Machine Translation Using Statistical Modeling

Herman Ney, and F. J. Och
Aachen University of Technology, Germany

Reference: Pattern Recognition in Speech and Language Processing, Ch11

Presented by Shih-Hung Liu

Outline

- Introduction
- Statistical decision theory and linguistics
- Alignment and lexicon models
  - HMM
  - IBM model 1-5
  - Search
- Alignment templates
  - From single words to word group
- Speech translation
  - The integrated approach
- Experimental results
- Summary

Introduction

Automatic translation of language or
Machine translation

written language or text input

Speech translation

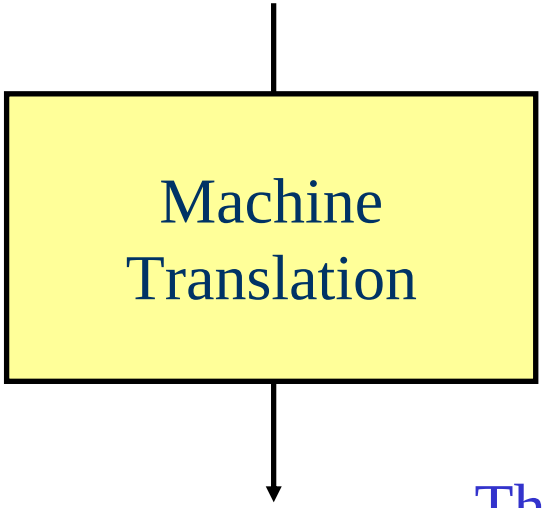
spontaneous spoken speech input

Bayes Decision Rule for Written Language Translation

- The Statistical Approach
  - Speech Recognition = Acoustic-Linguistic Modeling
    + Statistical Decision Theory
  - Machine Translation = Linguistic Modeling
    + Statistical Decision Theory
  - Advantages in using probability distributions
    - The probabilities are directly used as scores.
    - It is straightforward to combine scores.
    - Weak and vague dependences can be modeled easily.

Bayes Decision Rule for Written Language Translation

$$f_1^J = f_1 \cdots f_j \cdots f_J$$



$$e_1^I = e_1 \cdots e_i \cdots e_I$$

The string translation model

$$\hat{e}_1^I = \arg\max_{e_1^I} \{\Pr(e_1^I \mid f_1^J)\} = \arg\max_{e_1^I} \{\Pr(e_1^I)\Pr(f_1^J \mid e_1^I)\}$$

Language model of the target language

Flow chart

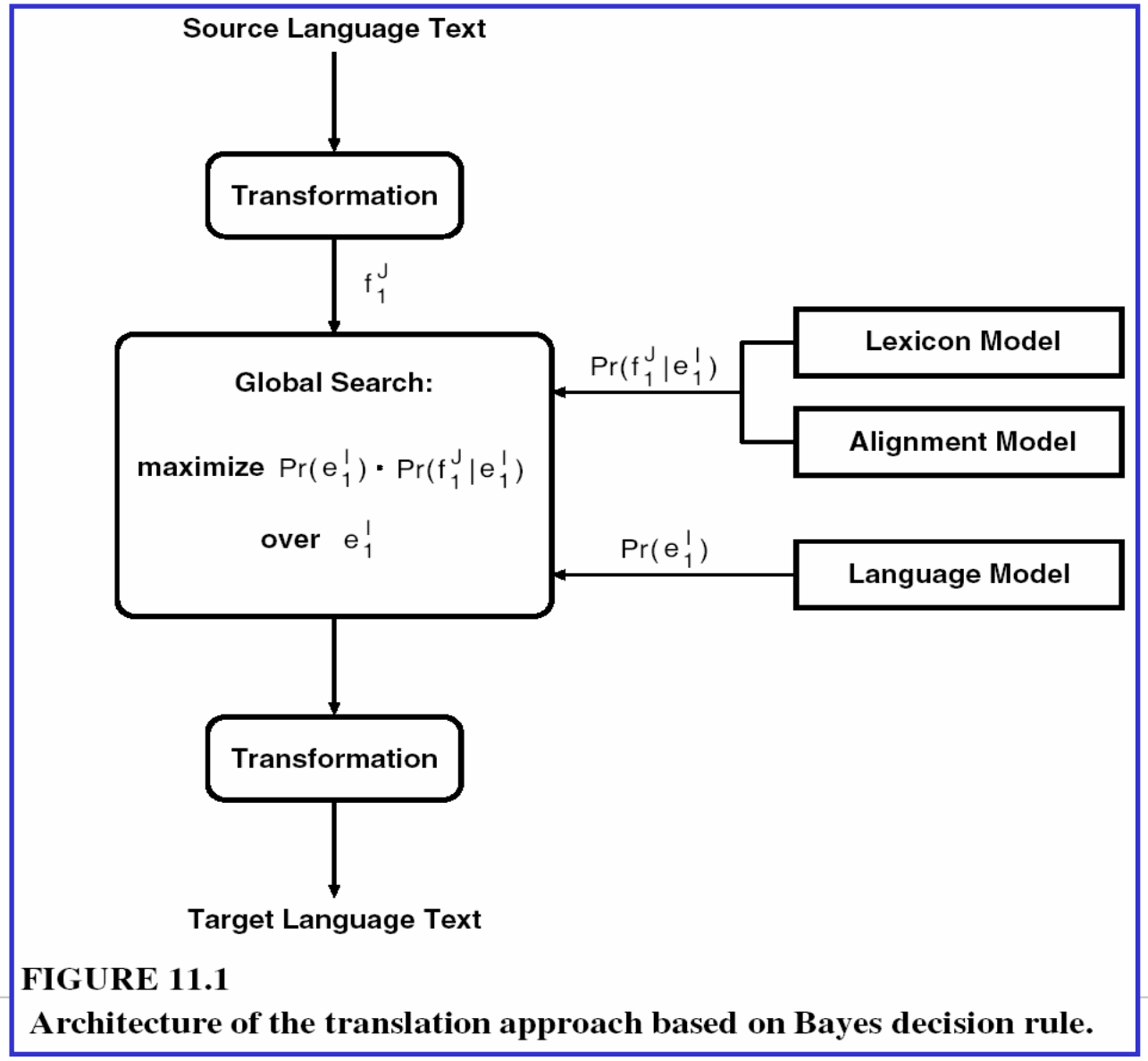


FIGURE 11.1
Architecture of the translation approach based on Bayes decision rule.

Alignment and Lexicon Models

- Model the string translation probability $\Pr(f_1^J \mid e_1^I)$
- We constrain this model by assigning each source word to exactly one target word. e_1^I
- Two approaches to alignment modeling are in more detail
  - HMM
  - IBM 1-5

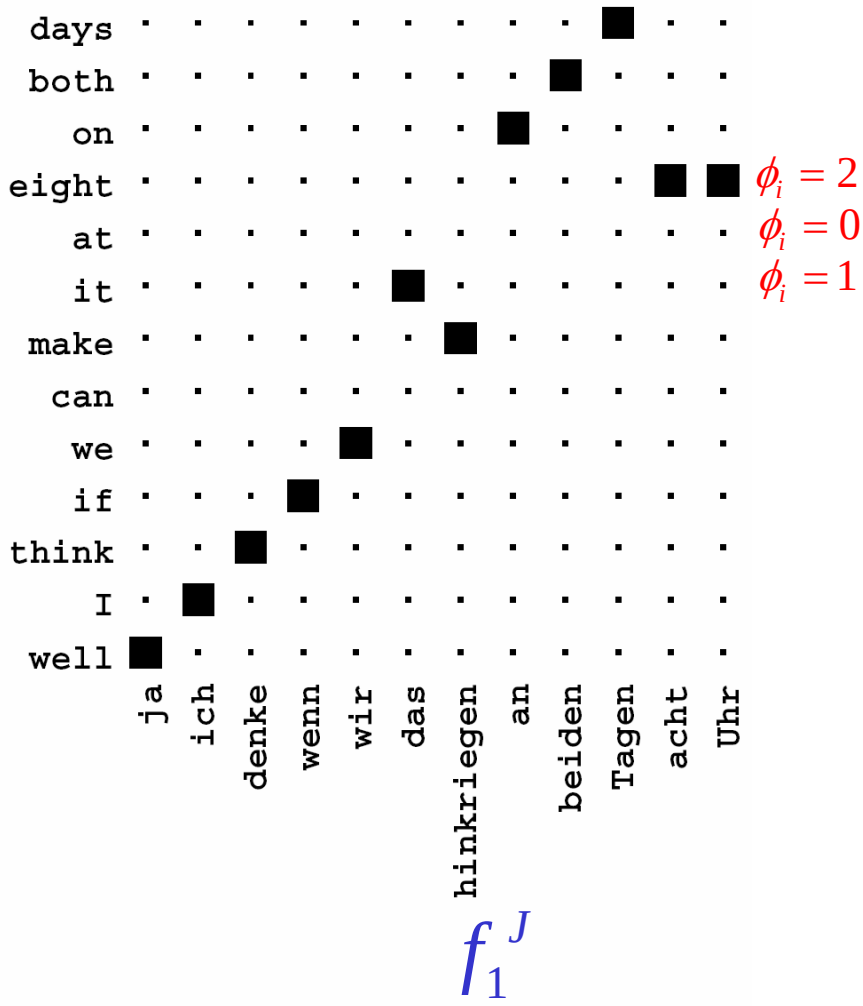


FIGURE 11.2
Example of an alignment for a German-English sentence pair.

HMM

$f_1^J \xrightarrow{a_1^J} e_1^I$

$a_1^J := a_1 \cdots a_j \cdots a_J$ is an alignment

$$\Pr(f_1^J \mid e_1^I) = \sum_{a_1^J} \Pr(f_1^J, a_1^J \mid e_1^I)$$

Length model

$$\Pr(f_1^J, a_1^J \mid e_1^I) = \frac{\Pr(f_1^J, a_1^J, e_1^I)}{\Pr(e_1^I)} = \frac{\Pr(J, e_1^I)}{\Pr(e_1^I)} \frac{\Pr(f_1^J, a_1^J, e_1^I)}{\Pr(J, e_1^I)} = \boxed{\Pr(J \mid e_1^I)} \Pr(f_1^J, a_1^J \mid e_1^I, J)$$

$$\Pr(f_1^J, a_1^J \mid e_1^I, J) = \Pr(f_1 \cdots f_J, a_1 \cdots a_J \mid e_1^I, J)$$

$$= \Pr(f_1, a_1 \mid e_1^I, J) \frac{\Pr(f_1 f_2, a_1 a_2 \mid e_1^I, J)}{\Pr(f_1, a_1 \mid e_1^I, J)} \frac{\Pr(f_1 \cdots f_3, a_1 \cdots a_3 \mid e_1^I, J)}{\Pr(f_1 f_2, a_1 a_2 \mid e_1^I, J)} \cdots \frac{\Pr(f_1 \cdots f_J, a_1 \cdots a_J \mid e_1^I, J)}{\Pr(f_1 \cdots f_{J-1}, a_1 \cdots a_{J-1} \mid e_1^I, J)}$$

$$= \Pr(f_1, a_1 \mid e_1^I, J) \Pr(f_2, a_2 \mid a_1, f_1, e_1^I, J) \Pr(f_3, a_3 \mid f_1 f_2, a_1 a_2, e_1^I, J) \cdots \Pr(f_J, a_J \mid f_1 \cdots f_{J-1}, a_1 \cdots a_{J-1}, e_1^I, J)$$

$$= \Pr(f_1, a_1 \mid e_1^I, J) \Pr(f_2, a_2 \mid a_1, f_1, e_1^I, J) \Pr(f_3, a_3 \mid f_1^2, a_1^2, e_1^I, J) \cdots \Pr(f_J, a_J \mid f_1^{J-1}, a_1^{J-1}, e_1^I, J)$$

$$= \prod_{j=1}^{J} \Pr(f_j, a_j \mid a_1^{j-1}, f_1^{j-1}, e_1^I, J)$$

$$\Pr(f_1^J \mid e_1^I) = \sum_{a_1^J} \Pr(J \mid e_1^I) \prod_{j=1}^{J} \Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, e_1^I, J) \Pr(f_j \mid a_1^j, f_1^{j-1}, e_1^I, J)$$

$$= \prod_{j=1}^{J} \frac{\Pr(f_j, a_j, a_1^{j-1}, f_1^{j-1}, e_1^I, J)}{\Pr(a_1^{j-1}, f_1^{j-1}, e_1^I, J)}$$

$$= \prod_{j=1}^{J} \frac{\Pr(f_j, a_j, a_1^{j-1}, f_1^{j-1}, e_1^I, J)}{\Pr(a_j, a_1^{j-1}, f_1^{j-1}, e_1^I, J)} \frac{\Pr(a_j, a_1^{j-1}, f_1^{j-1}, e_1^I, J)}{\Pr(a_1^{j-1}, f_1^{j-1}, e_1^I, J)}$$

$$= \prod_{j=1}^{J} \Pr(f_j \mid a_j, a_1^{j-1}, f_1^{j-1}, e_1^I, J) \Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, e_1^I, J)$$

$$= \prod_{j=1}^{J} \boxed{\Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, e_1^I, J) \Pr(f_j \mid a_1^j, f_1^{j-1}, e_1^I, J)}$$

Alignment model

Lexicon model

HMM

- Three cases are considered.

	HMM Type		
	baseline HMM	homogeneous HMM	context dependent HMM
Length model $\Pr(J \mid e_1^I)$	$p(J \mid I)$	$p(J \mid I)$	$p(J \mid I)$
Alignment model $\Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, e_1^I, J)$	$p(a_j \mid a_{j-1}, I, J)$	$p(a_j \mid a_{j-1}, I, J) := \frac{q(a_j - a_{j-1})}{\sum_{i=1}^{I} q(i - a_{j-1})}$	$p(a_j \mid a_{j-1}, f_{j-1}, I, J)$
Lexicon model $\Pr(f_j \mid f_1^{j-1}, a_1^j, e_1^I, J)$	$p(f_j \mid e_{a_j})$	$p(f_j \mid e_{a_j})$	$p(f_j \mid f_{j-1}, e_{a_{j-1}}, e_{a_j})$

HMM

- Search strategy for baseline HMM

$$\max_{e_1^I} \Pr(e_1^I \mid f_1^J) \approx \max_{e_1^I} \Pr(f_1^J \mid e_1^I) \Pr(e_1^I)$$

$$= \max_{e_1^I} \max_{a_1^J} \Pr(f_1^J, a_1^J \mid e_1^I) \Pr(e_1^I)$$

$$= \max_{e_1^I} \max_{a_1^J} \Pr(J \mid e_1^I) \Pr(f_1^J, a_1^J \mid e_1^I, J) \Pr(e_1^I)$$

$$= \max_{e_1^I} \Pr(J \mid e_1^I) \max_{a_1^J} \left(\prod_{j=1}^{J} \Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, e_1^I, J) \Pr(f_j \mid a_1^j, f_1^{j-1}, e_1^I, J) \right) \Pr(e_1^I)$$

$$= \max_{I} p(J \mid I) \max_{e_1^I, a_1^J} \left(\prod_{j=1}^{J} p(a_j \mid a_{j-1}, I, J) p(f_j \mid e_{a_j}) \right) \left(\prod_{i=1}^{I} p(e_i \mid e_{i-2}^{i-1}) \right)$$

DP: $O(I^2 \times J \times V_e^2)$ … Roger opinion

Models IBM 1-5

- Model IBM-1 and IBM-2: zero-order dependence

	Type	
	baseline HMM	IBM-1 and IBM-2
Length model $\Pr(J \mid e_1^I)$	$p(J \mid I)$	$p(J \mid I)$
Alignment model $\Pr(a_j \mid a_1^{j-1}, f_1^{j-1}, e_1^I, J)$	$p(a_j \mid a_{j-1}, I, J)$	$p(a_j \mid j, I, J)$ absolute position
Lexicon model $\Pr(f_j \mid f_1^{j-1}, a_1^j, e_1^I, J)$	$p(f_j \mid e_{a_j})$	$p(f_j \mid e_{a_j})$

Models IBM 1-5

- Model IBM-1 and IBM-2: zero-order dependence

$$\Pr(f_1^J \mid e_1^I) = \sum_{a_1^J} \Pr(f_1^J, a_1^J \mid e_1^I) = \sum_{a_1^J} \Pr(J \mid e_1^I) \prod_{j=1}^{J} \left[\Pr(a_j \mid a_1^{j-1}, f_1^{i-1}, e_1^I, J) \cdot \Pr(f_j \mid f_1^{j-1}, a_1^j, e_1^I, J)\right]$$

$$= \sum_{a_1^J} p(J \mid I) \prod_{j=1}^{J} \left[p(a_j \mid j, I, J) \cdot p(f_j \mid e_{a_j})\right] = p(J \mid I) \sum_{a_1^J} \prod_{j=1}^{J} \left[p(a_j \mid j, I, J) \cdot p(f_j \mid e_{a_j})\right]$$

$$= p(J \mid I) \sum_{a_1} \cdots \sum_{a_J} \prod_{j=1}^{J} \left[p(a_j \mid j, I, J) \cdot p(f_j \mid e_{a_j})\right]$$

$$= p(J \mid I) \sum_{a_1} \cdots \sum_{a_J} \left\{\left[p(a_1 \mid 1, I, J) \cdot p(f_1 \mid e_{a_1})\right] \cdots \left[p(a_j \mid j, I, J) \cdot p(f_j \mid e_{a_j})\right] \cdots \left[p(a_J \mid J, I, J) \cdot p(f_J \mid e_{a_J})\right]\right\}$$

$$= p(J \mid I) \sum_{a_1} \left[p(a_1 \mid 1, I, J) \cdot p(f_1 \mid e_{a_1})\right] \cdots \sum_{a_j} \left[p(a_j \mid j, I, J) \cdot p(f_j \mid e_{a_j})\right] \cdots \sum_{a_J} \left[p(a_J \mid J, I, J) \cdot p(f_J \mid e_{a_J})\right]$$

$$= p(J \mid I) \left(\sum_{a_1} \left[p(a_1 \mid 1, I, J) \cdot p(f_1 \mid e_{a_1})\right]\right) \cdots \left(\sum_{a_j} \left[p(a_j \mid j, I, J) \cdot p(f_j \mid e_{a_j})\right]\right) \cdots \left(\sum_{a_J} \left[p(a_J \mid J, I, J) \cdot p(f_J \mid e_{a_J})\right]\right)$$

$$= p(J \mid I) \prod_{j=1}^{J} \left(\sum_{a_j} \left[p(a_j \mid j, I, J) \cdot p(f_1 \mid e_{a_j})\right]\right) = p(J \mid I) \prod_{j=1}^{J} \left(\sum_{i} \left[p(i \mid j, I, J) \cdot p(f_1 \mid e_i)\right]\right)$$

$$= p(J \mid I) \prod_{j=1}^{J} \sum_{i=1}^{I} \left[p(i \mid j, I, J) \cdot p(f_1 \mid e_i)\right]$$

mixture weight as alignment probabilities

component distribution as lexicon probabilities

Models IBM 1-5

- Model IBM-1 and IBM-2: zero-order dependence
  - The model IBM-1 is a special case with a uniform alignment probability

$$p(i \mid j, I, J) = \frac{1}{I}$$

$$(\text{models } IBM-1)$$

  - The 'empty word' is added to the target sentence e_1^I to allow for source words which have no direct counterpart in the target sentence f_1^J .
  - Formally, the concept of the empty word is incorporated into the alignment models by adding the empty word e_0 at position $i = 0$ to the target sentence e_1^I and aligning all source words f_j without a direct translation to this empty word.

Models IBM 1-5

- Model IBM-3: fertility concept
  - For each target word e, there is a probability distribution over its possible fertilities ϕ :

$$p(\phi \mid e)$$

  - Experimentally, we observe that the fertilities on values from 0 to 4.

$$\text{fertility}: \quad \phi_i := \sum_j \delta(a_j, i)$$

  - Using this equation, we can start with an HMM or model IBM-2 and then compute initial values for the fertilities.
  - The fertility concept can be used to better model target words having no counterpart in the source sentence, i.e. $\phi_i = 0$

Models IBM 1-5

- Model IBM-4 and IBM-5: inverted alignments with first-order dependence
  - We assume that the probability distribution $\Pr(f_1^J, a_1^J \mid e_1^J)$ is the result of a process consisting of three steps.
    - Select a fertility ϕ_i for each hypothesized target word
    - For each target word e_i, we generate the set of associated ϕ_i source words f according to the fertility ϕ_i, where the (final) positions are not specified yet.
    - The source words are permute so that the observed sequence f_1^J is produced.
  - Inverted alignment:

$$e_1^I \xrightarrow{b_1^I} f_1^J$$

$$b_1^I := b_1 \cdots b_i \cdots b_I \text{ is an inveted alignment}$$

Models IBM 1-5

– Refinements

- We must take into account that the fertility of word e_i in position may be different from 1, e.g., for a fertility larger than 1, several i positions on the target axis j have to be produced.

- The dependence on b_{i-1} does not use the absolute positions, but only relative positions.

- To reduce the number of free parameters, the dependence on the words f_{b_i} and e_{i-1} is replaced by a dependence on the corresponding parts-of-speech or word classes $G(f_{b_i})$ and $G(e_{i-1})$:

$$p\left(b_i \mid b_{i-1}, G(f_{b_i}), G(e_{i-1})\right)$$

Search

- We use inverted alignments as in the model IBM-4 which define a mapping from target to source positions rather than the other way round.
- We allow several positions in the source language to be covered, i.e. we consider mappings B of the form: $B : i \rightarrow B_i \subset \{1, \cdots, j, \cdots, J\}$
- For this inverted alignment mapping with sets B_i of source positions, we again assume a sort of first-order model:

$$p(B_i \mid B_{i-1}, e_{i-1})$$

where we dropped the dependence on I and J

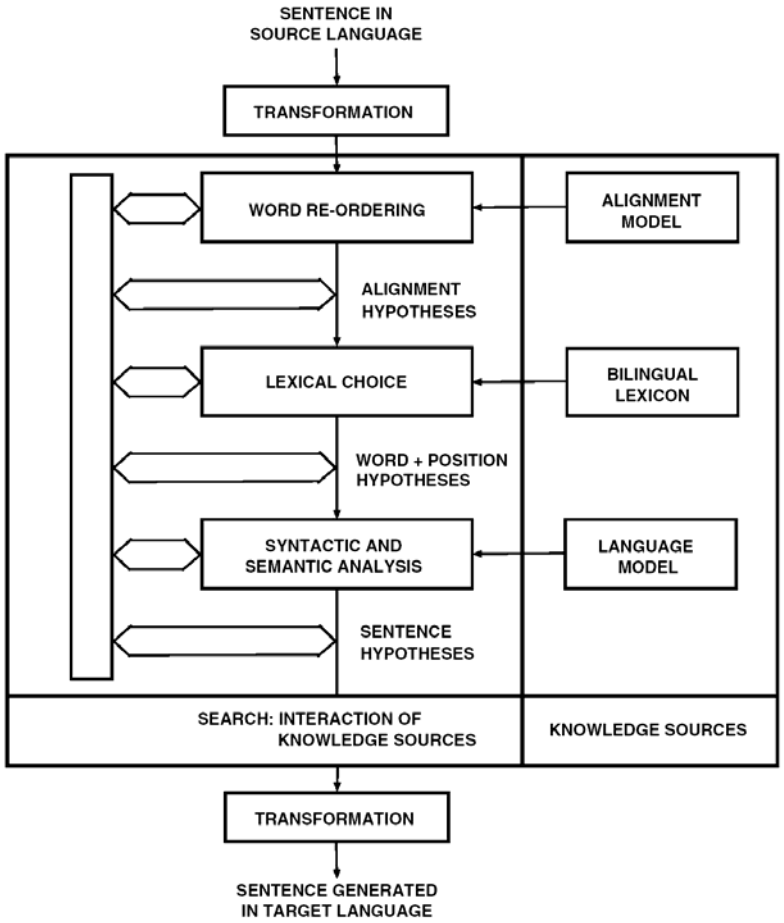


FIGURE 11.3
Illustration of search in statistical translation.

Search

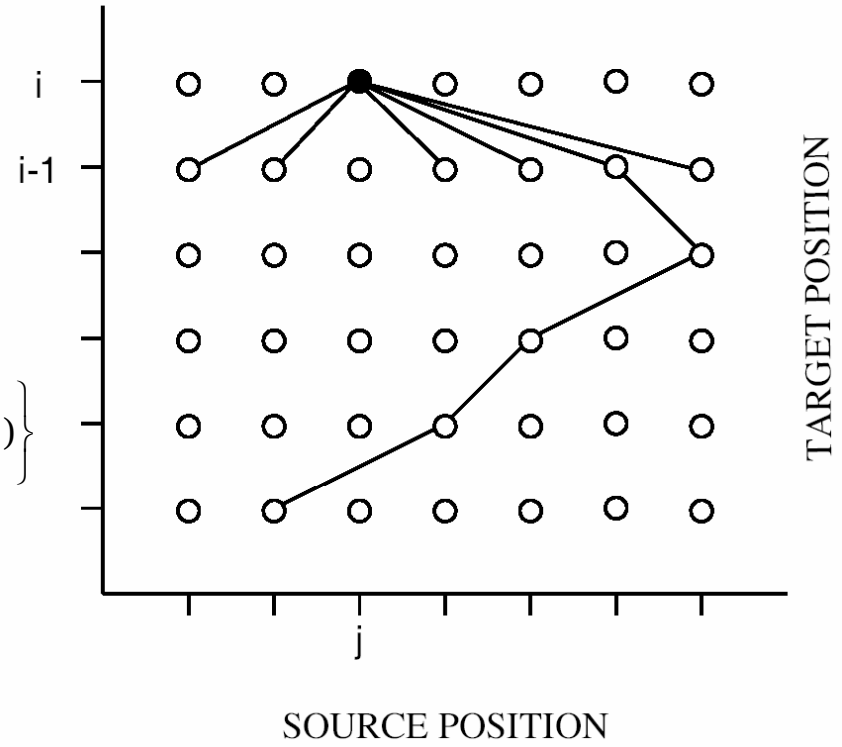
- We replace the sum over all alignments by the best alignment, which is referred to as maximum approximation in speech recognition.

- Using a trigram language model $p(e_i \mid e_{i-2}, e_{i-1})$, we obtain the following search criterion:

$$\max_{I}\left\{ p(J \mid I) \cdot \max_{B_1^I, e_1^I} \prod_{i=1}^{I} \left(p(e_i \mid e_{i-2}^{i-1}) \cdot p(B_i \mid B_{i-1}, e_{i-1}) \cdot \prod_{j \in B_i} p(f_j \mid e_i) \right) \right\}$$

- We can see that we can build up hypotheses of partial target sentences in a bottom-to-top strategy over the position i of the target sentence e_1^i .

- Beam search is used to handle the huge search space.



Constrain: all positions of the source sentence should be covered exactly once.

FIGURE 11.4
Illustration of bottom-to-top search.

Search

$$\max_{e_1^I} \Pr(e_1^I \mid f_1^J) \approx \max_I p(J \mid I) \max_{e_1^I, a_1^J} \left(\prod_{j=1}^{J} p(a_j \mid a_{j-1}, I, J) p(f_j \mid e_{a_j}) \right) \left(\prod_{i=1}^{I} p(e_i \mid e_{i-2}^{i-1}) \right)$$

$$\max_{e_1^I} \Pr(e_1^I \mid f_1^J) \approx \max_I p(J \mid I) \max_{e_1^I, B_1^I} \left(\prod_{i=1}^{I} p(B_i \mid B_{i-1}, e_{i-1}) \right) \left(\prod_{j=1}^{J} p(f_j \mid e_{a_j}) \right) \left(\prod_{i=1}^{I} p(e_i \mid e_{i-2}^{i-1}) \right)$$

$$= \max_I p(J \mid I) \max_{e_1^I, B_1^I} \left(\prod_{i=1}^{I} p(e_i \mid e_{i-2}^{i-1}) p(B_i \mid B_{i-1}, e_{i-1}) \right) \left(\prod_{j=1}^{J} p(f_j \mid e_{a_j}) \right)$$

$$= \max_I p(J \mid I) \max_{e_1^I, B_1^I} \left(\prod_{i=1}^{I} p(e_i \mid e_{i-2}^{i-1}) p(B_i \mid B_{i-1}, e_{i-1}) \prod_{j \in B_i}^{J} p(f_j \mid e_i) \right)$$

Algorithmic Differences between Speech Recognition and Language Translation

- Monotonicity
  - In speech Recognition, there is a strict monotonicity between the sequence of acoustic vectors and the sequence of recognized words or phonemes.
  - This is not the case for machine translation, and therefore the search problem becomes more complicated.
- Fertility
  - In machine translation, we have to decide whether a word is present in the target string or not. Therefore, it is important to assign a fertility to each word of the target vocabulary.
  - In speech recognition, the counterpart of a word is an HMM state. However, we never take decisions about states, but about whole phoneme models. There fore the concept of fertility is not really needed in speech recognition.

Alignment Templates: From Single Words to Word Groups

- We extend the approach to handle word groups or phrases rather than single words.
- A whole group of adjacent words in the source sentence may be aligned with a whole group of adjacent words in the target language.

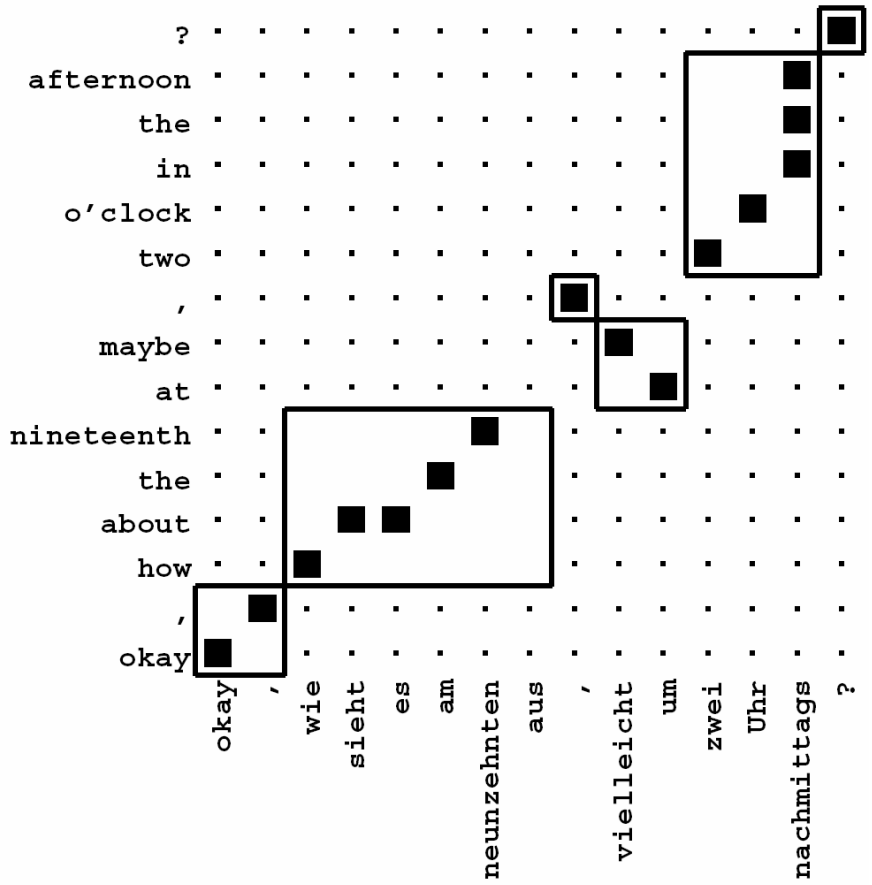


FIGURE 11.5
Example of alignment templates for a German-English sentence pair

Alignment Templates: From Single Words to Word Groups

- We first decompose both the source sentence f_1^J and the target sentence e_1^I into a sequence of word groups.

$$f_1^J = \tilde{f}_1^K \ , \ \tilde{f}_k = f_{j_{k-1}+1}, \cdots, f_{j_k} \ , \ k = 1, \cdots, K$$

$$e_1^I = \tilde{e}_1^K \ , \ \tilde{e}_k = e_{i_{k-1}+1}, \cdots, e_{i_k} \ , \ k = 1, \cdots, K$$

- Then the alignment between word groups.

$$\begin{aligned}
\Pr(f_1^J \mid e_1^I) &= \Pr(\tilde{f}_1^K \mid \tilde{e}_1^K) \\
&= \sum_{\tilde{a}_1^K} \Pr(\tilde{a}_1^K, \tilde{f}_1^K \mid \tilde{e}_1^K) \\
&= \sum_{\tilde{a}_1^K} \Pr(\tilde{a}_1^K \mid \tilde{e}_1^K) \cdot \Pr(\tilde{f}_1^K \mid \tilde{a}_1^K, \tilde{e}_1^K) \\
&= \sum_{\tilde{a}_1^K} \prod_{k=1}^{K} p(\tilde{a}_k \mid \tilde{a}_{k-1}, K) \cdot p(\tilde{f}_k \mid \tilde{e}_{\tilde{a}_k})
\end{aligned}$$

alignment within word group

Alignment Templates: From Single Words to Word Groups

- We introduce a new hidden variable z which will be referred to as alignment template.

$$p(\tilde{f} \mid \tilde{e}) = \sum_{z} p(z \mid \tilde{e}) \cdot p(\tilde{f} \mid z, \tilde{e})$$

- The probability $p(z \mid \tilde{e})$ and $p(\tilde{f} \mid z, \tilde{e})$ are determined using the aligned training corpus and are set to zero if the triple $(\tilde{f}, \tilde{e}, z)$ did not occur in the training corpus. If the triple did occur in the training corpus, we use the following model for $p(\tilde{f} \mid z, \tilde{e})$

$$p(\tilde{f} \mid z, \tilde{e}) = \prod_{j=1}^{J'} \sum_{i=1}^{I'} p(i \mid j, z) \cdot p(\tilde{f}_j \mid \tilde{e}_i)$$

$$\text{where} \quad p(i \mid j, z) = \frac{z_{ij}}{\sum_{i'} z_{i'j}}$$

Search

- To perform the search, we use the following models
  - As language model, we use a class-based *n*-gram (e.g. 3- or 5- gram) language model with backing-off. Typically, this is slightly better than the standard bigram language model.
  - We assume that all possible segmentations have the same probability.
  - The alignment model at the template level is an HMM-type alignment model. Obviously, as usual, all words in the source string must be covered.
- We have to allow for all possible segmentations of the source sentence into word groups, for all possible alignments between the word groups and for possible alignments within the word groups.

Experimental Results

- VERBMOBIL
  - Translation of spoken dialogues.
  - In the domains of appointment scheduling and travel planning.
  - A native German speaker and a native English speaker conduct a dialogue where they can only interact by speaking and listening to the VERBMOBIL system.
- Corpus
  - Spoken dialogues were recorded.
  - These dialogues were manually transcribed and later manually translated by VERBMOBIL partners.
  - Each of these so-called dialogue turns may consist of several sentences spoken by the same speaker.
  - There is no one-to-one correspondence between source and target sentences.

Experimental Results

- The turns are split into shorter segments using punctuation marks as potential split points.
- A dynamic programming approach is used to find the optimal segmentation points. (the punctuation marks in source and target sentences are not necessarily identical)

TABLE 11.1
Bilingual training corpus, recognition lexicon and translation lexicon.

		German	English
Training Text	Sentence Pairs	58 073	
	Words	418 979	453 632
	Words + Punct.Marks	519 523	549 921
	Vocabulary	7 940	4 673
	Singletons	44.8%	37.6%
Recognition	Vocabulary	10 157	6 871
Translation	Added Word Pairs	12 779	
	Vocabulary	11 501	6 867

Experimental Results

- Offline Results
  - We briefly report experimental offline results for the following translation approaches:
    - Single-word based approach
    - Alignment template approach
    - Cascaded transducer approach

TABLE 11.2
Comparison of three statistical translation approaches (test on text input: 251 sentences = 2197 words + 430 punctuation marks).

Translation Approach	mWER [%]	SSER [%]
Single-Word Based	38.2	35.7
Alignment Template	36.0	29.0
Cascaded Transducers	>40.0	>40.0

Speech Translation: The Integrated Approach

- Principle

$$x_1^T \rightarrow f_1^J \rightarrow e_1^I$$

feature vectors

$$\arg\max_{e_1^I} \Pr(e_1^I \mid x_1^T) = \arg\max_{e_1^I} \{\Pr(e_1^I) \cdot \Pr(x_1^T \mid e_1^I)\}$$

$$= \arg\max_{e_1^I} \left\{ \Pr(e_1^I) \cdot \sum_{f_1^J} \Pr(f_1^J, x_1^T \mid e_1^I) \right\}$$

$$= \arg\max_{e_1^I} \left\{ \Pr(e_1^I) \cdot \sum_{f_1^J} \Pr(f_1^J \mid e_1^I) \Pr(x_1^T \mid f_1^J, e_1^I) \right\}$$

$$= \arg\max_{e_1^I} \left\{ \Pr(e_1^I) \cdot \sum_{f_1^J} \Pr(f_1^J \mid e_1^I) \Pr(x_1^T \mid f_1^J) \right\}$$

$$= \arg\max_{e_1^I} \left\{ \Pr(e_1^I) \cdot \max_{f_1^J} \{\Pr(f_1^J \mid e_1^I) \Pr(x_1^T \mid f_1^J)\} \right\}$$

Speech Translation: The Integrated Approach

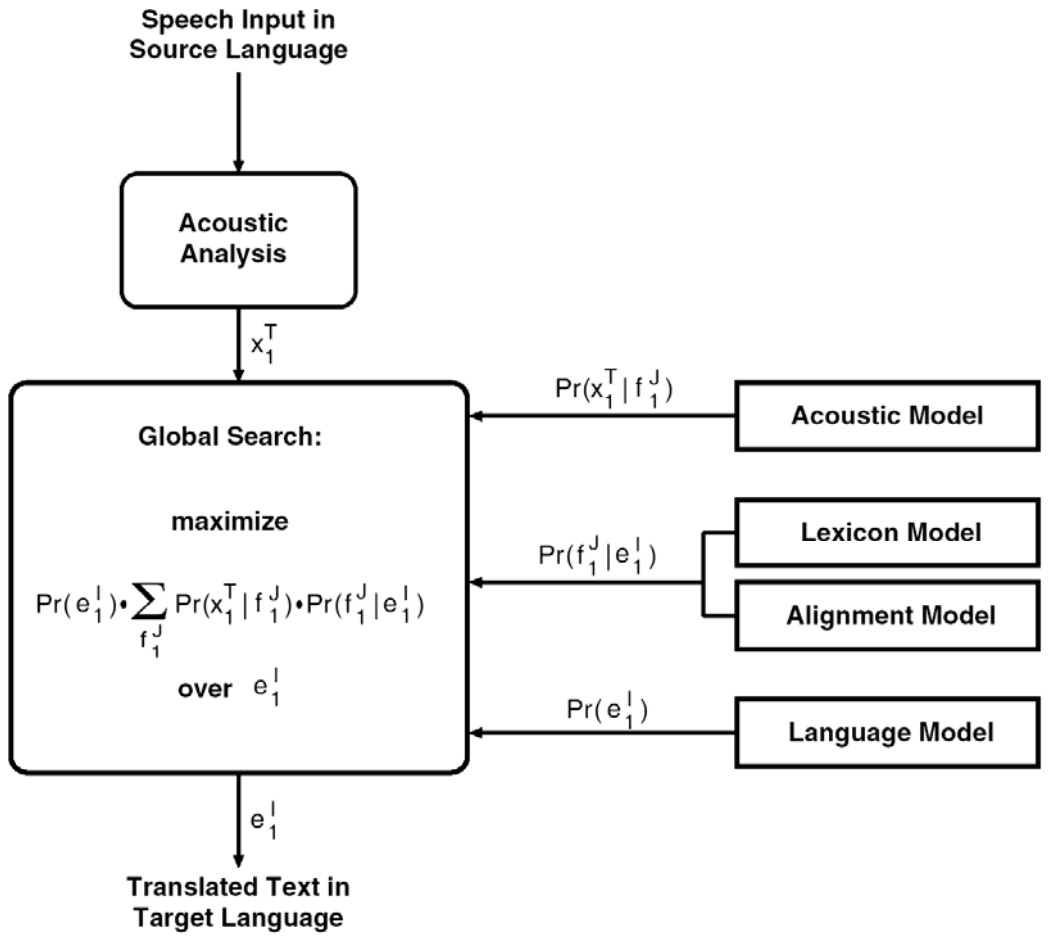


FIGURE 11.6
Integrated architecture of speech translation approach based on Bayes decision rule.

Speech Translation: The Integrated Approach

- Practical Implementation

$$p(f_j \mid f_{j-1}, e_{a_j}) \quad \text{in lieu of} \quad p(f_j \mid e_{a_j})$$

$$\Pr(f_1^J \mid e_1^I) = \sum_{a_1^J} \prod_j \left[p(a_j \mid a_{j-1}, I) \cdot p(f_j \mid f_{j-1}, e_{a_j}) \right]$$

  - For the sake of simplicity, bigram dependence will be used.

$$\Pr(e_1^I) = p(e_i \mid e_{i-1})$$

  - Key Issue
    - The question of how the requirement of having both a well-formed source sentence f_1^J and well-formed target sentence e_1^I at the same time is satisfied.

Speech Translation: The Integrated Approach

- Summary
  - No approaches fully implements the integrated coupling of recognition and translation from a statistical point of view.
  - We consider this integrated approach and its suitable implementation to be an open question for future research on spoken language translation.