

Lecture 18
Natural Language Processing

Marco Chiarandini

Department of Mathematics & Computer Science
University of Southern Denmark

Slides by Dan Klein at Berkeley

Course Overview

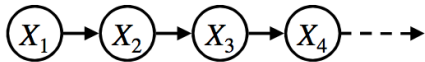
- ✓ Introduction
 - ✓ Artificial Intelligence
 - ✓ Intelligent Agents
- ✓ Search
 - ✓ Uninformed Search
 - ✓ Heuristic Search
- ✓ Uncertain knowledge and Reasoning
 - ✓ Probability and Bayesian approach
 - ✓ Bayesian Networks
 - ✓ Hidden Markov Chains
 - ✓ Kalman Filters
- ✓ Learning
 - ✓ Supervised
 - Decision Trees, Neural Networks
 - Learning Bayesian Networks
 - ✓ Unsupervised
 - EM Algorithm
- ✓ Reinforcement Learning
 - ▶ Games and Adversarial Search
 - ▶ Minimax search and Alpha-beta pruning
 - ▶ Multiagent search
 - ▶ Knowledge representation and Reasoning
 - ▶ Propositional logic
 - ▶ First order logic
 - ▶ Inference
 - ▶ Planning

Outline

1. Recap
2. Speech Recognition
3. Machine Translation
 - Statistical MT
 - Rule-based MT

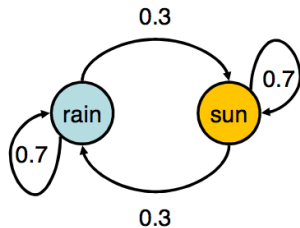
Recap: Sequential data

Markov models



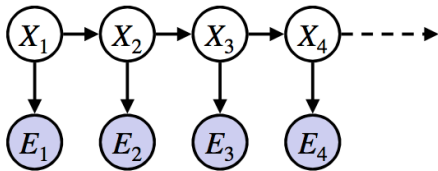
$$P(X_1)$$

$$P(X_i|X_{i-1})$$



$$P(E|X)$$

Hidden Markov models



X	E	P
rain	umbrella	0.9
rain	no umbrella	0.1
sun	umbrella	0.2
sun	no umbrella	0.8

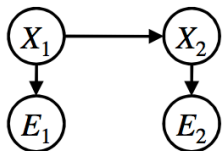
Recap: Filtering

Elapse time: compute $P(X_t | e_{1:t-1})$

$$P(x_t | e_{1:t-1}) = \sum_{x_{t-1}} P(x_{t-1} | e_{1:t-1}) \cdot P(x_t | x_{t-1})$$

Observe: compute $P(X_t | e_{1:t})$

$$P(x_t | e_{1:t}) \propto P(x_t | e_{1:t-1}) \cdot P(e_t | x_t)$$



Belief: $\langle P(\text{rain}), P(\text{sun}) \rangle$

$P(X_1)$ $\langle 0.5, 0.5 \rangle$ *Prior on X_1*

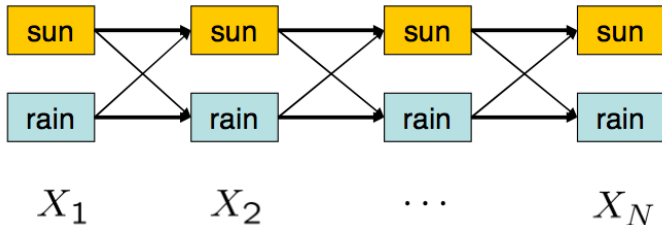
$P(X_1 | E_1 = \text{umbrella})$ $\langle 0.82, 0.18 \rangle$ *Observe*

$P(X_2 | E_1 = \text{umbrella})$ $\langle 0.63, 0.37 \rangle$ *Elapse time*

$P(X_2 | E_1 = \text{umb}, E_2 = \text{umb})$ $\langle 0.88, 0.12 \rangle$ *Observe*

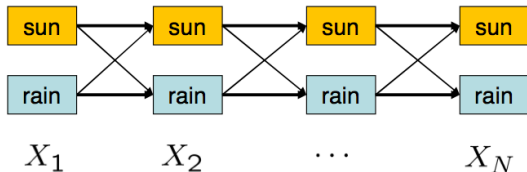
Recap: State Trellis

- ▶ State trellis: graph of states and transitions over time



- ▶ Each arc represents some transition $x_{t-1} \rightarrow x_t$
- ▶ Each arc has weight $\Pr(x_t | x_{t-1}) \Pr(e_t | x_t)$
- ▶ Each path is a sequence of states
- ▶ The product of weights on a path is the seq's probability
- ▶ Can think of the Forward (and now Viterbi) algorithms as computing sums of all paths (best paths) in this graph

Recap: Forward/Viterbi



$$f_t[x_t] = P(x_t, e_{1:t})$$

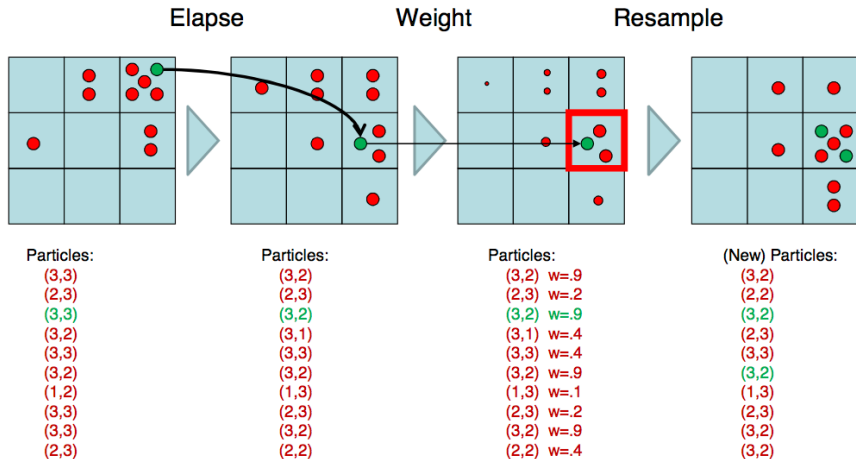
$$m_t[x_t] = \max_{x_{1:t-1}} P(x_{1:t-1}, x_t, e_{1:t})$$

$$= P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1}) f_{t-1}[x_{t-1}]$$

$$= P(e_t|x_t) \max_{x_{t-1}} P(x_t|x_{t-1}) m_{t-1}[x_{t-1}]$$

Recap: Particle Filtering

Particles: track samples of states rather than an explicit distribution



Natural Language

- ▶ 100.000 years ago humans started to speak
- ▶ 7.000 years ago humans started to write

Machines process natural language to:

- ▶ acquire information
- ▶ communicate with humans

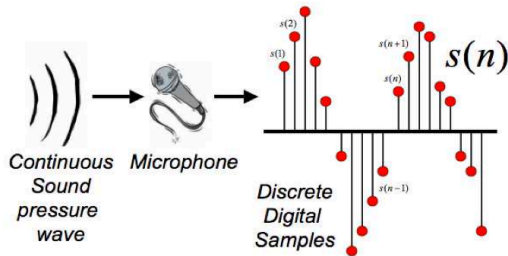
Natural Language Processing

- ▶ Speech technologies
 - ▶ Automatic speech recognition (ASR)
 - ▶ Text-to-speech synthesis (TTS)
 - ▶ Dialog systems
- ▶ Language processing technologies
 - ▶ Machine translation
 - ▶ Information extraction
 - ▶ Web search, question answering
 - ▶ Text classification, spam filtering, etc.

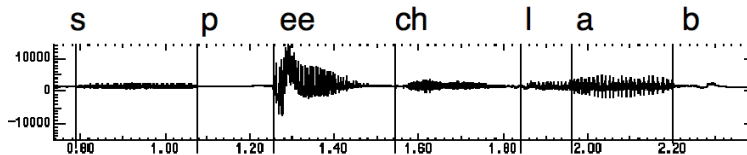
Outline

1. Recap
2. Speech Recognition
3. Machine Translation
 - Statistical MT
 - Rule-based MT

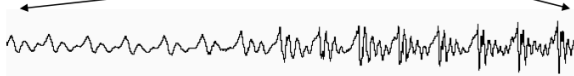
Digitalizing Speech



Speech input is an acoustic wave form

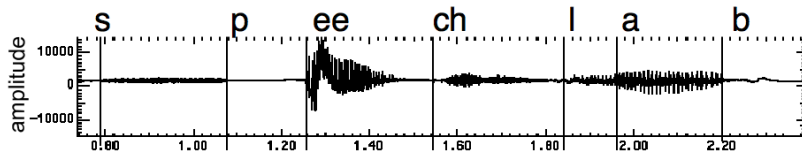


“l” to “a”
transition:

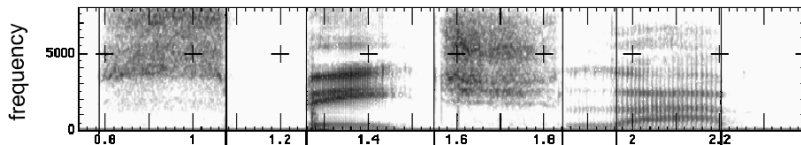


Spectral Analysis

- Frequency gives pitch; amplitude gives volume
 - sampling at ~8 kHz phone, ~16 kHz mic (kHz=1000 cycles/sec)

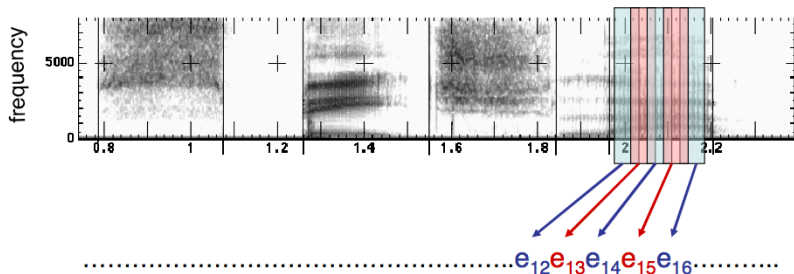


- Fourier transform of wave displayed as a spectrogram
 - darkness indicates energy at each frequency



Acoustic Feature Sequence

- Time slices are translated into acoustic feature vectors (~39 real numbers per slice)

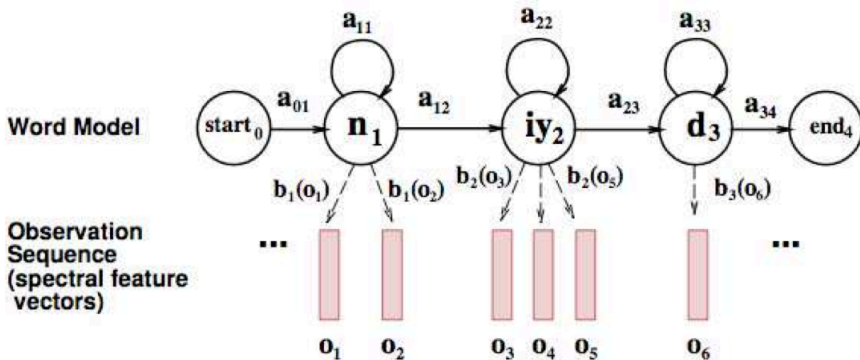


- These are the observations, now we need the hidden states X

State Space

- ▶ $\Pr(E|X)$ encodes which acoustic vectors are appropriate for each phoneme (each kind of sound)
- ▶ $\Pr(X|X')$ encodes how sounds can be strung together
- ▶ We will have one state for each sound in each word
- ▶ From some state x , can only:
 - ▶ Stay in the same state (e.g. speaking slowly)
 - ▶ Move to the next position in the word
 - ▶ At the end of the word, move to the start of the next word
- ▶ We build a little state graph for each word and chain them together to form our state space X

HMM for speech



Transition with Bigrams

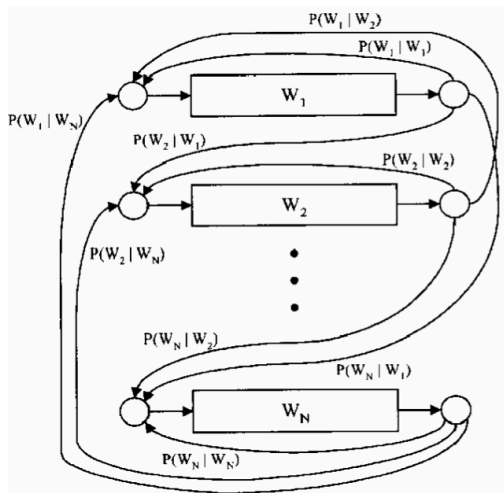


Figure from Huang et al page 618

Training Counts

198015222 the first
 194623024 the same
 168504105 the following
 158562063 the world
 ...
 14112454 the door

 23135851162 the *

$$\hat{P}(\text{door}|\text{the}) = \frac{14112454}{23135851162}$$

$$= 0.0006$$

Decoding

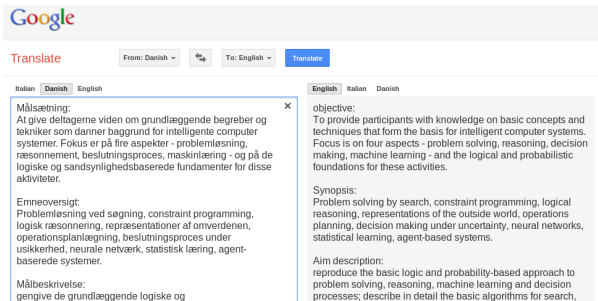
- ▶ While there are some practical issues, finding the words given the acoustics is an HMM inference problem
- ▶ We want to know which state sequence $x_{1:T}$ is most likely given the evidence $e_{1:T}$:

$$\begin{aligned}x_{1:T}^* &= \arg \max_{x_{1:T}} P(x_{1:T} | e_{1:T}) \\ &= \arg \max_{x_{1:T}} P(x_{1:T}, e_{1:T})\end{aligned}$$

- ▶ From the sequence x , we can simply read off the words

Outline

1. Recap
2. Speech Recognition
3. Machine Translation
 - Statistical MT
 - Rule-based MT



The screenshot shows the Google Translate interface. The source language is set to Danish and the target language is English. The input text is a Danish paragraph about a course objective. The output text is the English translation of that paragraph.

From: Danish **To:** English **Translate**

Italian Danish English

Målsætning:
At give deltagerne viden om grundlæggende begreber og tekniker som danner baggrund for intelligente computer systemer. Fokus er på fire aspekter - problemløsning, ræsonnement, beslutningsproces, maskinlæring - og på de logiske og sandsynlighedsbaserede fundamentet for disse aktiviteter.

Emneoversigt:
Problemløsning ved søgning, constraint programming, logisk ræsonnering, repræsentationer af omverdenen, operationsplanlægning, beslutningsproces under usikkerhed, neurale netværk, statistisk læring, agent-baserede systemer.

Målbekrivelse:
gengive de grundlæggende logiske og

English Italian Danish

objective:
To provide participants with knowledge on basic concepts and techniques that form the basis for intelligent computer systems. Focus is on four aspects - problem solving, reasoning, decision making, machine learning - and the logical and probabilistic foundations for these activities.

Synopsis:
Problem solving by search, constraint programming, logical reasoning, representations of the outside world, operations planning, decision making under uncertainty, neural networks, statistical learning, agent-based systems.

Aim description:
reproduce the basic logic and probability-based approach to problem solving, reasoning, machine learning and decision processes; describe in detail the basic algorithms for search,

- ▶ **Fundamental goal:** analyze and process human language, broadly, robustly, accurately...
- ▶ **End systems that we want to build:**
Ambitious: speech recognition, machine translation, information extraction, dialog interfaces, question answering...
Modest: spelling correction, text categorization, language recognition, genre classification.

Language Models

- ▶ Language defined by a sequence of strings and rules called **grammars**.
- ▶ Formal languages also need **semantics** that define meaning.
- ▶ Natural Languages:
 1. not definitive: is disagreement with grammar rules
“Not to be invited is sad”
“To be not invited is sad”
 2. ambiguous:
“Entire store 25% off”
“I will bring my bike tomorrow if it looks nice in the morning.”
 3. large and constantly changing

- ▶ n -gram sequence of n characters or sequence of n words, syllables
- ▶ n -gram models: define probability distributions for these sequences
- ▶ n -gram model is defined as a **Markov chain** of order $n - 1$.
For a trigram:

$$p(c_i | c_{1:i-1}) = p(c_i | c_{i-2:i-1})$$
$$p(c_{1:N}) = \prod_{i=1}^N \Pr(c_i | c_{1:i-1}) = \prod_{i=1}^N \Pr(c_i | c_{i-2:i-1})$$

- ▶ 100 chars \rightsquigarrow millions of entries
- ▶ with words even worse
- ▶ **Corpus** body of text

Language identification

Learned from corpus:

$$p(c_i \mid c_{i-2:i-1}, l)$$

Most probable language:

$$\begin{aligned} l^* &= \operatorname{argmax}_l p(l \mid c_{1:N}) \\ &= \operatorname{argmax}_l p(l) p(c_{1:N} \mid l) \end{aligned} \quad (\text{Bayes})$$

$$= \operatorname{argmax}_l p(l) \prod_{i=1}^N p(c_i \mid c_{i-2:i-1}, l) \quad (\text{Markov property})$$

Computers can reach 99% accuracy

Machine Translation

Rough translation: gives the main point but contains errors

Pre-edited translation: original text written in constrained language easier to translate automatically

Restricted-source translation: fully automatic but only on technical content as e.g. weather forecast

Machine Translation Systems

Very simplified there are three types of machine translation

Statistical machine translation (SMT) learn relational dependencies of features such as grams, lemmas, etc. • Requires large data sets
• Example: google translate • Relatively easy to implement

Rule-based machine translation (RBMT) use grammatical rules and language constructions to analyze syntax and semantics • Use moderate size data sets • Long development time and expertise

Hybrid machine translation either construct from RBMT and use SMT to post-process and optimize the result • Or use grammatical rules to derive further features to then be fed in the statistical learning machine • New direction of research.

Brief History



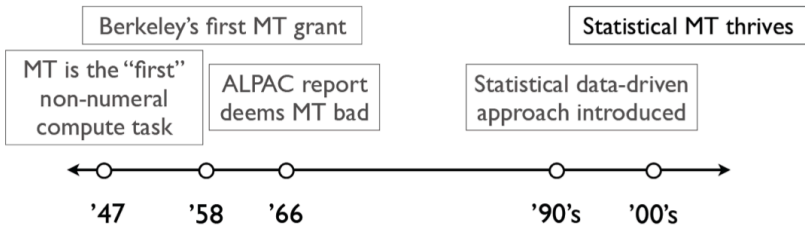
Warren Weaver

When I look at an article in Russian, I say: "This is really written in English, but it has been coded in some strange symbols. I will now proceed to decode."



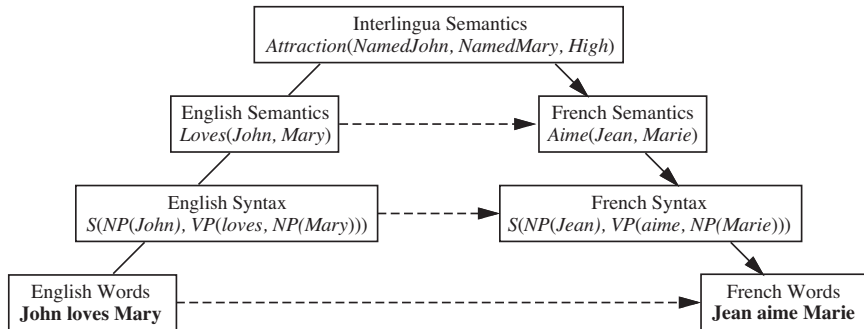
John Pierce

"Machine Translation" presumably means going by algorithm from machine-readable source text to useful target text... In this context, there has been no machine translation...



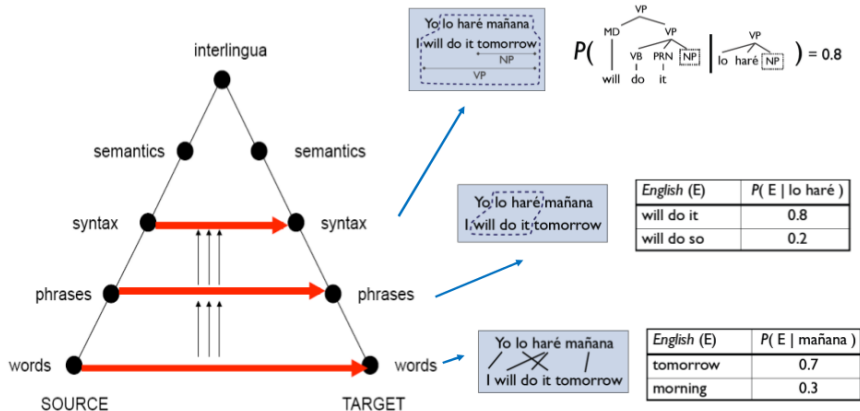
- ▶ Interlingual model: the source language, i.e. the text to be translated is transformed into an interlingua, i.e., an abstract language-independent representation. The target language is then generated from the interlingua.
- ▶ Transfer model: the source language is transformed into an abstract, less language-specific representation. Linguistic rules which are specific to the language pair then transform the source language representation into an abstract target language representation and from this the target sentence is generated.
- ▶ Direct model: words are translated directly without passing through an additional representation.

Levels of Transfer



Vauquois pyramid

Levels of Transfer



The problem with dictionary look ups

顶部	/ top /roof/
顶端	/summit/peak/ top /apex/
顶头	/coming directly towards one/ top /end/
盖	/lid/ top /cover/canopy/build/Gai/
盖帽	/surpass/ top /
极	/extremely/pole/utmost/ top /collect/receive/
尖峰	/peak/ top /
面	/fade/side/surface/aspect/ top /face/flour/
摘心	/ top /topping/

Statistical machine translation

Data driven MT

Recap
Speech Recognition
Machine Translation

Target language corpus:

I will get to it soon

See you later

He will do it

Sentence-aligned parallel corpus:

Yo lo haré mañana
I will do it tomorrow

Hasta pronto
See you soon

Hasta pronto
See you around

Machine translation system:

Yo lo haré pronto

NOVEL SENTENCE

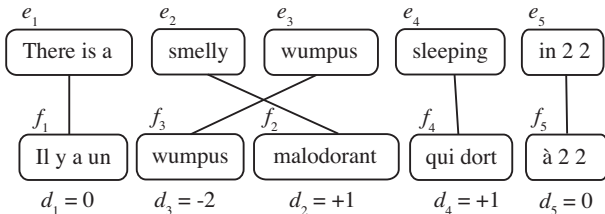
Model of
translation

I will do it soon

- ▶ e sequence of strings in English
- ▶ f sequence of strings in French

$$f^* = \operatorname{argmax}_f \Pr(f | e) = \operatorname{argmax}_f \Pr(e | f) \Pr(f)$$

- ▶ $\Pr(e | f)$ learned from bilingual (parallel) corpus made of phrases seen before



Given English sentence e find French sentence f^* :

1. break English e into phrases e_1, \dots, e_n
2. $\forall e_i$ choose the French f_i : $\Pr(f_i | e_i)$
3. choose a permutation of phrases f_1, \dots, f_n
 $\forall f_i$ choose distortion d_i : num. of words that phrase f_i has moved wrt f_{i-1}

$$\Pr(f, d | e) = \prod_{i=1}^n \Pr(f_i | e_i) \Pr(d_i)$$

with 100 French phrases for a 5-gram English there are 100^5 different 5-gram and $5!$ reorderings.

Learn probabilities

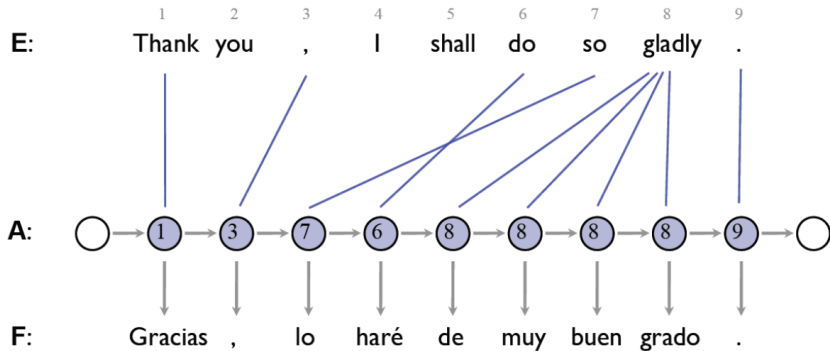
1. Parallel corpus: parliamentary debates, web pages
2. Segment into sentences. Periods are good indicators with some care.
3. Align sentences. length of sentences is an indicator, landmarks another
4. Align phrases within sentence: iterative process, aggregation of evidence, no other pair appear so frequently in the corpus. $\Pr(f_i | e_i)$
5. Extract distortions: count how often distortions appear in the corpus after phrase alignment (smoothing)
6. Improve estimates of $\Pr(f | e)$ and $\Pr(d)$ with EM.

Learning to translate

CLASSIC SOUPS

			Sm.	Lg.				
清	燉	雞	湯	57.	House Chicken Soup (Chicken, Celery, Potato, Onion, Carrot)	1.50	2.75	
雞	飯	湯	58.	Chicken Rice Soup	1.85	3.25		
雞	麵	湯	59.	Chicken Noodle Soup	1.85	3.25		
廣	東	雲	吞	60.	Cantonese Wonton Soup	1.50	2.75	
蕃	茄	蛋	湯	61.	Tomato Clear Egg Drop Soup	1.65	2.95	
雲	吞	湯	62.	Regular Wonton Soup	1.10	2.10		
酸	辣	湯	63.	Hot & Sour Soup	1.10	2.10		
蛋	花	湯	64.	Egg Drop Soup	1.10	2.10		
雲	蛋	湯	65.	Egg Drop Wonton Mix	1.10	2.10		
豆	腐	菜	湯	66.	Tofu Vegetable Soup	NA	3.50	
雞	玉	米	湯	67.	Chicken Corn Cream Soup	NA	3.50	
蟹	肉	玉	米	湯	68.	Crab Meat Corn Cream Soup	NA	3.50
海	鮮	湯	69.	Seafood Soup	NA	3.50		

An HMM model

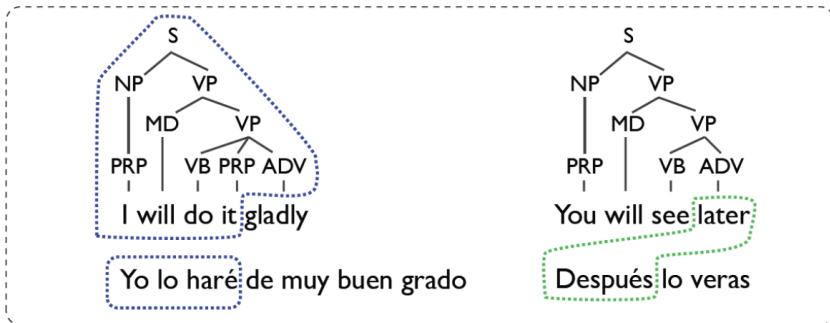


Model Parameters

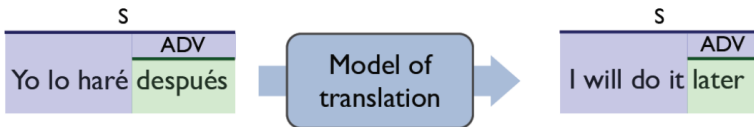
Emissions: $P(F_1 = \text{Gracias} \mid A_1 = \text{Thank})$

Transitions: $P(A_2 = 3 \mid A_1 = 1)$

Machine translation systems



Machine translation system:



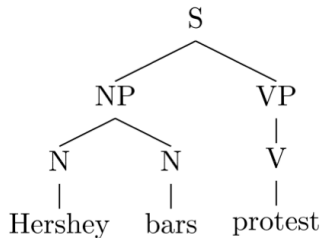
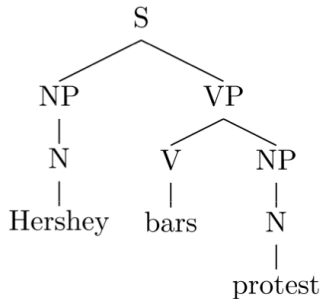
Grammars

Grammars: set of rules (from left to right) that describe how to form strings from the language's alphabet that are valid according to the language's syntax ([Language generator](#)).

Parsing is the process of [recognizing](#) a string in natural languages by breaking it down to a set of symbols and analyzing each one against the grammar of the language, ie, determining whether the string belongs to the language or is grammatically incorrect. The result is a parse tree.

- ▶ context free grammars (see http://en.wikipedia.org/wiki/Chomsky_hierarchy)
- ▶ probabilistic context free grammars
- ▶ lexicalized probabilistic context free grammars

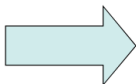
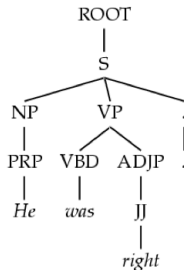
Parsing as search



Hershey bars protest

Probabilistic Context Free Grammars

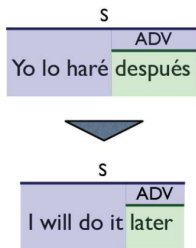
- Natural language grammars are very ambiguous!
- PCFGs are a formal probabilistic model of trees
 - Each “rule” has a conditional probability (like an HMM)
 - Tree’s probability is the product of all rules used
- Parsing: Given a sentence, find the best tree – search!



ROOT → S	375/420
S → NP VP .	320/392
NP → PRP	127/539
VP → VBD ADJP	32/401
.....	

Hybrid Systems

Synchronous Derivation



Synchronous Grammar Rules

$S \rightarrow \langle \text{Yo lo haré ADV} ; \text{I will do it ADV} \rangle$

$ADV \rightarrow \langle \text{después} ; \text{later} \rangle$

The translated sentence can be checked against a monolingual corpus.

A Statistical Model

*Translation model components
factor over applied rules*

How well are these rules supported by the data?

Language model factors over n-grams

How well is this output sentence supported by the data?

Machine Translation

- ▶ Translate text from one language to another
- ▶ Recombines fragments of example translations
- ▶ Challenges:
 - ▶ What fragments? [learning to translate]
 - ▶ How to make efficient? [fast translation search]

Machine Translation

- ▶ After a first bubble now full speed in the sector
- ▶ In spite of the economical crisis 7% growth on world basis
- ▶ Commercial and technological focus
- ▶ Danish is a marginal language and existing systems cannot be applied reliably
- ▶ www.eicom.dk and www.oversaetterhuset.dk search development in collaboration with research institutions (SDU, CBS, ASB)

Need for human resources, possibilities for thesis and individual study activities together with:

- ▶ Visual Interactive Syntax Learning project at the Institute for Language and Communication of SDU

http://beta.visl.sdu.dk/constraint_grammar.html

- ▶ Eckhard Bick project leader

http://en.wikipedia.org/wiki/Eckhard_Bick

If interested contact me.