Machine Translation at Edinburgh

Factored Translation Models and Discriminative Training

Philipp Koehn, University of Edinburgh

9 July 2007

Overview

- Intro: Machine Translation at Edinburgh
- Factored Translation Models
- Discriminative Training

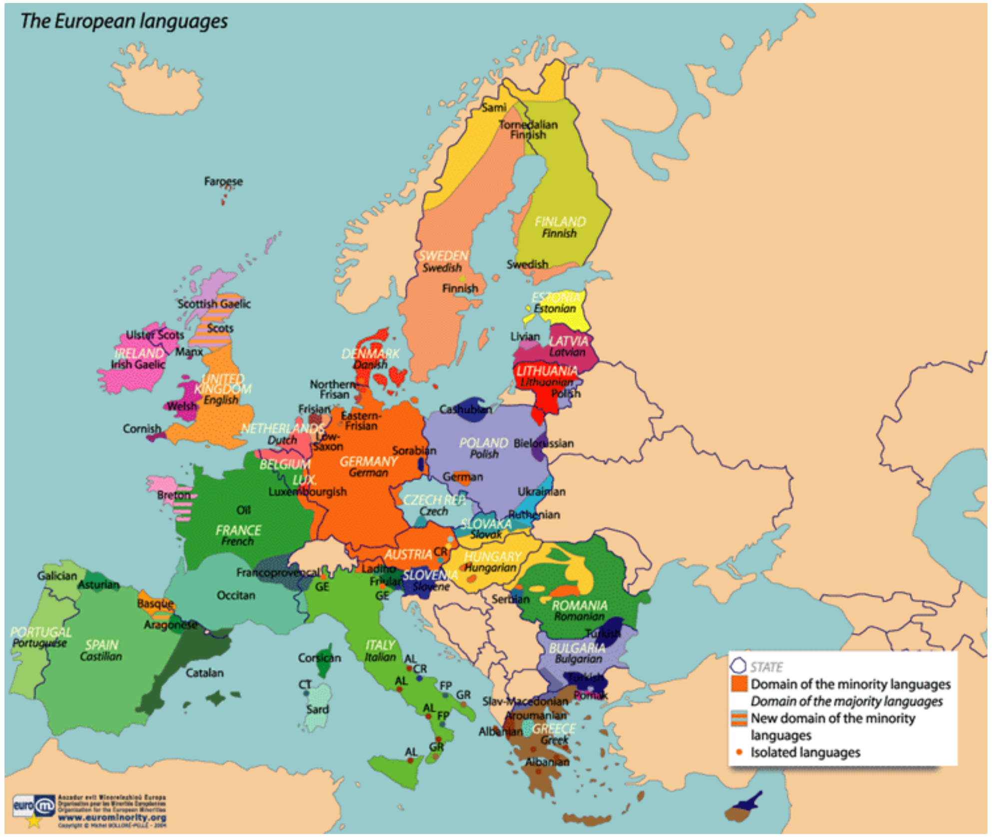
The European Challenge

Many languages

- 11 official languages in EU-15
- 20 official languages in EU-25
- many more minority languages

Challenge

- European reports, meetings, laws, etc.
- develop technology to **enable use of local languages** as much as possible



Existing MT systems for EU languages

[from Hutchins, 2005]

	Cze	Dan	Dut	Eng	Est	Fin	Fre	Ger	Gre	Hun	Ita	Lat	Lit	Mal	Pol	Por	Slo	Slo	Spa	Swe	
Czech	–	.	.	1	.	.	1	1	.	.	1	.	.	.	.	.	.	.	.	.	4
Danish	.	–	.	.	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	.	1
Dutch	.	.	–	6	.	.	2	1	.	.	.	.	.	.	.	.	.	.	.	.	9
English	2	.	6	–	.	.	42	48	3	3	29	1	.	.	7	30	2	.	48	1	222
Estonian	.	.	.	.	–	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0
Finnish	.	.	.	2	.	–	.	1	.	.	.	.	.	.	.	.	.	.	.	.	3
French	1	.	2	38	.	.	–	22	3	.	9	.	.	.	1	5	.	.	10	.	91
German	1	1	1	49	.	1	23	–	.	1	8	.	.	.	4	3	2	.	8	1	103
Greek	.	.	.	2	.	.	3	.	–	.	.	.	.	.	.	.	.	.	.	.	5
Hungarian	.	.	.	1	.	.	.	1	.	–	.	.	.	.	.	.	.	.	.	.	2
Italian	1	.	.	25	.	.	9	8	.	.	–	.	.	.	1	3	.	.	7	.	54
Latvian	.	.	.	1	.	.	.	.	.	.	.	–	.	.	.	.	.	.	.	.	1
Lithuanian	.	.	.	.	.	.	.	.	.	.	.	.	–	.	.	.	.	.	.	.	0
Maltese	.	.	.	.	.	.	.	.	.	.	.	.	.	–	.	.	.	.	.	.	0
Polish	.	.	.	6	.	.	1	3	.	.	1	.	.	.	–	2	.	.	1	.	14
Portuguese	.	.	.	25	.	.	4	4	.	.	3	.	.	.	1	–	.	.	6	.	43
Slovak	.	.	.	1	.	.	.	1	.	.	.	.	.	.	.	.	–	.	.	.	2
Slovene	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	–	.	.	0
Spanish	1	.	.	42	.	.	8	7	.	.	7	.	.	.	1	6	.	.	–	.	72
Swedish	.	.	.	2	.	.	.	1	.	.	.	.	.	.	.	.	.	.	.	–	3
	6	1	9	201	0	1	93	99	6	4	58	1	0	0	15	49	4	0	80	2	

Goals of the EuroMatrix Project

- Machine translation between **all EU language pairs**
  - baseline machine translation performance for all pairs
  - → starting point for national research efforts
  - more intensive effort on specific language pairs
- Creating an **open research** environment
  - open source **tools** for baseline machine translation system
  - collection of open data **resources**
  - open **evaluation campaigns** and **research workshops** ("marathons")
- Scientific **approaches**
  - **statistical** phrase-based, extended by factored approach
  - **hybrid** statistical/rule-based
  - tree-transfer based on **tecto-grammatic** probabilistic models

Translating between all EU-15 languages

- Statistical methods allow the rapid development of MT systems
- BLEU scores for 110 statistical machine translation systems

	da	de	el	en	es	fr	fi	it	nl	pt	sv
da	-	18.4	21.1	28.5	26.4	28.7	14.2	22.2	21.4	24.3	28.3
de	22.3	-	20.7	25.3	25.4	27.7	11.8	21.3	23.4	23.2	20.5
el	22.7	17.4	-	27.2	31.2	32.1	11.4	26.8	20.0	27.6	21.2
en	25.2	17.6	23.2	-	30.1	31.1	13.0	25.3	21.0	27.1	24.8
es	24.1	18.2	28.3	30.5	-	40.2	12.5	32.3	21.4	35.9	23.9
fr	23.7	18.5	26.1	30.0	38.4	-	12.6	32.4	21.1	35.3	22.6
fi	20.0	14.5	18.2	21.8	21.1	22.4	-	18.3	17.0	19.1	18.8
it	21.4	16.9	24.8	27.8	34.0	36.0	11.0	-	20.0	31.2	20.2
nl	20.5	18.3	17.4	23.0	22.9	24.6	10.3	20.0	-	20.7	19.0
pt	23.2	18.2	26.4	30.1	37.9	39.0	11.9	32.0	20.2	-	21.9
sv	30.3	18.9	22.8	30.2	28.6	29.7	15.3	23.9	21.9	25.9	-

[from Koehn, 2005]

Moses: Open Source Toolkit

- **Open source** statistical machine translation system (developed from scratch 2006)
  - state-of-the-art **phrase-based** approach
  - novel methods: **factored translation models**, **confusion network decoding**
  - support for **very large models** through **memory-efficient** data structures
- Documentation, source code, binaries **available at** `http://www.statmt.org/moses/`
- Development also **supported by**
  - EC-funded **TC-STAR** project
  - **US** funding agencies DARPA, NSF
  - universities (Edinburgh, Maryland, MIT, ITC-irst, RWTH Aachen, ...)

Factored Translation Models

- **Motivation**
- Example
- Model and Training
- Decoding
- Experiments
- Outlook

Statistical machine translation today

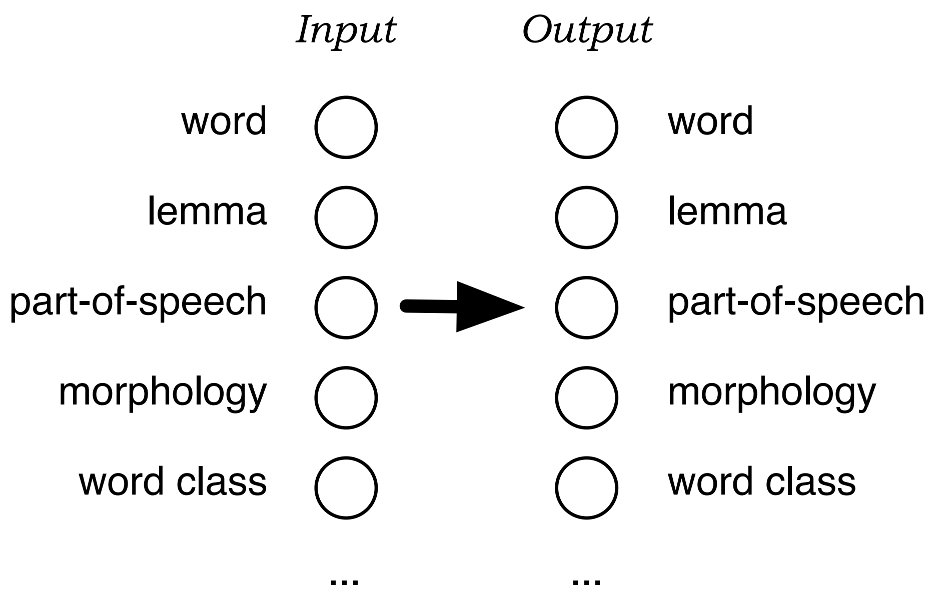
- Best performing methods based on **phrases**
  - short sequences of words
  - no use of explicit syntactic information
  - no use of morphological information
  - currently best performing method
- Progress in **syntax-based** translation
  - tree transfer models using syntactic annotation
  - still shallow representation of words and non-terminals
  - active research, improving performance

One motivation: morphology

- Models treat car and cars as completely different words
  - training occurrences of car have no effect on learning translation of cars
  - if we only see car, we do not know how to translate cars
  - rich morphology (German, Arabic, Finnish, Czech, ...) → many word forms
- Better approach
  - analyze surface word forms into **lemma** and **morphology**, e.g.: *car +plural*
  - translate lemma and morphology separately
  - generate target surface form

Factored translation models

- **Factored represention** of words



- Goals
  - **Generalization**, e.g. by translating lemmas, not surface forms
  - **Richer model**, e.g. using syntax for reordering, language modeling)

Related work

- **Back off** to representations with richer statistics (lemma, etc.)
  [Nießen and Ney, 2001, Yang and Kirchhoff 2006, Talbot and Osborne 2006]
- Use of additional annotation in **pre-processing** (POS, syntax trees, etc.)
  [Collins et al., 2005, Crego et al, 2006]
- Use of additional annotation in **re-ranking** (morphological features, POS, syntax trees, etc.)
  [Och et al. 2004, Koehn and Knight, 2005]

→ we pursue an **integrated approach**

- Use of syntactic **tree structure**
  [Wu 1997, Alshawi et al. 1998, Yamada and Knight 2001, Melamed 2004, Menezes and Quirk 2005, Chiang 2005, Galley et al. 2006]

→ may be **combined** with our approach

Factored Translation Models

- Motivation
- **Example**
- Model and Training
- Decoding
- Experiments
- Outlook

Decomposing translation: example

- **Translate** lemma and syntactic information **separately**

$$\begin{bmatrix} \text{lemma} \end{bmatrix} \Rightarrow \begin{bmatrix} \text{lemma} \end{bmatrix}$$

$$\begin{bmatrix} \text{part-of-speech} \\ \text{morphology} \end{bmatrix} \Rightarrow \begin{bmatrix} \text{part-of-speech} \\ \text{morphology} \end{bmatrix}$$

Decomposing translation: example

- **Generate surface** form on target side

$$\begin{bmatrix}\text{surface}\end{bmatrix}$$

$$\Uparrow$$

$$\begin{bmatrix}\text{lemma}\\ \text{part-of-speech}\\ \text{morphology}\end{bmatrix}$$

Translation process: example

Input: (Autos, Auto, NNS)

1. Translation step: lemma ⇒ lemma
   (?, car, ?), (?, auto, ?)
2. Generation step: lemma ⇒ part-of-speech
   (?, car, NN), (?, car, NNS), (?, auto, NN), (?, auto, NNS)
3. Translation step: part-of-speech ⇒ part-of-speech
   (?, car, NN), (?, car, NNS), (?, auto, NNP), (?, auto, NNS)
4. Generation step: lemma,part-of-speech ⇒ surface
   (car, car, NN), (cars, car, NNS), (auto, auto, NN), (autos, auto, NNS)

Factored Translation Models

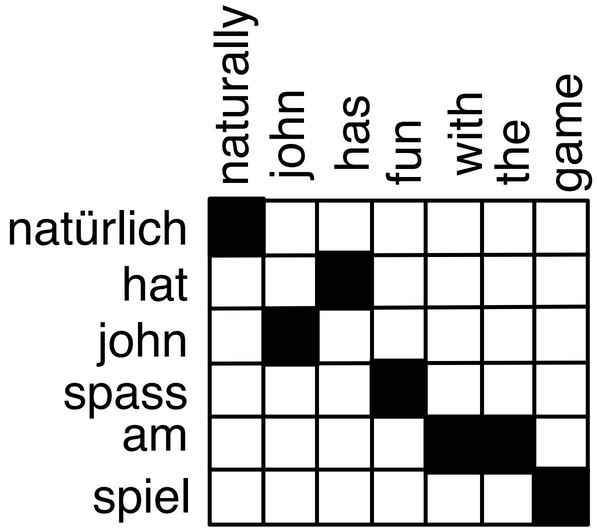
- Motivation
- Example
- **Model and Training**
- Decoding
- Experiments
- Outlook

Model

- Extension of **phrase model**
- Mapping of foreign words into English words broken up into steps
  - **translation step**: maps foreign factors into English factors (on the phrasal level)
  - **generation step**: maps English factors into English factors (for each word)
- Each step is modeled by one or more **feature functions**
  - fits nicely into log-linear model
  - weight set by discriminative training method
- Order of mapping steps is chosen to optimize search

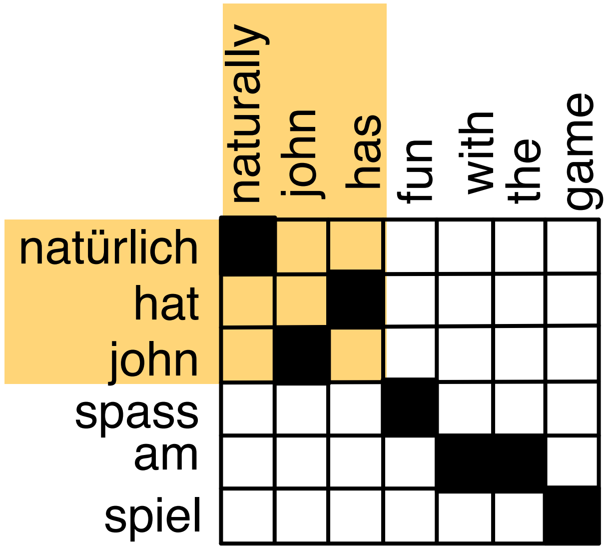
Phrase-based training

- Establish word alignment (GIZA++ and symmetrization)



Phrase-based training

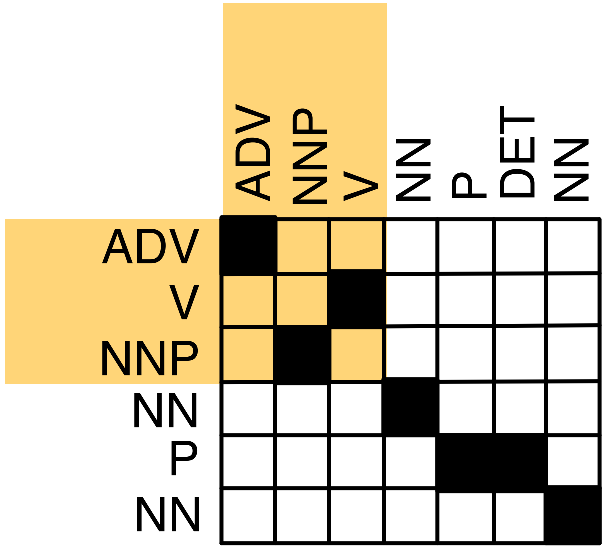
- Extract phrase



⇒ natürlich hat john — naturally john has

Factored training

- Annotate training with factors, extract phrase



$\Rightarrow$ ADV V NNP — ADV NNP V

Training of generation steps

- Generation steps map target factors to target factors
  - typically trained on target side of parallel corpus
  - may be trained on additional monolingual data
- Example: The/DET man/NN sleeps/VBZ
  - count collection
    - count(the,DET)++
    - count(man,NN)++
    - count(sleeps,VBZ)++
  - evidence for probability distributions (max. likelihood estimation)
    - p(DET|the), p(the|DET)
    - p(NN|man), p(man|NN)
    - p(VBZ|sleeps), p(sleeps|VBZ)

Factored Translation Models

- Motivation
- Example
- Model and Training
- **Decoding**
- Experiments
- Outlook

Phrase-based translation

- Task: **translate this sentence** from German into English

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

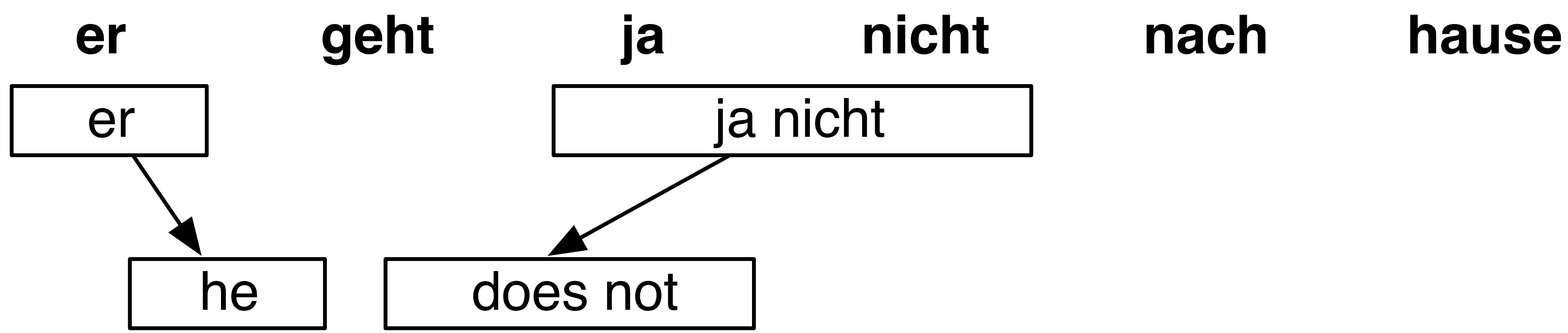
Translation step 1

- Task: translate this sentence from German into English

- **Pick** phrase in input, **translate**

Translation step 2

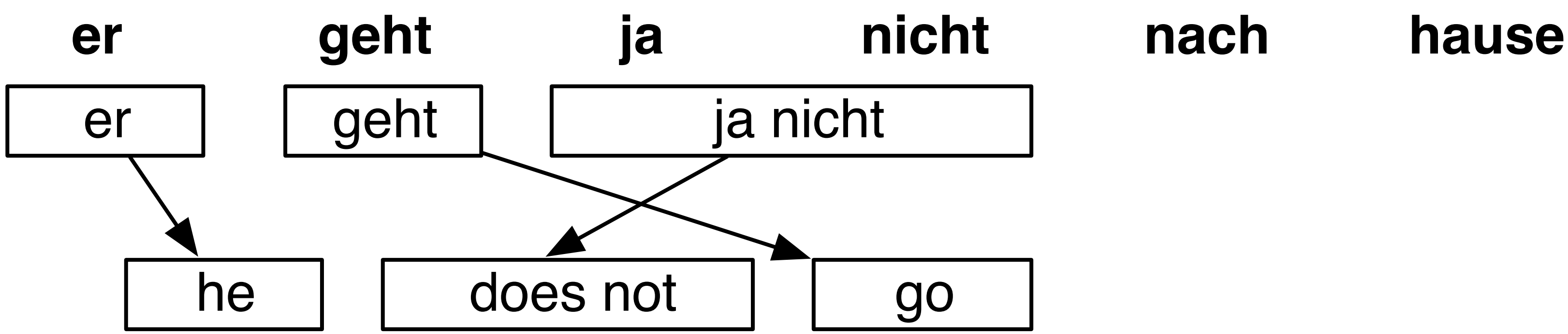
- Task: translate this sentence from German into English



- 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 step 3

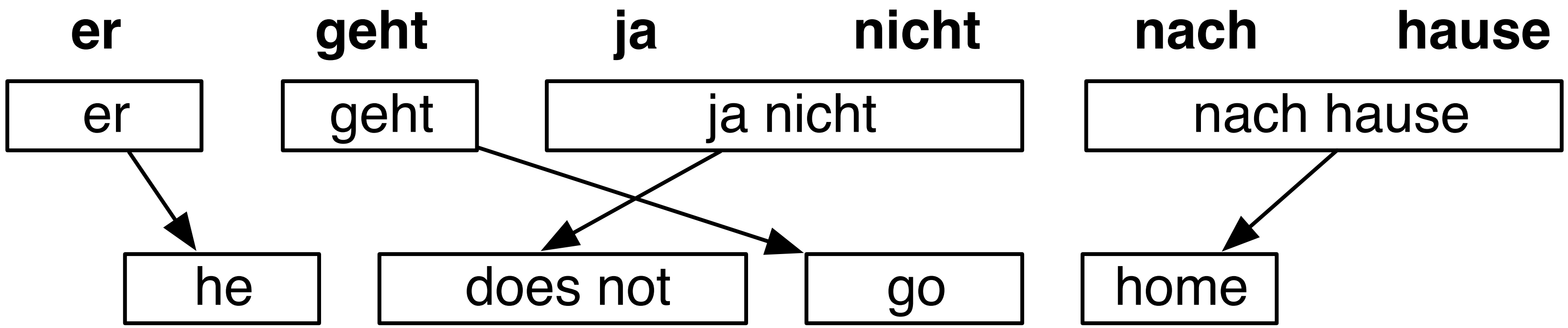
- Task: translate this sentence from German into English



- Pick phrase in input, translate

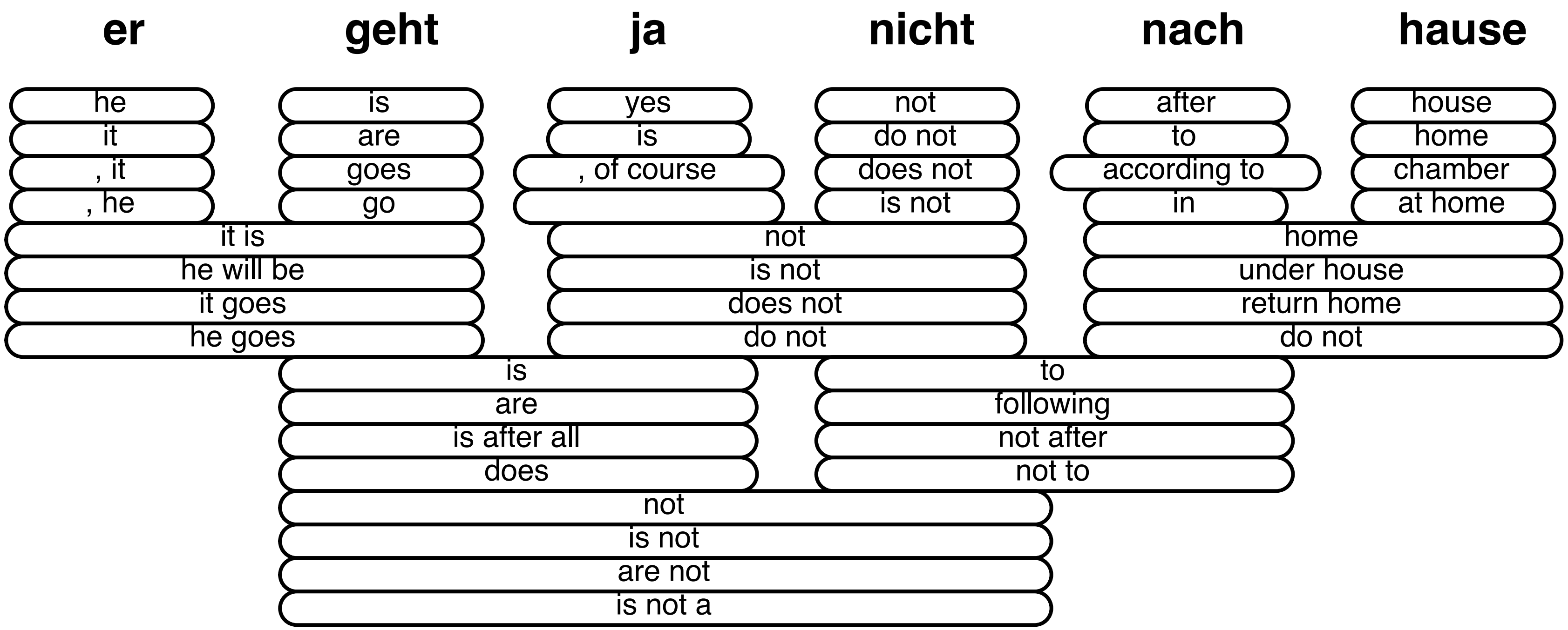
Translation step 4

- Task: translate this sentence from German into English



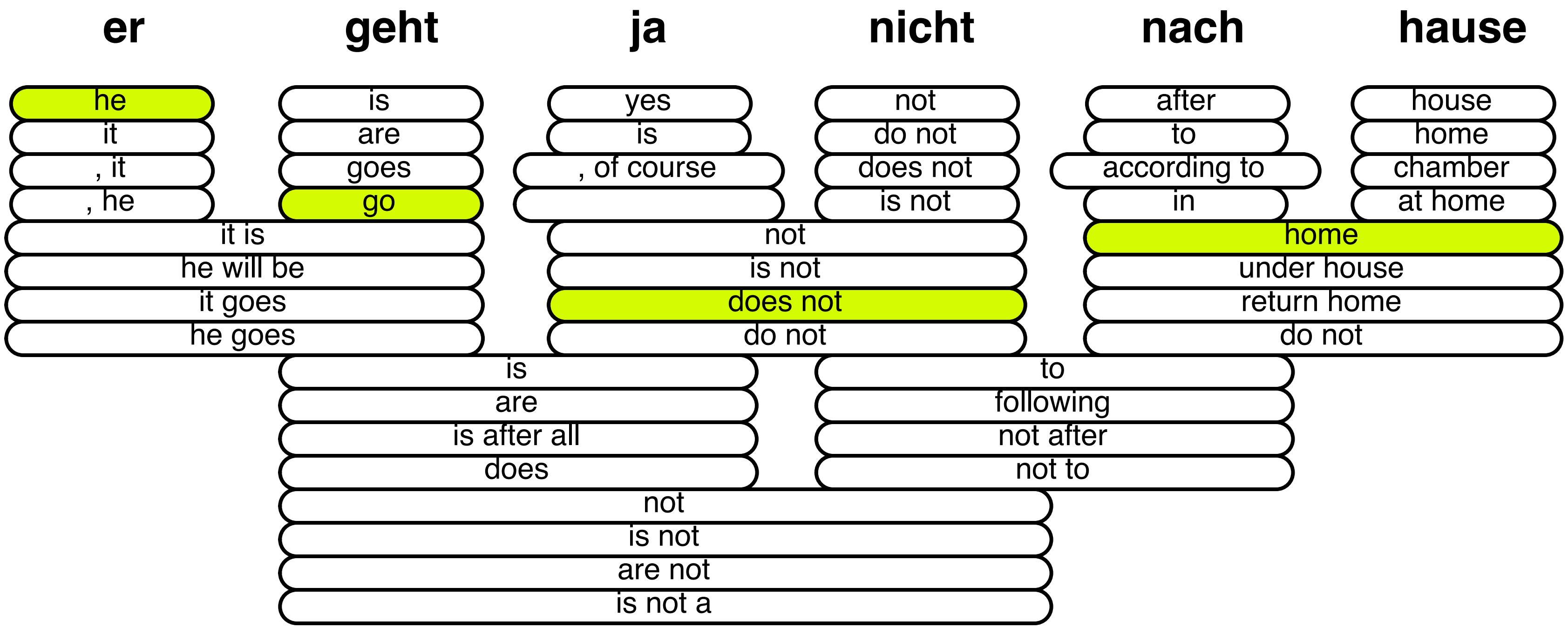
- Pick phrase in input, translate

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

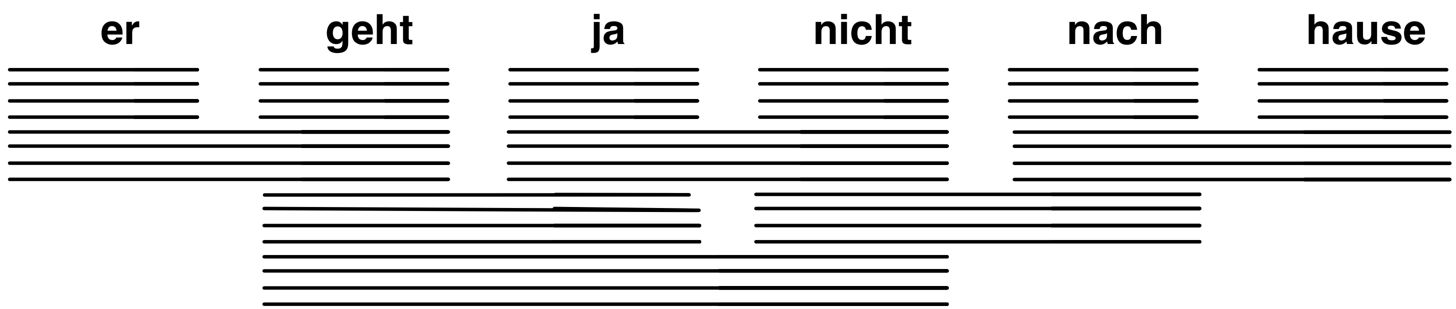
Translation options



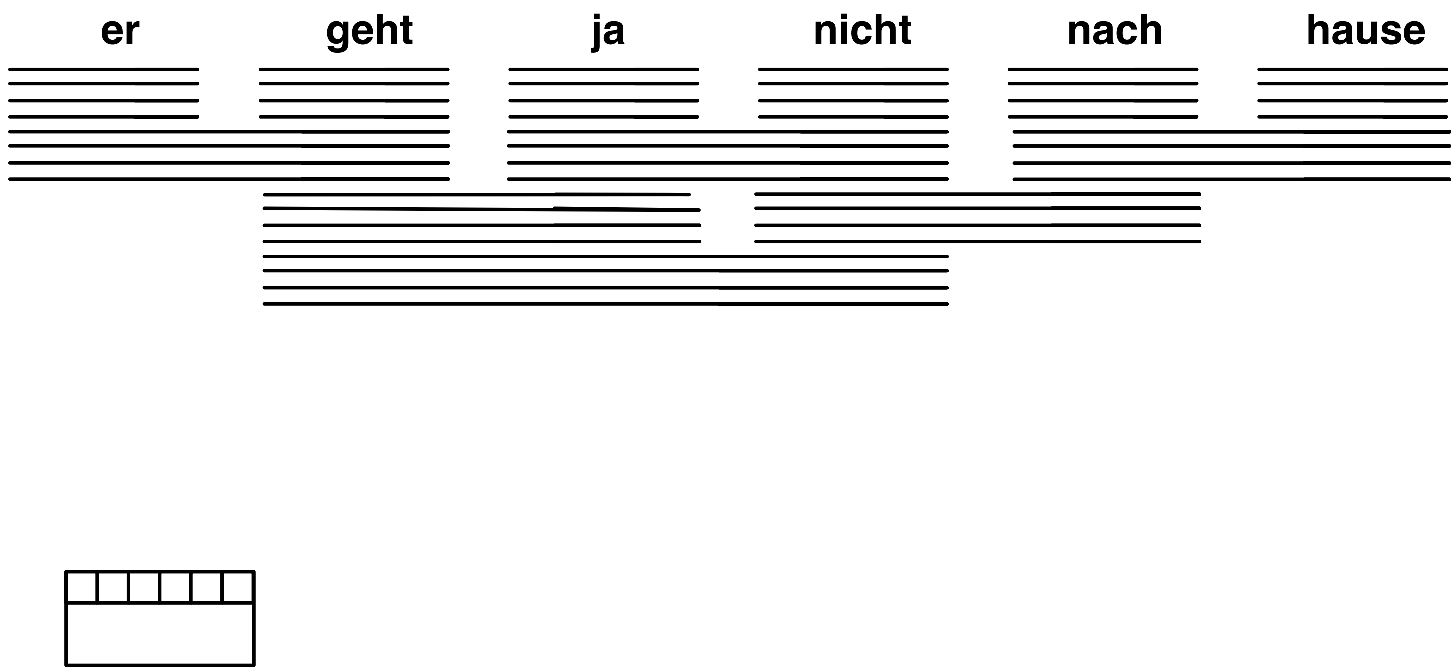
- The machine translation decoder does not know the right answer

→ **Search problem** solved by heuristic beam search

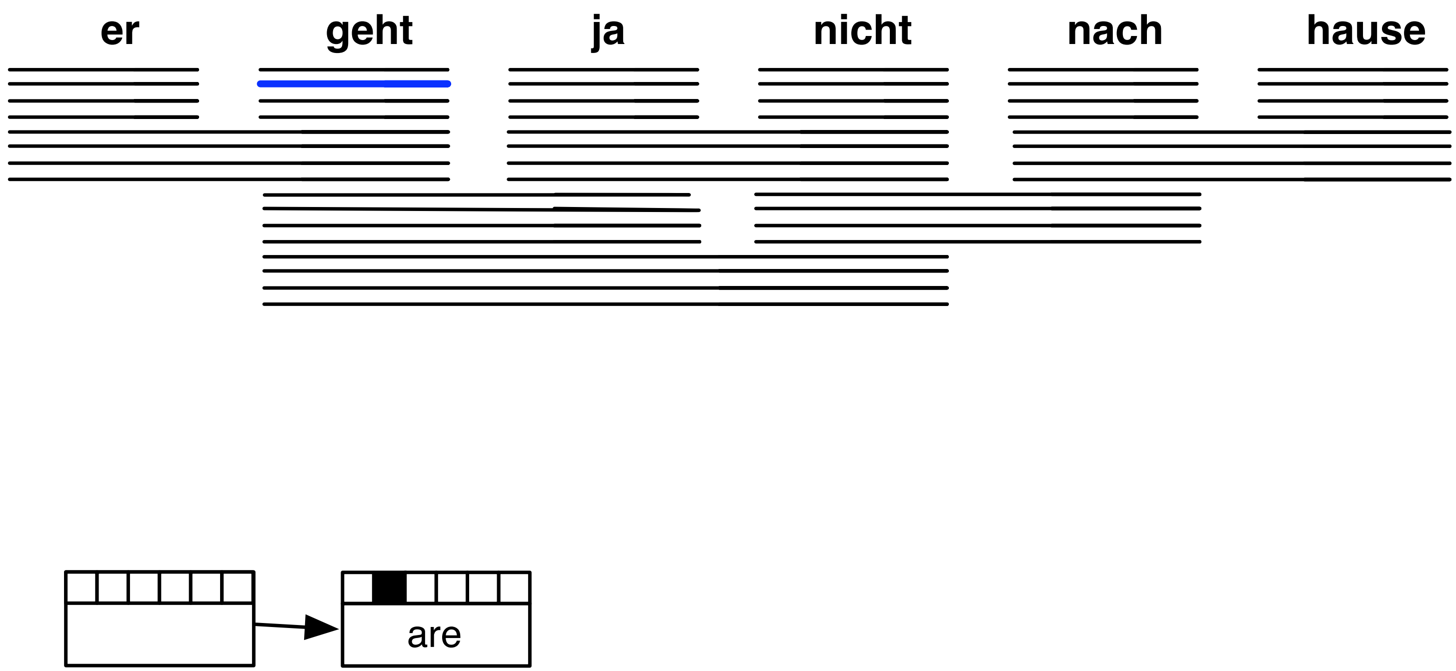
Decoding process: precompute translation options



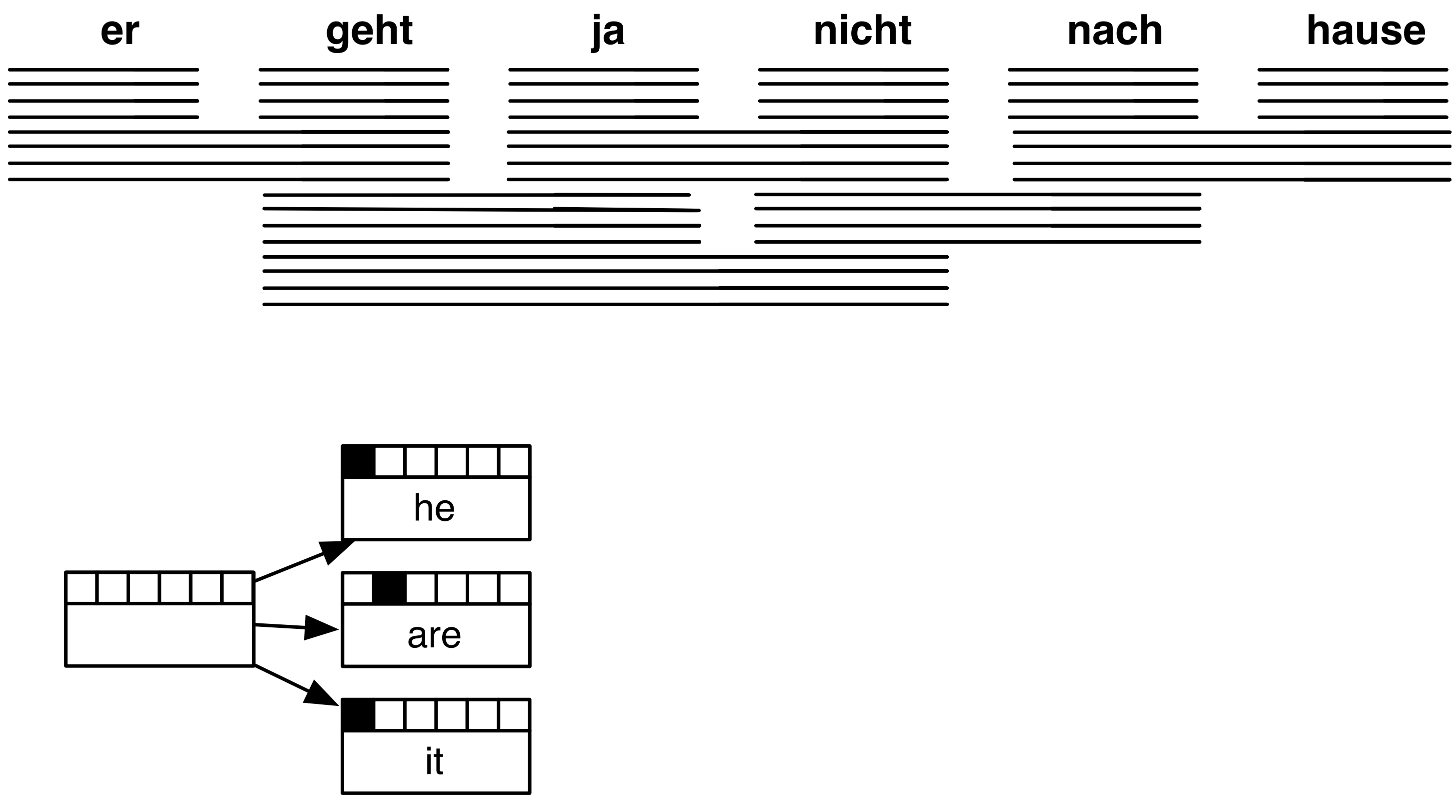
Decoding process: start with initial hypothesis



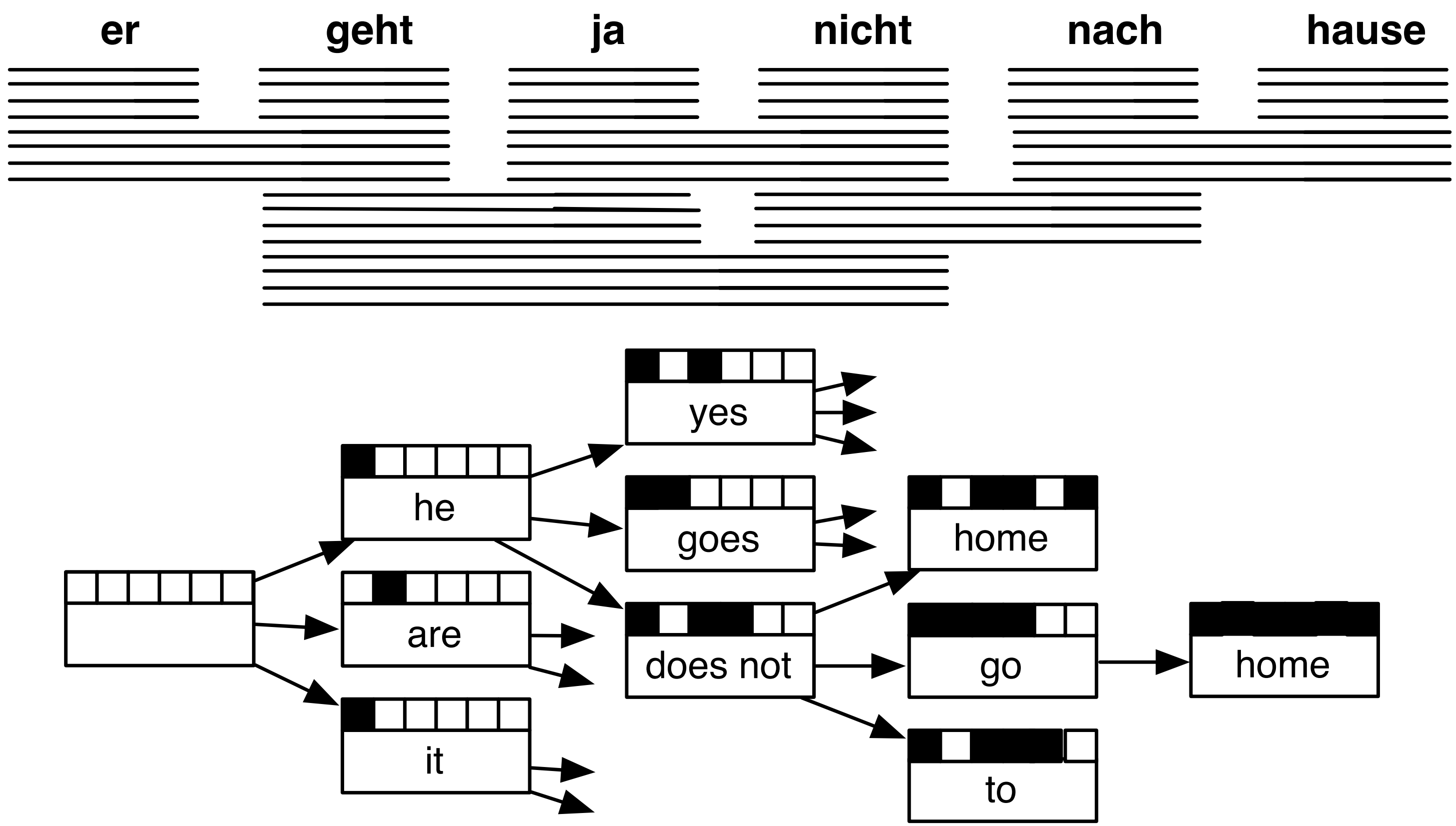
Decoding process: hypothesis expansion



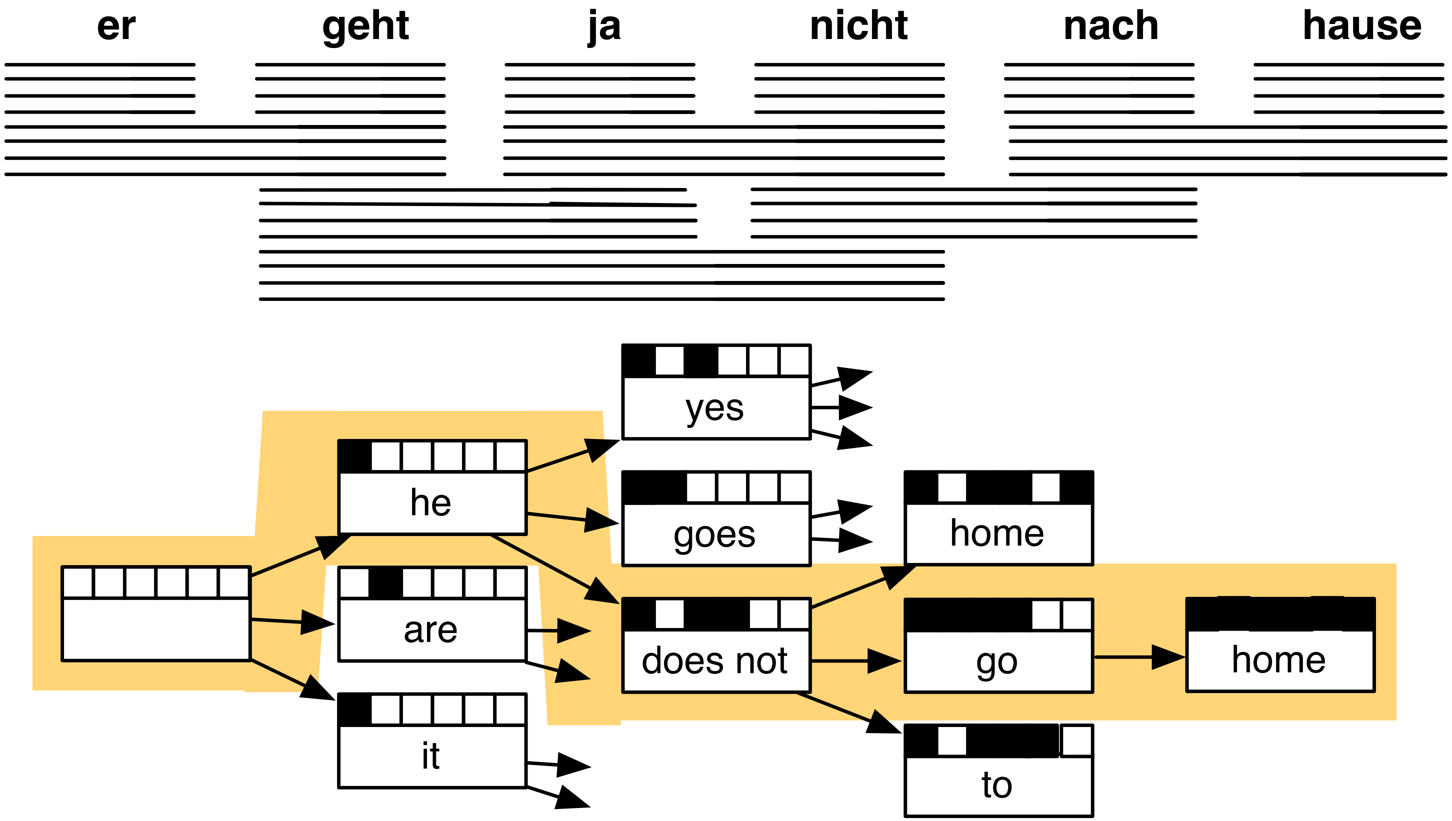
Decoding process: hypothesis expansion



Decoding process: hypothesis expansion



Decoding process: find best path

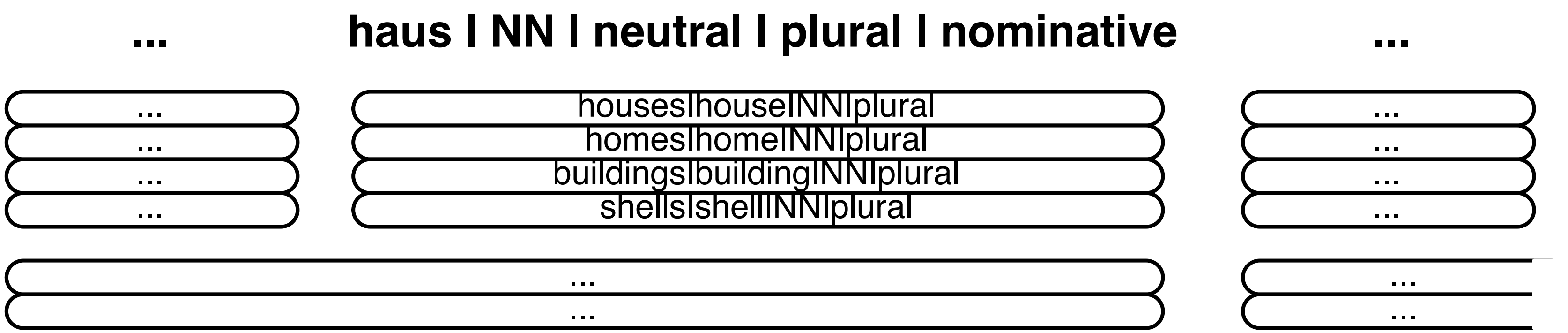


Factored model decoding

- Factored model decoding introduces **additional complexity**
- Hypothesis expansion not any more according to simple translation table, but by **executing a number of mapping steps**, e.g.:
  1. translating of lemma → lemma
  2. translating of part-of-speech, morphology → part-of-speech, morphology
  3. generation of surface form
- Example: haus|NN|neutral|plural|nominative → { houses|house|NN|plural, homes|home|NN|plural, buildings|building|NN|plural, shells|shell|NN|plural }
- Each time, a hypothesis is expanded, these mapping steps have to applied

Efficient factored model decoding

- Key insight: executing of mapping steps can be **pre-computed** and stored as translation options
  - apply mapping steps to all input phrases
  - store results as **translation options**
- → decoding algorithm **unchanged**



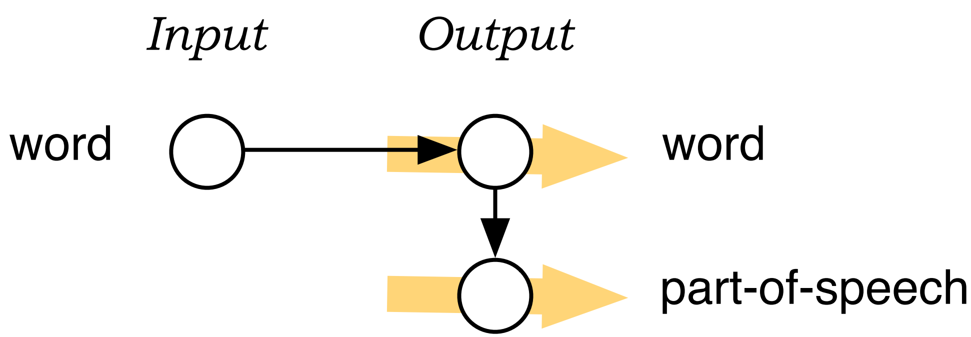
Efficient factored model decoding

- Problem: **Explosion** of translation options
  - originally limited to 20 per input phrase
  - even with simple model, now 1000s of mapping expansions possible
- Solution: **Additional pruning** of translation options
  - **keep only the best** expanded translation options
  - current default 50 per input phrase
  - decoding only about 2-3 times slower than with surface model

Factored Translation Models

- Motivation
- Example
- Model and Training
- Decoding
- **Experiments**
- Outlook

Adding linguistic markup to output



- Generation of POS tags on the target side
- Use of high order language models over POS (7-gram, 9-gram)
- Motivation: syntactic tags should enforce syntactic sentence structure model not strong enough to support major restructuring

Some experiments

- English–German, Europarl, 30 million word, test2006

Model	**BLEU**
best published result	18.15
baseline (surface)	18.04
surface + POS	18.15

- German–English, News Commentary data (WMT 2007), 1 million word

Model	**BLEU**
Baseline	18.19
With POS LM	19.05

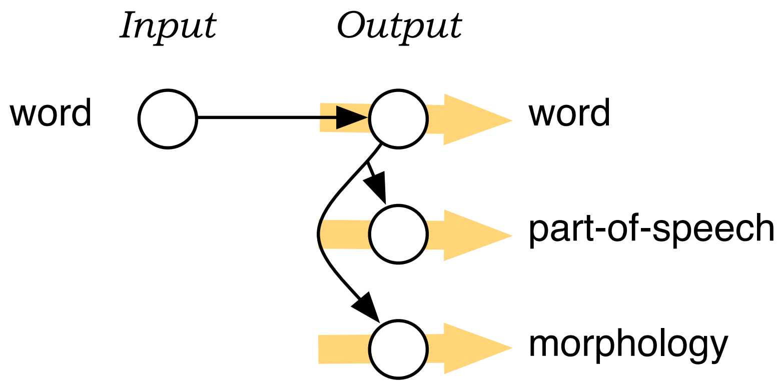
- Improvements under sparse data conditions
- Similar results with CCG supertags [Birch et al., 2007]

Sequence models over morphological tags

die	hellen	Sterne	erleuchten	das	schwarze	Himmel
(the)	(bright)	(stars)	(illuminate)	(the)	(black)	(sky)
fem	fem	fem	-	neutral	neutral	male
plural	plural	plural	plural	sgl.	sgl.	sgl
nom.	nom.	nom.	-	acc.	acc.	acc.

- Violation of noun phrase agreement in gender
  - das schwarze and schwarze Himmel are perfectly fine bigrams
  - but: das schwarze Himmel is not
- If relevant n-grams does not occur in the corpus, a lexical n-gram model would **fail to detect** this mistake
- Morphological sequence model: p(N-male|J-neutral) $>$ p(N-male|J-neutral)

Local agreement (esp. within noun phrases)



- High order language models over POS and morphology
- Motivation
  - DET-sgl NOUN-sgl good sequence
  - DET-sgl NOUN-plural bad sequence

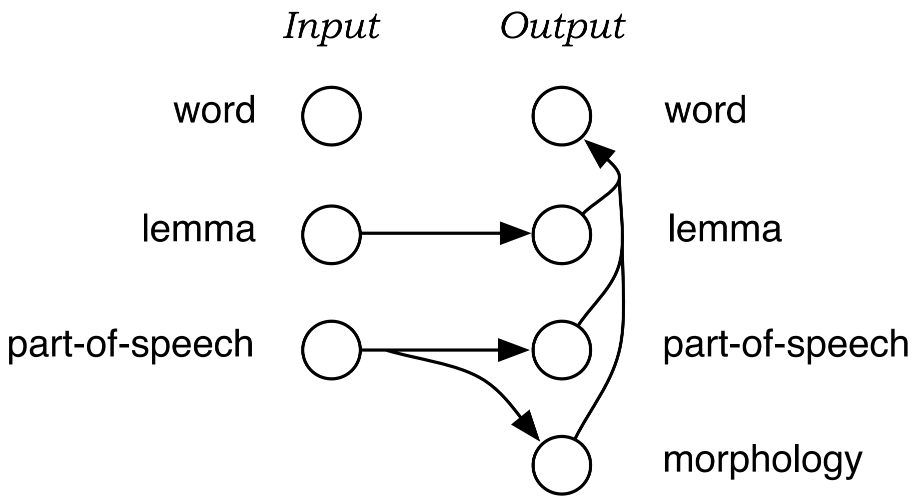
Agreement within noun phrases

- Experiment: 7-gram POS, morph LM in addition to 3-gram word LM
- Results

Method	Agreement errors in NP	devtest	test
baseline	15% in NP $\geq$ 3 words	18.22 BLEU	18.04 BLEU
factored model	4% in NP $\geq$ 3 words	18.25 BLEU	18.22 BLEU

- Example
  - baseline: ... zur zwischenstaatlichen methoden ...
  - factored model: ... zu zwischenstaatlichen methoden ...
- Example
  - baseline: ... das zweite wichtige änderung ...
  - factored model: ... die zweite wichtige änderung ...

Morphological generation model



- Our motivating example
- Translating lemma and morphological information more robust

Initial results

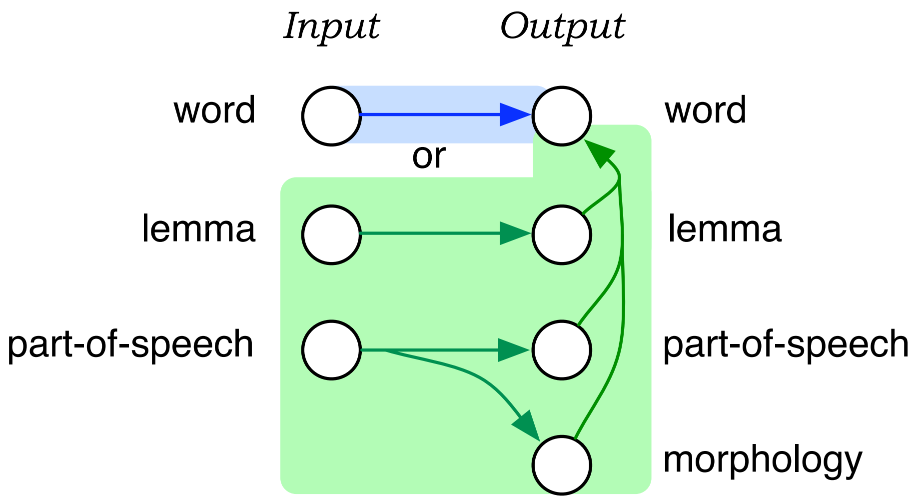
- Results on 1 million word News Commentary corpus (German–English)

System	In-doman	Out-of-domain
Baseline	18.19	15.01
With POS LM	19.05	15.03
Morphgen model	14.38	11.65

- What went wrong?
  - why back-off to lemma, when we know how to translate surface forms?

$\rightarrow$ loss of information

Solution: alternative decoding paths



- Allow both surface form translation and morphgen model
  - prefer surface model for known words
  - morphgen model acts as back-off

Results

- Model now beats the baseline:

System	In-doman	Out-of-domain
Baseline	**18.19**	**15.01**
With POS LM	19.05	15.03
Morphgen model	14.38	11.65
Both model paths	**19.47**	**15.23**

Using POS in reordering

- **Reordering** is often due to syntactic reasons
  - French-English: NN ADJ → ADJ NN
  - Chinese-English: NN1 F NN2 → NN1 NN2
  - Arabic-English: VB NN → NN VB
- Extension of lexicalized reordering model
  - already have model that learns p(monotone|bleue)
  - can be extended to p(monotone|ADJ)
- Gains in preliminary experiments

Other experiments

- Use of CCG supertags on target side
  - Birch et al. [ACL-WS-SMT 2007]
  - Hassan et al. [ACL 2007]
- Handling rich Czech morphology
  - Bojar [ACL WS on SMT, 2007]
- Use of automatic word classes
  - Shen et al. [IWSLT 2006]
- Using POS in reordering
  - Rawlik [UG4 project at U Edinburgh, 2006]
- Additional experiments
  - Report from JHU Summer Workshop 2006

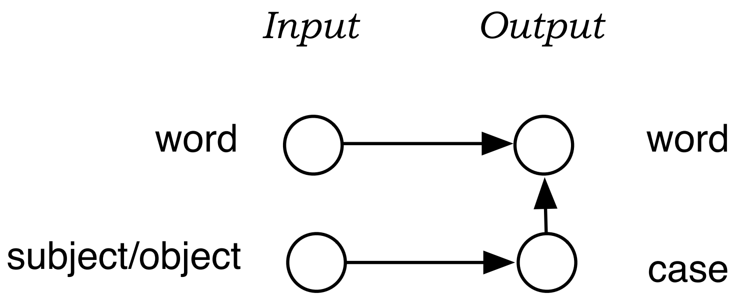
Factored Translation Models

- Motivation
- Example
- Model and Training
- Decoding
- Experiments
- **Outlook**

Adding annotation to the source

- Source words may **lack sufficient information** to map phrases
  - English-German: what case for noun phrases?
  - Chinese-English: plural or singular
  - pronoun translation: what do they refer to?
- Idea: **add additional information** to the source that makes the required information available locally (where it is needed)

Case information for English–German



- Detect in English, if noun phrase is subject/object (using parse tree)
- Map information into case morphology of German
- Use case morphology to generate correct word form

Long range agreement

- Lexical n-gram language model would prefer

the paintings of the old man is beautiful

old man is is a **better trigram** than old man are

- Correct translation

the	**paintings**	**of**	**the**	**old**	**man**	**are**	**beautiful**
-	SBJ-plural	-	-	-	-	V-plural	-

- **Special tag** that tracks *count* of *subject* and *verb*
  p(-,SBJ-plural,-,-,-,-,V-plural,-) $>$ p(-,SBJ-plural,-,-,-,-,V-singular,-)

Shallow syntactic features

the	**paintings**	**of**	**the**	**old**	**man**	**are**	**beautiful**
-	plural	-	-	-	singular	plural	-
B-NP	I-NP	B-PP	I-PP	I-PP	I-PP	V	B-ADJ
SBJ	SBJ	OBJ	OBJ	OBJ	OBJ	V	ADJ

- Shallow syntactic tasks have been formulated as sequence labeling tasks
  - base noun phrase chunking
  - syntactic role labeling

Long range reordering

- **Long range** reordering
  - movement often not limited to local changes
  - German-English: SBJ AUX OBJ V → SBJ AUX V OBJ
- **Asynchronous** models
  - some factor mappings (POS, syntactic chunks) may have longer scope than others (words)
  - larger mappings form template for shorter mappings
  - computational problems with this

Conclusions

- Framework for integration additional annotation
  - integrated in model and search
- Improvements shown with low-level syntactic markup
  - POS, morphology
  - word classes [Shen et al., 2006], CCG [Birch et al., 2007]
- Implemented in open source Moses decoder
  - try it yourself!

Factored models: open questions

- Same **phrase segmentation** for all translation steps?
- Better parameter **estimation** (too many features for MERT?)
- **Other decoding steps** besides phrase translation and word generation (for instance alignment templates)?
- Integration of simple **tools** such as morphological analyzers/generators?
- What **annotation** is useful?
  - translation: mostly lexical, or lemmas for richer statistics, enriching source
  - reordering: syntactic information useful
  - language model: syntactic information for overall grammatical coherence

Discriminative Training

- Evolution from generative to discriminative models
  - IBM Models: purely generative
  - MERT: discriminative training of generative components
  - More features → better discriminative training needed
- Perceptron algorithm
- Problem: overfitting
- Problem: matching reference translation

The birth of SMT: generative models

- The definition of translation probability follows a **mathematical derivation**

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

- Occasionally, some **independence assumptions** are thrown in
for instance IBM Model 1: word translations are independent of each other

$$p(\mathbf{e}|\mathbf{f}, a) = \frac{1}{Z} \prod_i p(e_i|f_{a(i)})$$

- Generative story leads to **straight-forward estimation**
  - maximum likelihood estimation of component probability distribution
  - **EM algorithm** for discovering hidden variables (alignment)

Log-linear models

- IBM Models provided mathematical justification for factoring **components** together

$$p_{LM} \times p_{TM} \times p_D$$

- These may be **weighted**

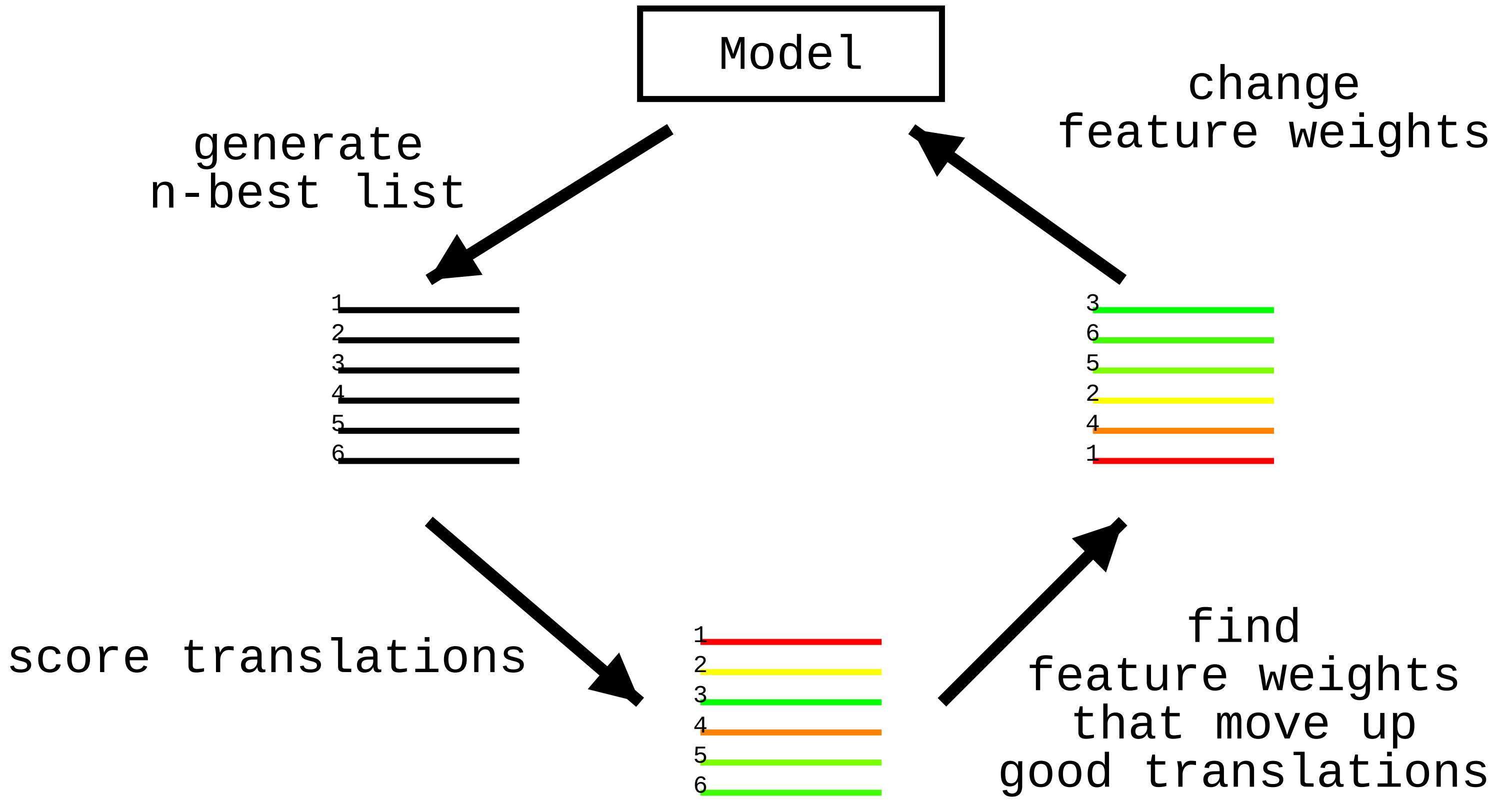
$$p_{LM}^{\lambda_{LM}} \times p_{TM}^{\lambda_{TM}} \times p_D^{\lambda_D}$$

- **Many components** p_i with weights λ_i

$$\prod_i p_i^{\lambda_i} = exp(\sum_i \lambda_i log(p_i))$$

$$log \prod_i p_i^{\lambda_i} = \sum_i \lambda_i log(p_i)$$

Discriminative training



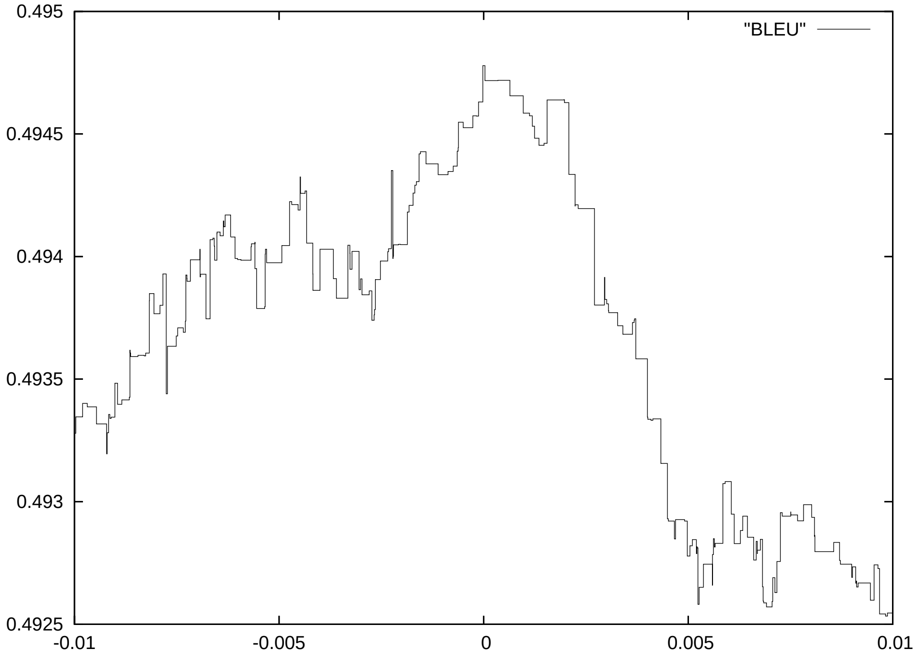
Och's minimum error rate training (MERT)

- **Line search** for best feature weights

```
given:  sentences with n-best list of
translations
iterate n times
    randomize starting feature weights
        iterate until convergences
            for each feature
                find best feature weight
                update if different from current
return best feature weights found in any
iteration
```

BLEU error surface

- Varying one parameter: a rugged line with many local optima



Unstable outcomes: weights vary

component	run 1	run 2	run 3	run 4	run 5	run 6
distance	0.059531	0.071025	0.069061	0.120828	0.120828	0.072891
lexdist 1	0.093565	0.044724	0.097312	0.108922	0.108922	0.062848
lexdist 2	0.021165	0.008882	0.008607	0.013950	0.013950	0.030890
lexdist 3	0.083298	0.049741	0.024822	-0.000598	-0.000598	0.023018
lexdist 4	0.051842	0.108107	0.090298	0.111243	0.111243	0.047508
lexdist 5	0.043290	0.047801	0.020211	0.028672	0.028672	0.050748
lexdist 6	0.083848	0.056161	0.103767	0.032869	0.032869	0.050240
lm 1	0.042750	0.056124	0.052090	0.049561	0.049561	0.059518
lm 2	0.019881	0.012075	0.022896	0.035769	0.035769	0.026414
lm 3	0.059497	0.054580	0.044363	0.048321	0.048321	0.056282
ttable 1	0.052111	0.045096	0.046655	0.054519	0.054519	0.046538
ttable 1	0.052888	0.036831	0.040820	0.058003	0.058003	0.066308
ttable 1	0.042151	0.066256	0.043265	0.047271	0.047271	0.052853
ttable 1	0.034067	0.031048	0.050794	0.037589	0.037589	0.031939
phrase-pen.	0.059151	0.062019	-0.037950	0.023414	0.023414	-0.069425
word-pen	-0.200963	-0.249531	-0.247089	-0.228469	-0.228469	-0.252579

Unstable outcomes: scores vary

- Even different scores with different runs (varying 0.40 on dev, 0.89 on test)

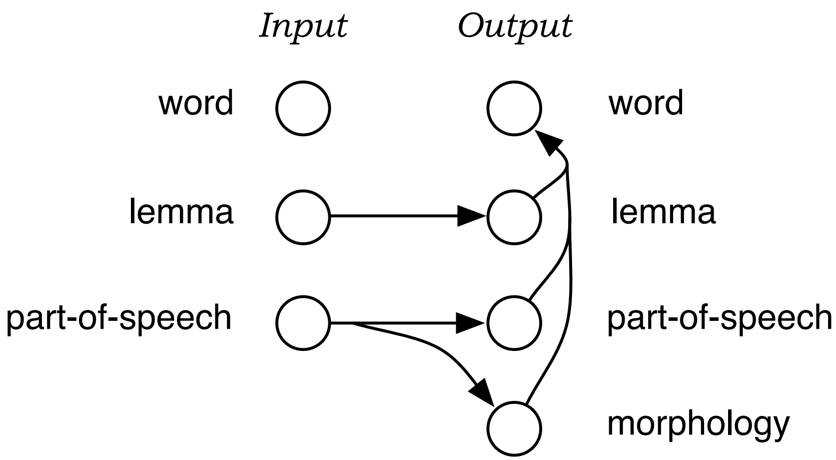
run	iterations	dev score	test score
1	8	50.16	51.99
2	9	50.26	51.78
3	8	50.13	51.59
4	12	50.10	51.20
5	10	50.16	51.43
6	11	50.02	51.66
7	10	50.25	51.10
8	11	50.21	51.32
9	10	50.42	51.79

More features: more components

- We would like to add **more components** to our model
  - multiple language models
  - domain adaptation features
  - various special handling features
  - using linguistic information

→ MERT becomes even **less reliable**
  - runs many more iterations
  - fails more frequently

More features: factored models



- Factored translation models break up phrase mapping into smaller steps
  - multiple translation tables
  - multiple generation tables
  - multiple language models and sequence models on factors

→ **Many more features**

Millions of features

- Why **mix** of discriminative training and generative models?
- Discriminative training of all components
  - phrase table [Liang et al., 2006]
  - language model [Roark et al, 2004]
  - additional features
- **Large-scale** discriminative training
  - millions of features
  - training of full training set, not just a small development corpus

Perceptron algorithm

- Translate each sentence
- If no match with reference translation: update features

```
set all lambda = 0
do until convergence
    for all foreign sentences f
        set e-best to best translation according to model
        set e-ref to reference translation
        if e-best != e-ref
            for all features feature-i
                lambda-i += feature-i(f,e-ref)
                          - feature-i(f,e-best)
```

Problem: overfitting

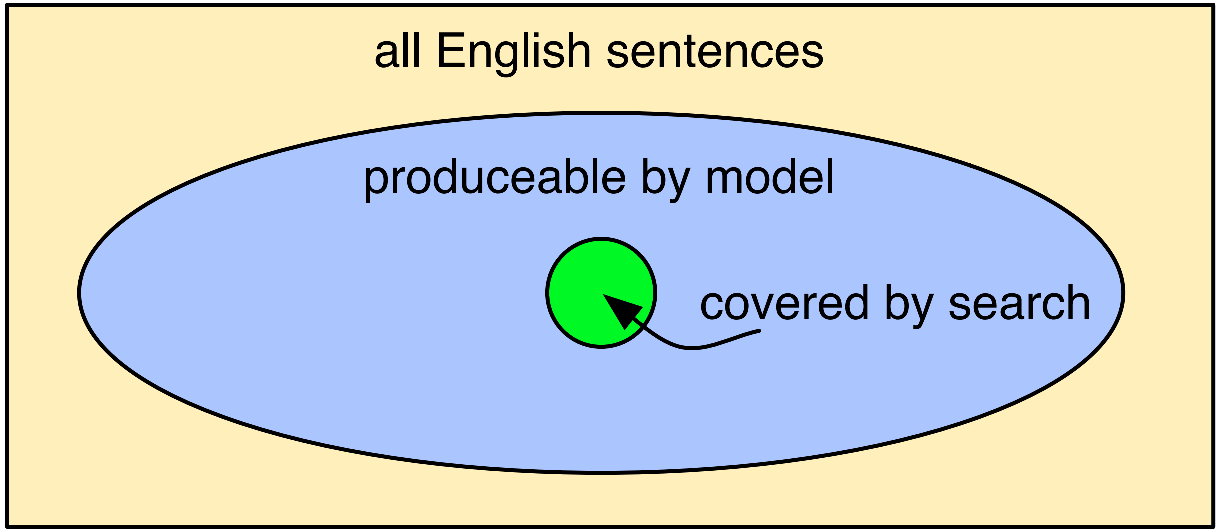
- Fundamental problem in machine learning
  - what works best for training data, may not work well in general
  - **rare, unrepresentative features** may get too much weight
- **Especially severe problem** in phrase-based models
  - **long phrase pairs** explain well *individual sentences*
  - ... but are less general, *suspect to noise*
  - EM training of phrase models [Marcu and Wong, 2002] has same problem

Solutions

- **Restrict to short phrases**, e.g., maximum 3 words (current approach)
  - limits the power of phrase-based models
  - ... but not very much [Koehn et al, 2003]
- **Jackknife**
  - collect phrase pairs from one part of corpus
  - optimize their feature weights on another part
- IBM direct model: **only one-to-many** phrases [Ittycheriah and Salim Roukos, 2007]

Problem: reference translation

- Reference translation may be anywhere in this box



- If produceable by model → we can compute feature scores
- If not → we can not

Some solutions

- **Skip sentences**, for which reference can not be produced
  - invalidates large amounts of training data
  - biases model to shorter sentences
- Declare candidate translations closest to reference as **surrogate**
  - closeness measured for instance by smoothed BLEU score
  - may be not a very good translation: odd feature values, training is severely distorted

Conclusions

- Currently have proof-of-concept implementation
- Future work: Overcome various technical challenges
  - reference translation may not be produceable
  - overfitting
  - mix of binary and real-valued features
  - scaling up
- More and more features are unavoidable, let's deal with them