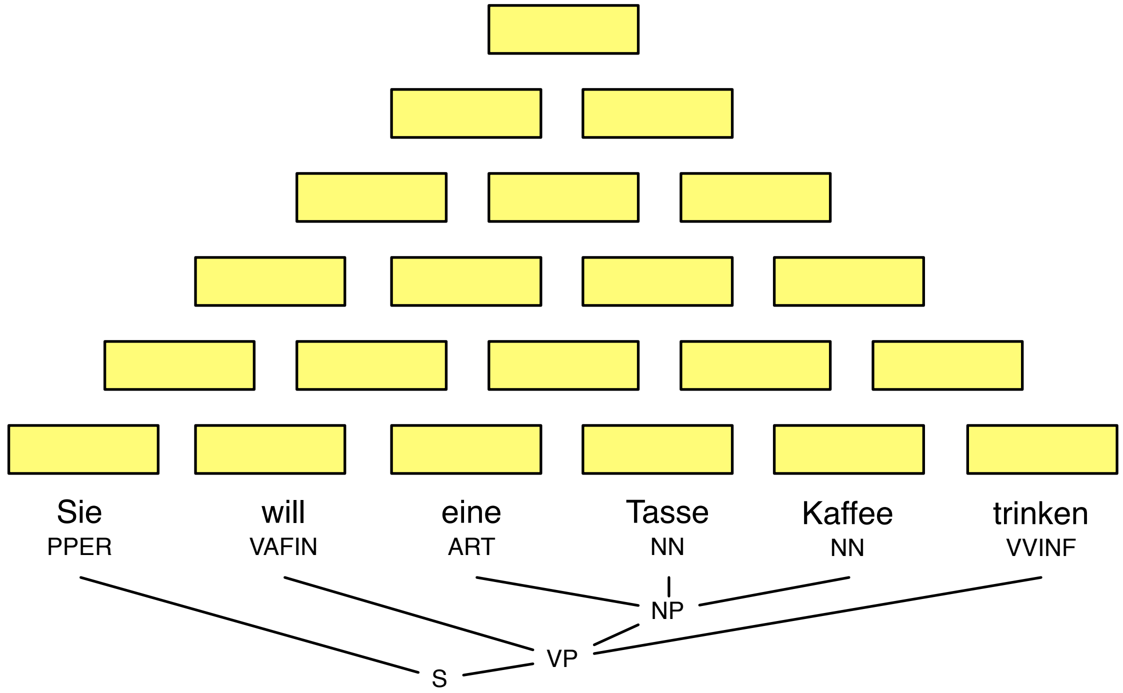
Syntax Decoding



Syntactic Decoding

Inspired by monolingual syntactic chart parsing:

During decoding of the source sentence,
a chart with translations for the $O(n^2)$ spans has to be filled



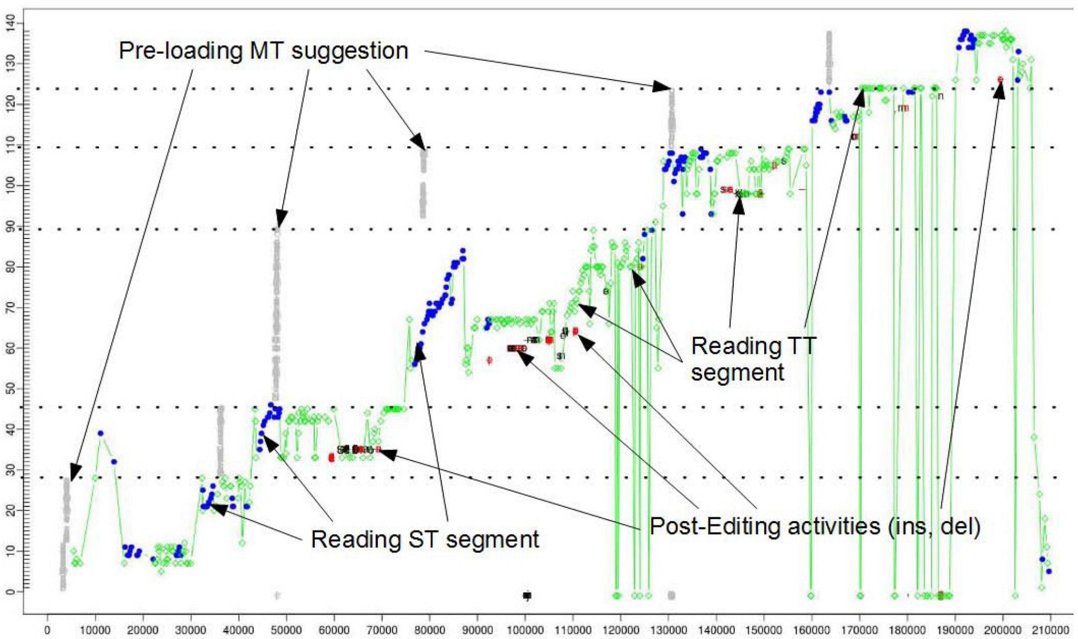
Comments

- Syntax-based models proven to work well for German, Chinese
- Decoding more complex and slower
- Needed: syntactic parser and hand-holding for each language pair

in defense of sequence models

Evidence from Human Translators

- Translation process studies (e.g., in CASMACAT)
- Humans start translating after reading a few words



Left-to-Right Parsing

Push Down Automaton

The interesting lecture ends soon

Left-to-Right Parsing

Push Down Automaton

look up POS tag

The	interesting	lecture	ends	soon
DET				

Left-to-Right Parsing

Push Down Automaton

look up POS tag

The	interesting	lecture	ends	soon
DET	JJ			
	DET			

Left-to-Right Parsing

Push Down Automaton

look up POS tag

The	interesting	lecture	ends	soon
DET	JJ	N		
	DET	JJ		
		DET		

Left-to-Right Parsing

Push Down Automaton

apply rule

The	interesting	lecture	ends	soon
DET	JJ	NP		
	DET			

Left-to-Right Parsing

Push Down Automaton

look up POS tag

The	interesting	lecture	ends	soon
DET	JJ	NP	VB	
	DET		NP	

Left-to-Right Parsing

Push Down Automaton

look up POS tag

The	interesting	lecture	ends	soon
DET	JJ	NP	VB	RB
	DET		NP	VB
				NP

Left-to-Right Parsing

Push Down Automaton

apply rule

The	interesting	lecture	ends	soon
DET	JJ	NP	VB	VP
	DET		NP	NP

Left-to-Right Parsing

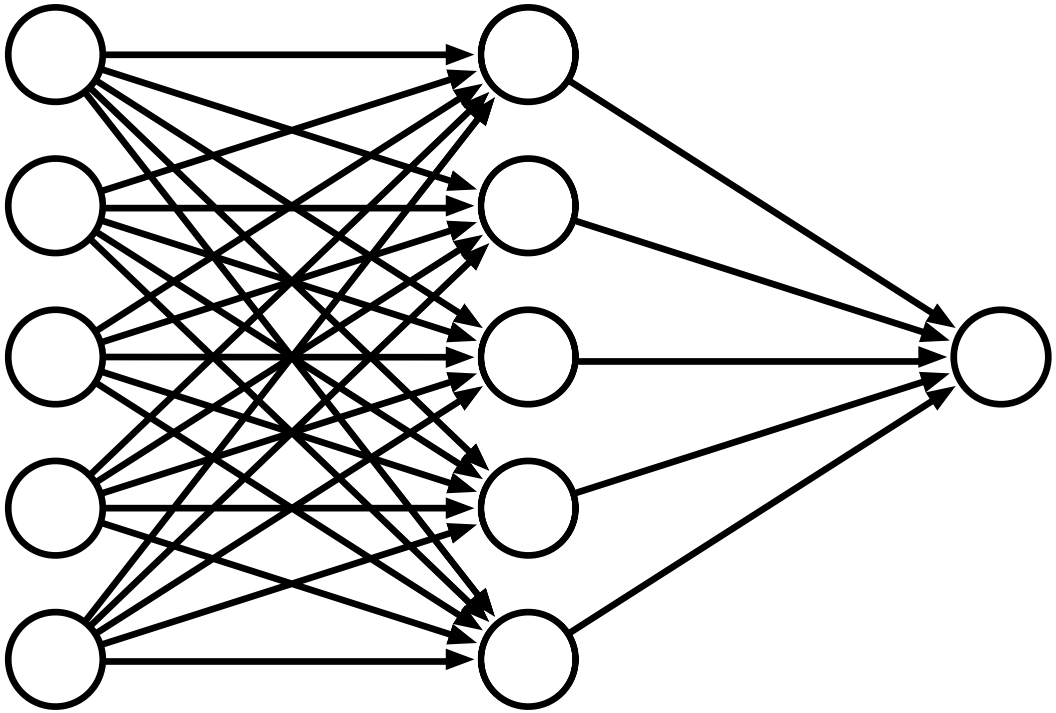
Push Down Automaton

apply rule

The	interesting	lecture	ends	soon
DET	JJ	NP	VB	S
	DET		NP	

neural translation

Neural Networks

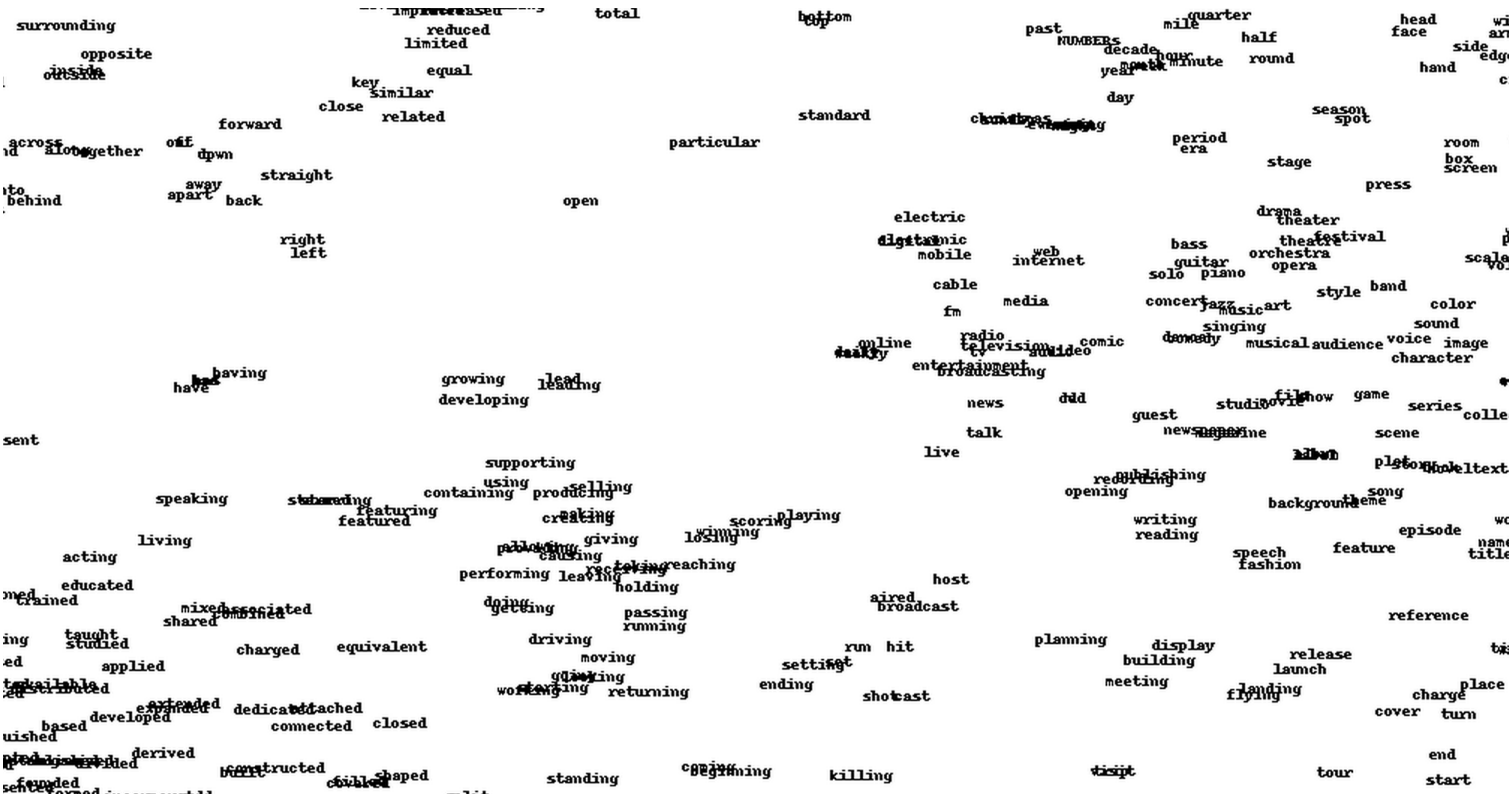


- Real valued vector representations
- Multiple layers of computation
- Non-linear functions

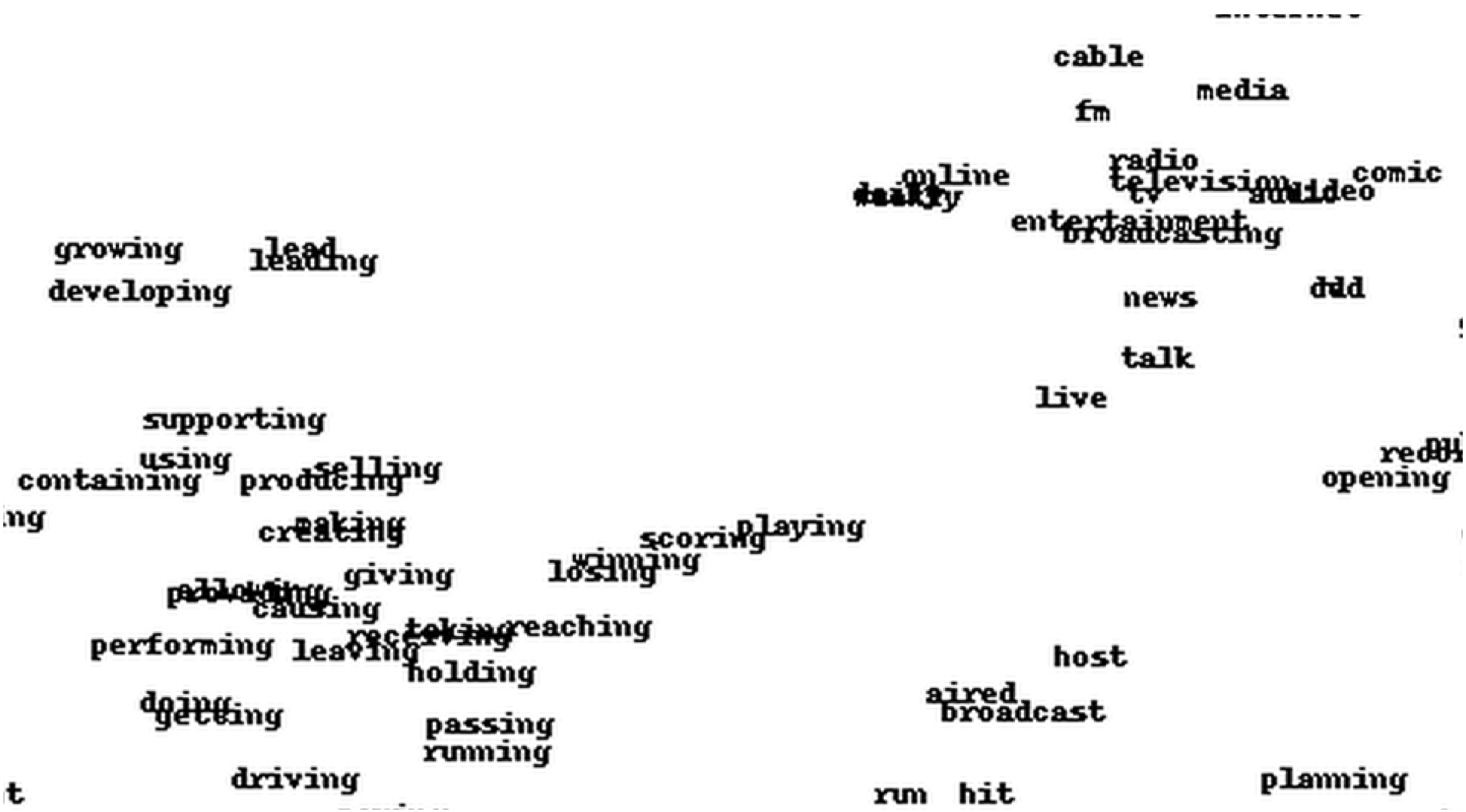
$$\vec{h} = \text{sigmoid}(W\vec{x})$$

$$\vec{y} = \text{sigmoid}(V\vec{h})$$

Word Embeddings



Word Embeddings



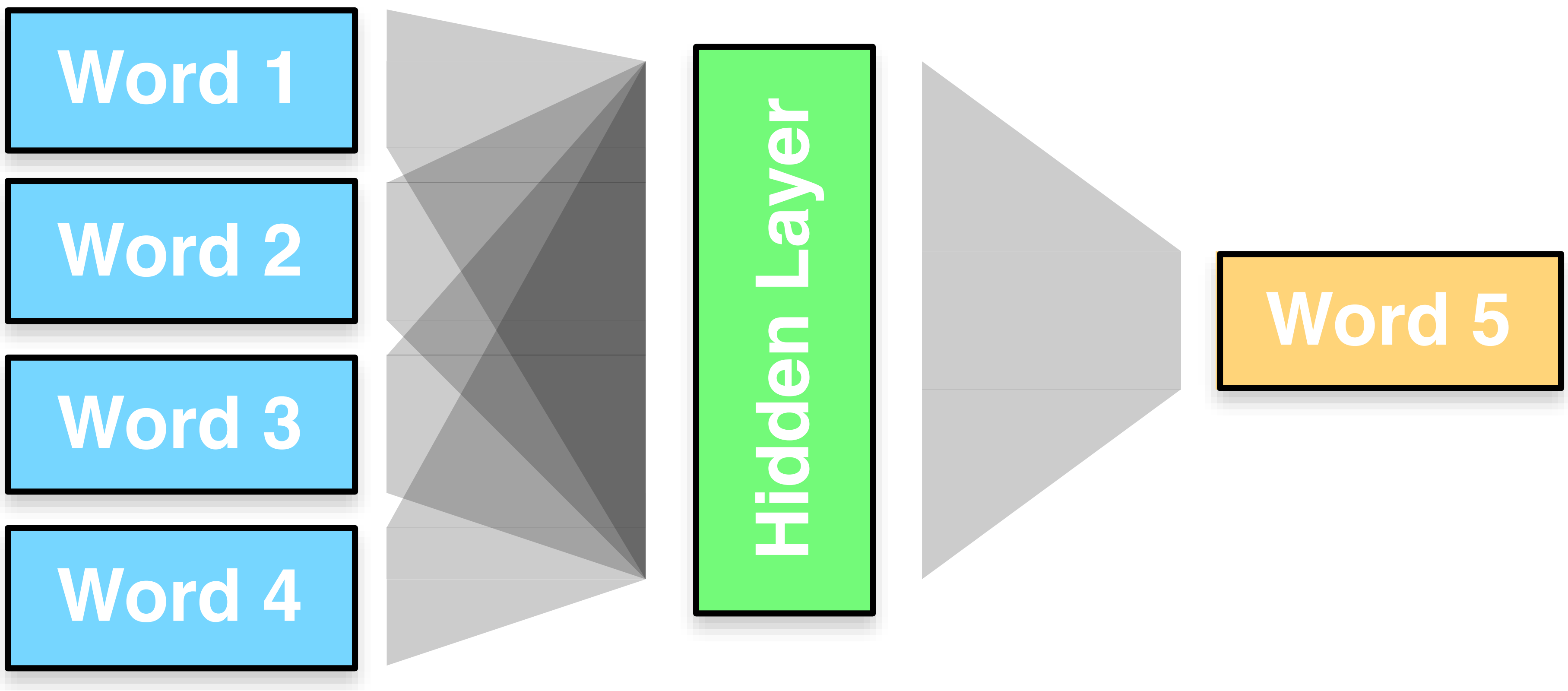
Why Neural Machine Translation?

- Word embeddings allow learning from *similar* examples

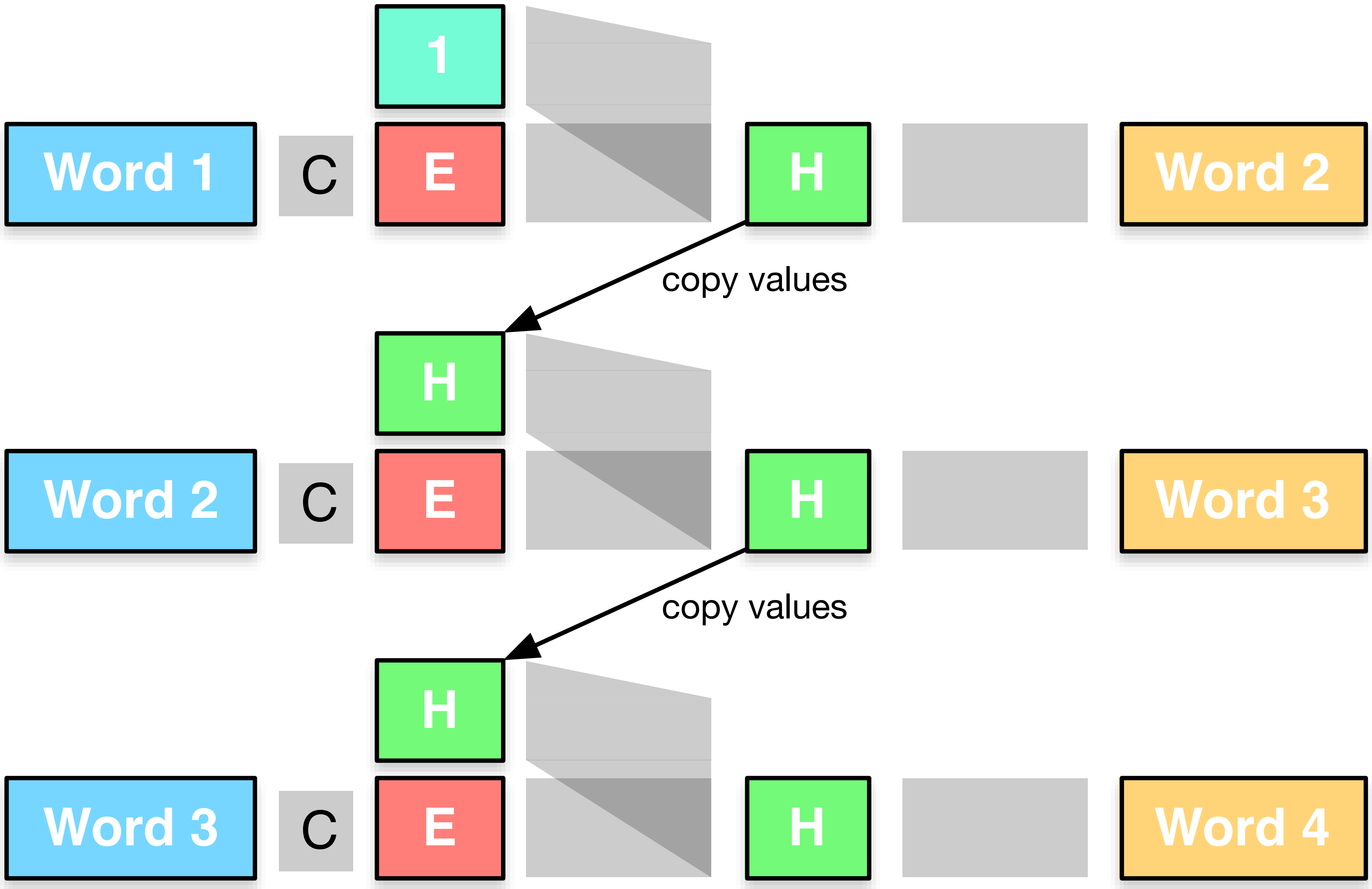
- Condition on a lot of context without backoff schemes

- Maybe there is something to non-linearity

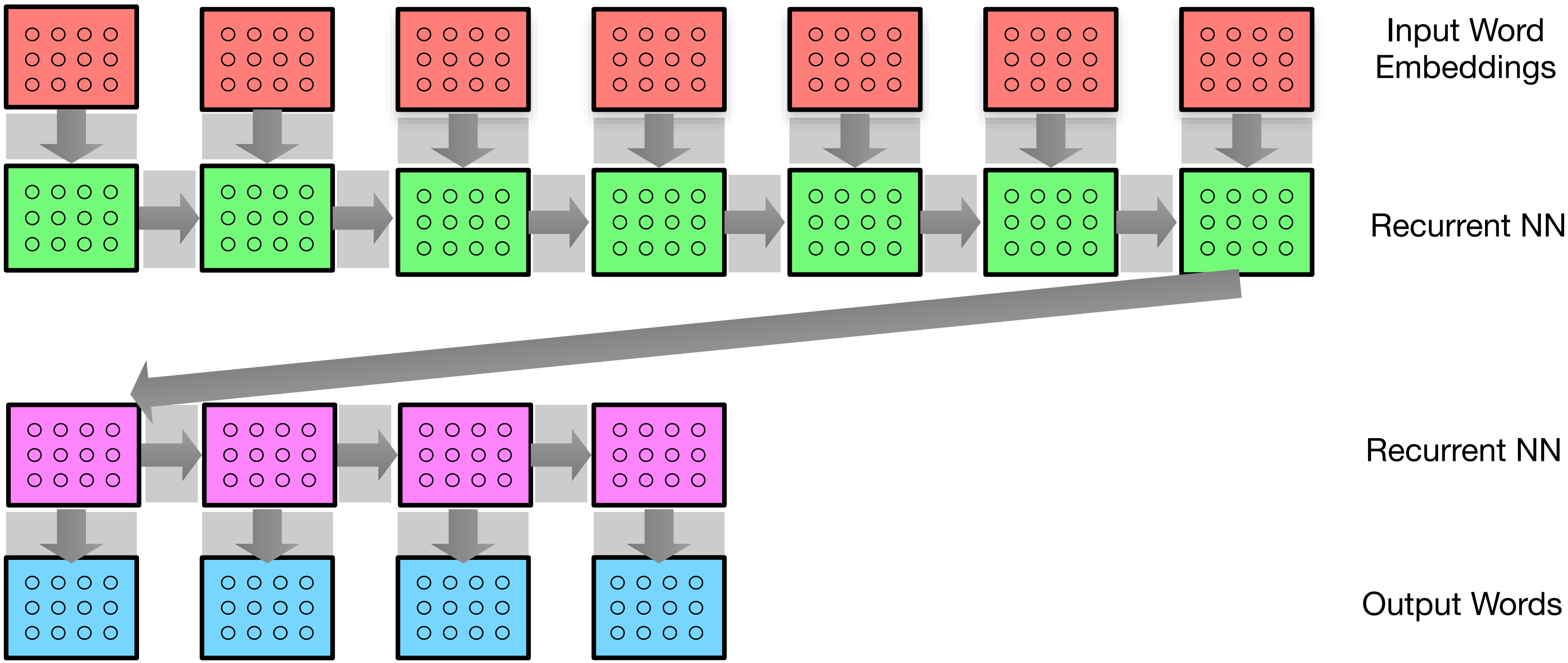
Neural N-Gram Language Model



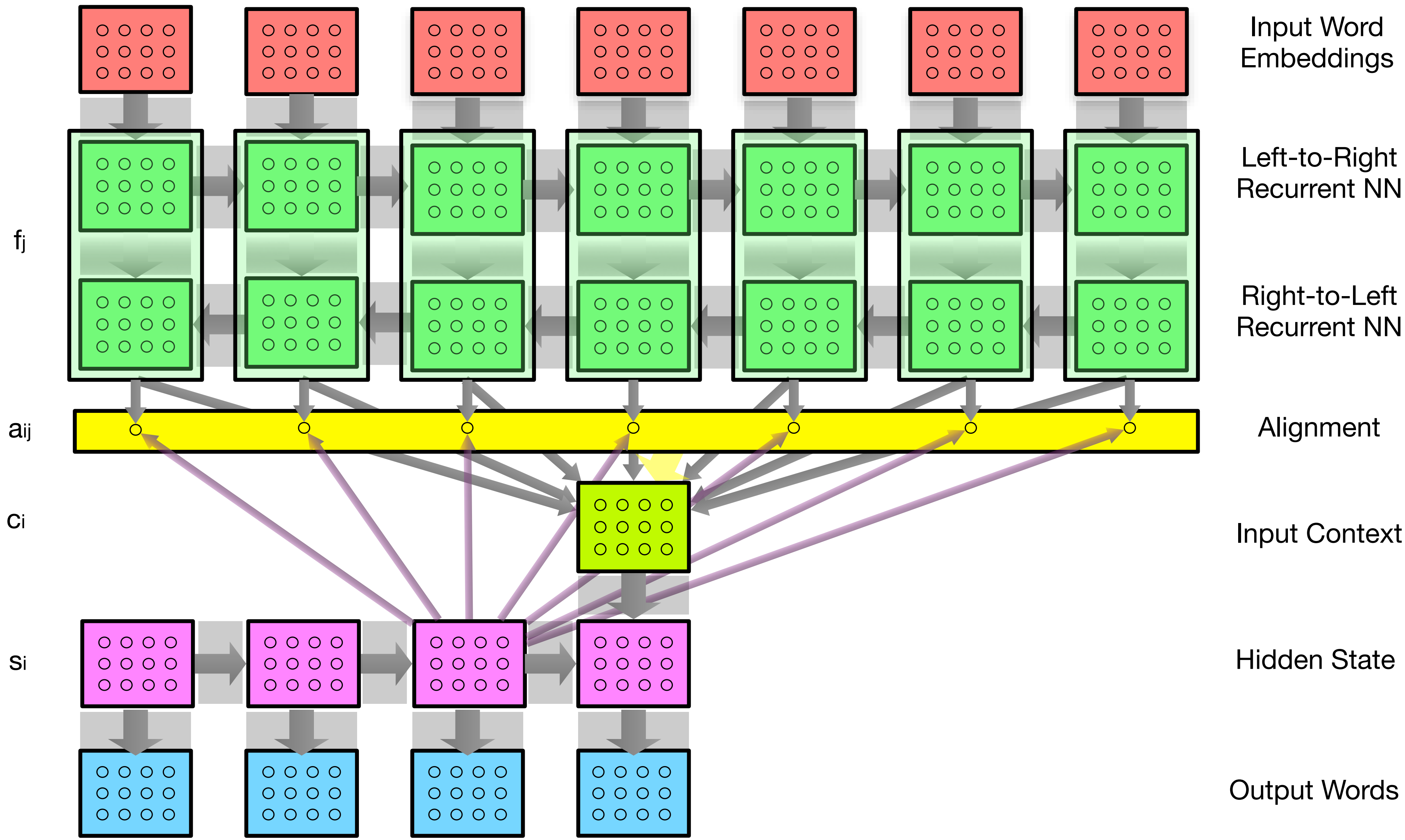
Recurrent Neural Networks



Encoder-Decoder Translation Model



Attention Translation Model



practical matters

How Good is MT?

Portuguese:

A seleção portuguesa de futebol, que se sagrou no domingo pela primeira vez campeã europeia, ao vencer por 1-0 a França na final, foi hoje recebida em euforia por milhares de pessoas no aeroporto Humberto Delgado, em Lisboa.

O avião Eusbio, que foi escoltado por dois aviões da Força Area Portuguesa desde a entrada em território português, aterrou em Lisboa às 12:40, tendo passado por um improvisado 'arco do triunfo', formado por dois jatos de água com as duas cores principais da bandeira nacional.

Google Translate:

The Portuguese national soccer team, which won on Sunday for the first time European champions by winning 1-0 to France in the final, was received today in euphoria by thousands of people at the airport Humberto Delgado in Lisbon.

The plane Eusebius, who was escorted by two aircraft of the Portuguese Air Force since the entry into Portuguese territory, landed in Lisbon at 12:40, having gone through a makeshift 'triumphal arch', formed by two water jets with two colors main national flag.

How Good is MT?

Portuguese:

A seleção portuguesa de futebol, que se sagrou no domingo pela primeira vez campeã europeia, ao vencer por 1-0 a França na final, foi hoje recebida em euforia por milhares de pessoas no aeroporto Humberto Delgado, em Lisboa.

O avião Eusbio, que foi escoltado por dois aviões da Força Area Portuguesa desde a entrada em território português, aterrou em Lisboa às 12:40, tendo passado por um improvisado 'arco do triunfo', formado por dois jatos de água com as duas cores principais da bandeira nacional.

Google Translate:

The Portuguese national soccer team, which won on Sunday for the first time European champions by winning 1-0 to France in the final, was received today in euphoria by thousands of people at the airport Humberto Delgado in Lisbon.

The plane Eusebius, who was escorted by two aircraft of the Portuguese Air Force since the entry into Portuguese territory, landed in Lisbon at 12:40, having gone through a makeshift 'triumphal arch', formed by two water jets with two colors main national flag.

What Works Best?

- WMT evaluation campaign

- Winner English–German (with official ties)

System	2008	2009	2010	2011	2012	2013	2014	2015	2016
rule	X	X		X	X	X			
phrase			X	X	X	X	X		
syntax							X	X	
neural								X	X

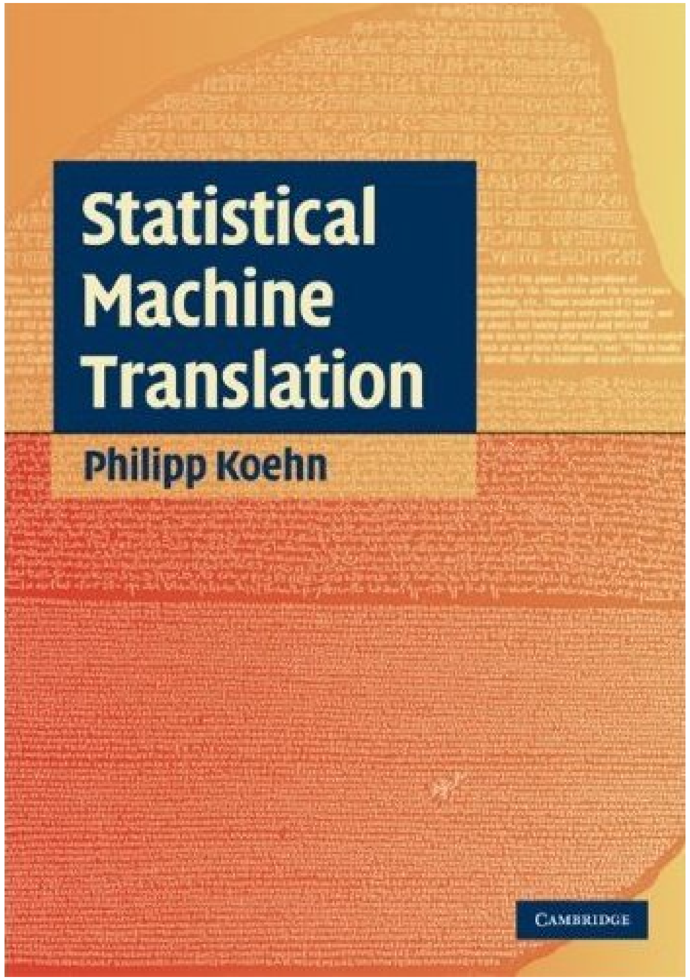
- For other language pairs, phrase-based systems dominated longer

Software

- **Moses** statistical machine translation toolkit
  - developed since 2006
  - reference implementation of state-of-the art methods
  - used in academia as benchmark and testbed
  - extensive commercial deployment
  - `http://www.statmt.org/moses/`

- **DL4MT** (or **Nematus**) neural translation toolkit
  - developed since 2016
  - state-of-the-art performance in 2016
  - `https://github.com/rsennrich/nematus`

Textbook



New chapter on neural machine translation:
http://mt-class.org/jhu/assets/papers/neural-network-models.pdf

Thank You

questions?