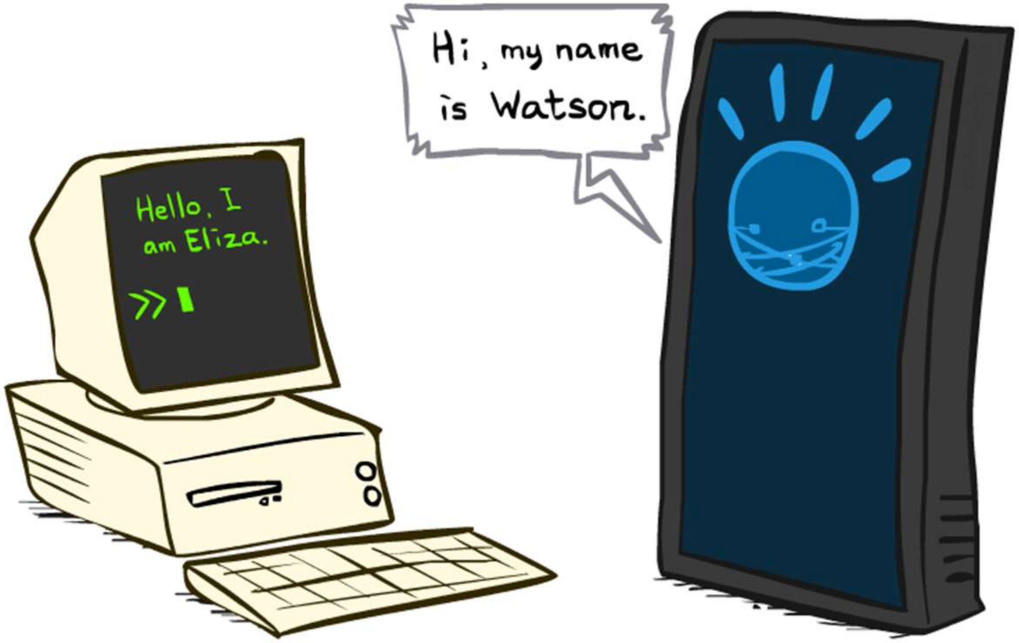
CSE 473: Artificial Intelligence

Advanced Applic's: Natural Language Processing

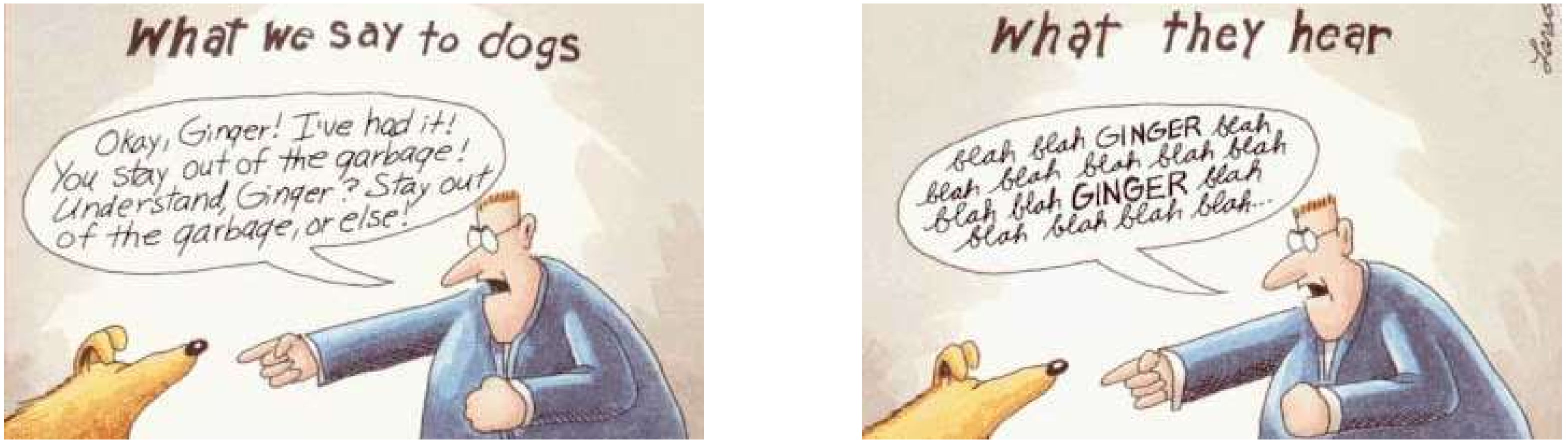


Steve Tanimoto --- University of Washington

[Some of these slides were created by Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley.

All CS188 materials are available at http://ai.berkeley.edu.]

What is NLP?


What we say to dogs
Okay, Ginger! I've had it! You stay out of the garbage! Understand, Ginger? Stay out of the garbage, or else!
What they hear
blah blah GINGER blah blah blah blah blah blah blah blah GINGER blah blah blah blah blah blah blah...

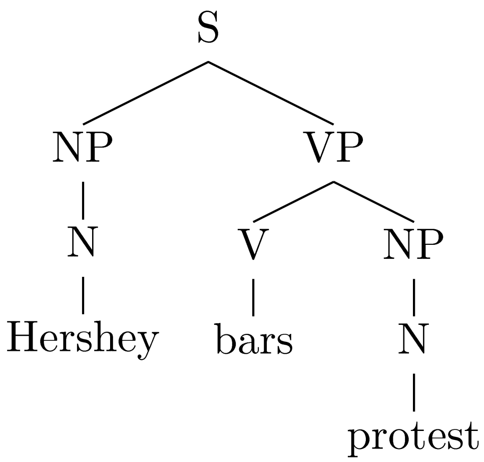
- Fundamental goal: analyze and process human language, broadly, robustly, accurately…
- End systems that we want to build:
  - Ambitious: speech recognition, machine translation, information extraction, dialog interfaces, question answering…
  - Modest: spelling correction, text categorization…

Problem: Ambiguities

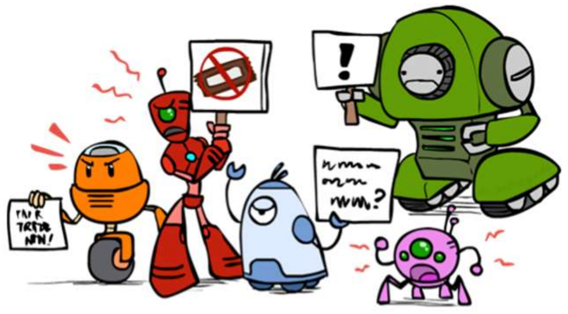
- Headlines:
  - Enraged Cow Injures Farmer With Ax
  - Hospitals Are Sued by 7 Foot Doctors
  - Ban on Nude Dancing on Governor’s Desk
  - Iraqi Head Seeks Arms
  - Local HS Dropouts Cut in Half
  - Juvenile Court to Try Shooting Defendant
  - Stolen Painting Found by Tree
  - Kids Make Nutritious Snacks

- Why are these funny?

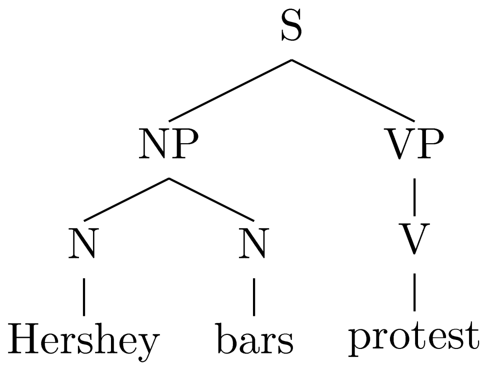
Parsing as Search



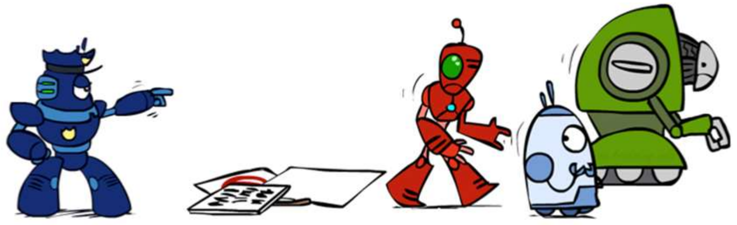
S
NP VP
N V NP
Hershey bars N
protest



Hershey bars protest

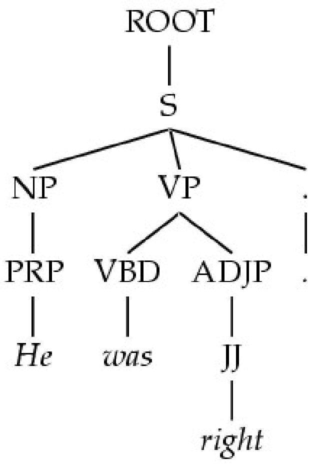


S
NP VP
N N V
Hershey bars protest



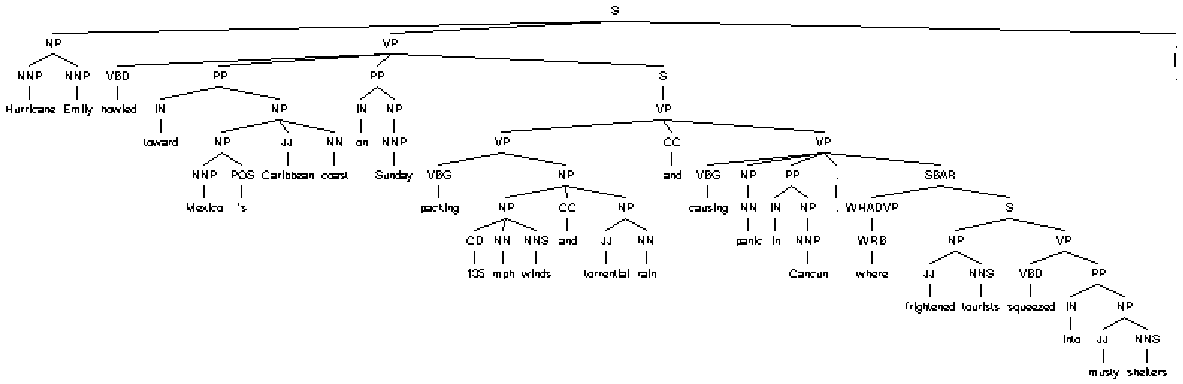
Grammar: PCFGs

- 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
…..	

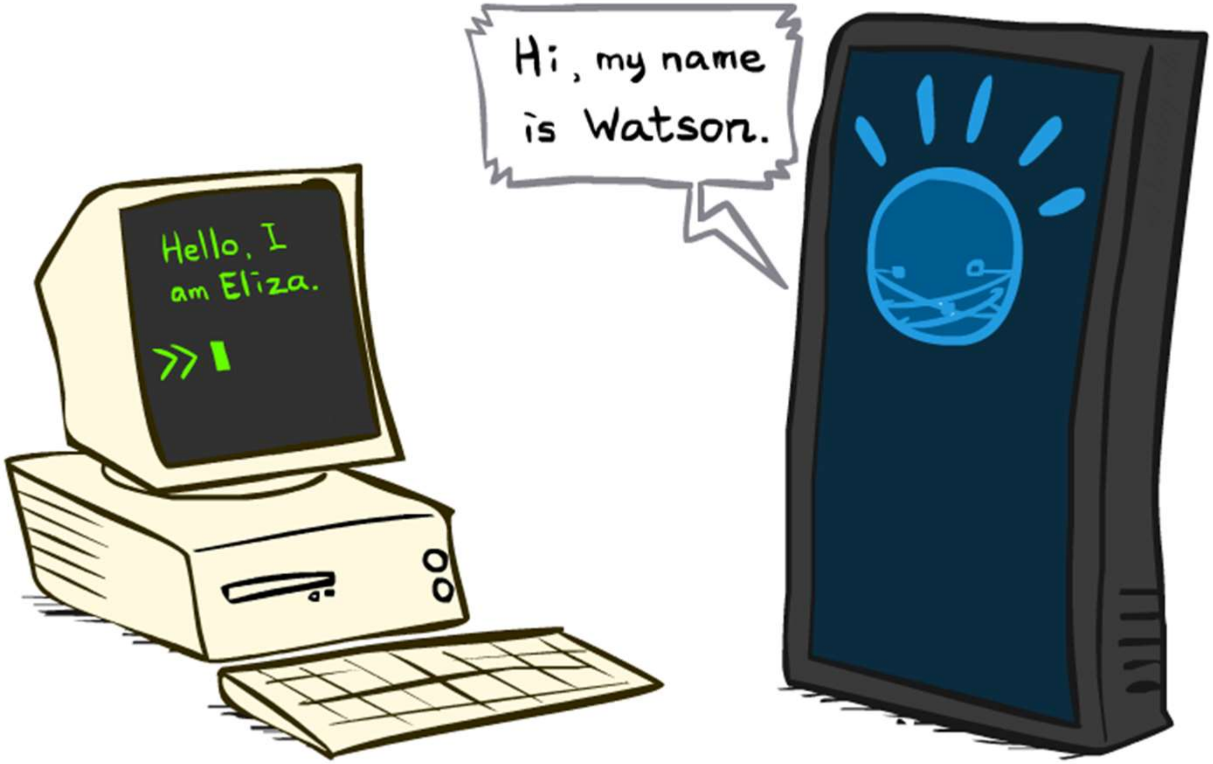
Syntactic Analysis



Hurricane Emily howled toward Mexico 's Caribbean coast on Sunday packing 135 mph winds and torrential rain and causing panic in Cancun, where frightened tourists squeezed into musty shelters.

[Demo: Berkeley NLP Group Parser http://tomato.banatao.berkeley.edu:8080/parser/parser.html]

Dialog Systems

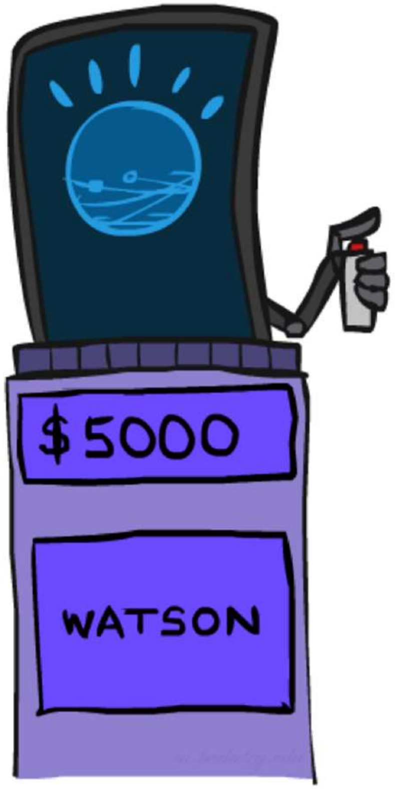


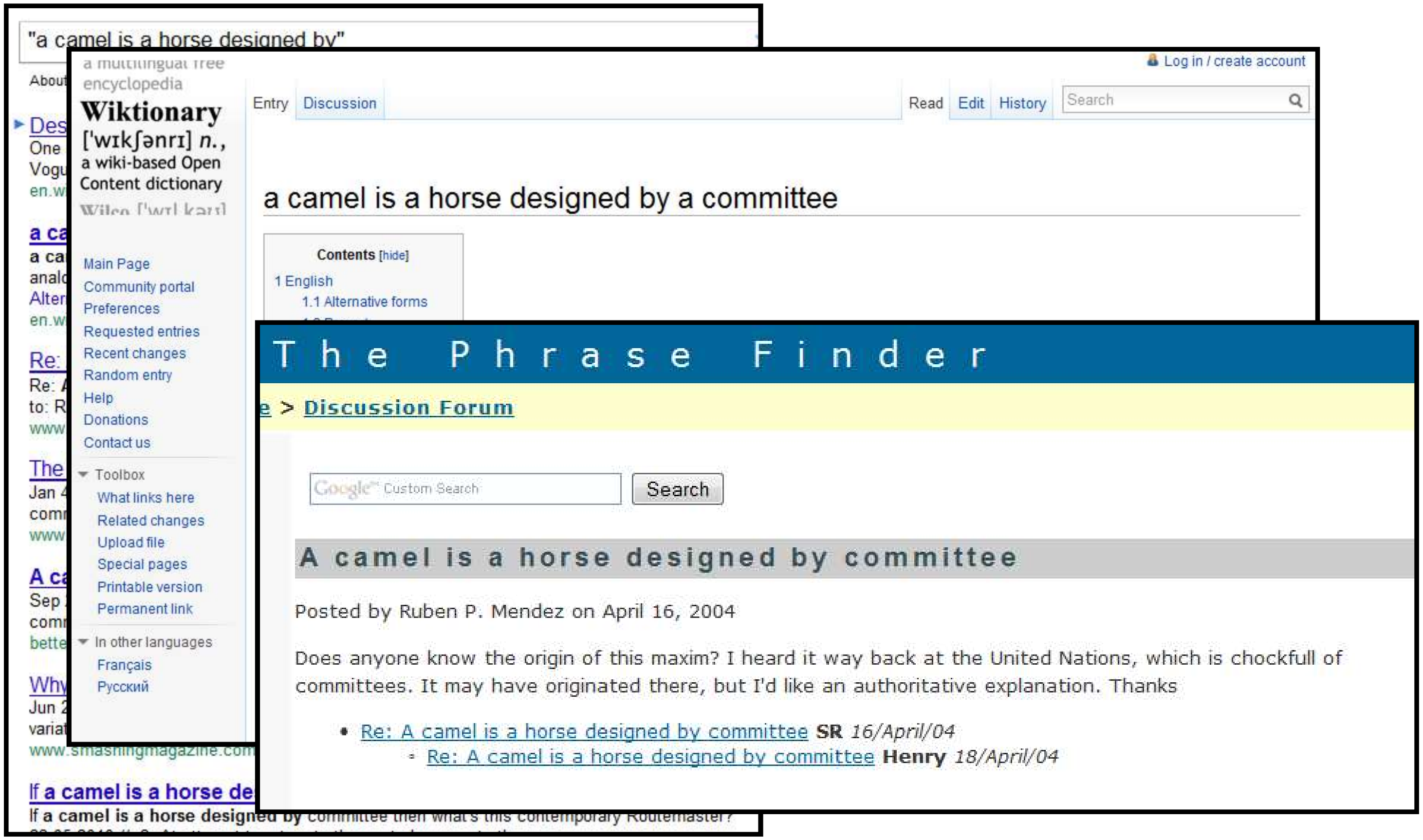
ELIZA

- A “psychotherapist” agent (Weizenbaum, ~1964)
- Led to a long line of chatterbots
- How does it work:
  - Trivial NLP: string match and substitution
  - Trivial knowledge: tiny script / response database
  - Example: matching “I remember __” results in “Do you often think of __”?
- Can fool some people some of the time?

[Demo: http://nlp-addiction.com/eliza]

Watson





a camel is a horse designed by a committee

If a camel is a horse de

If a camel is a horse designed by committee then what's this contemporary Routemaster?

www.smashingmagazine.com

A camel is a horse designed by committee

Posted by Ruben P. Mendez on April 16, 2004

Does anyone know the origin of this maxim? I heard it way back at the United Nations, which is chockfull of committees. It may have originated there, but I'd like an authoritative explanation. Thanks

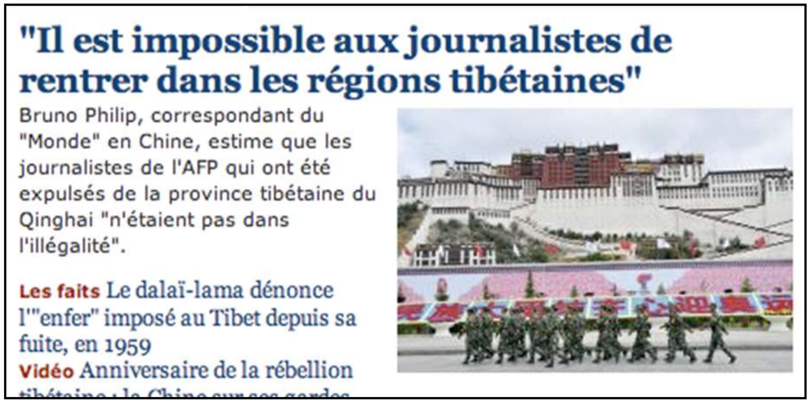
- Re: A camel is a horse designed by committee **SR** *16/April/04*
  - Re: A camel is a horse designed by committee **Henry** *18/April/04*

What’s in Watson?

- A question-answering system (IBM, 2011)
- Designed for the game of Jeopardy
- How does it work:
  - Sophisticated NLP: deep analysis of questions, noisy matching of questions to potential answers
  - Lots of data: onboard storage contains a huge collection of documents (e.g. Wikipedia, etc.), exploits redundancy
  - Lots of computation: 90+ servers
- Can beat all of the people all of the time?

Machine Translation

Machine Translation

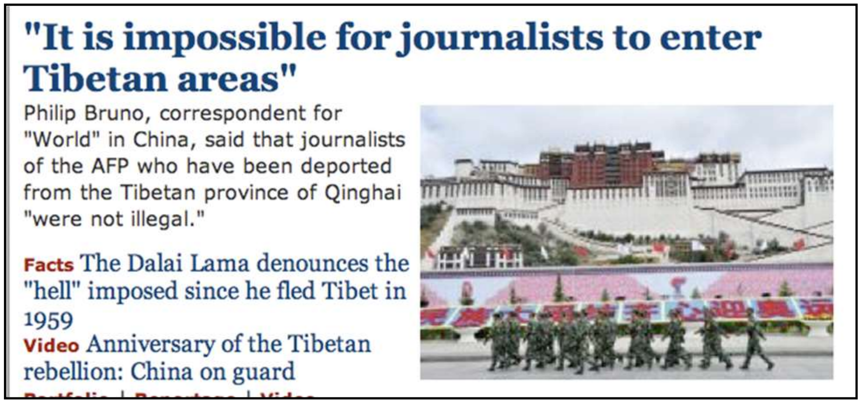


"Il est impossible aux journalistes de rentrer dans les régions tibétaines"

Bruno Philip, correspondant du "Monde" en Chine, estime que les journalistes de l'AFP qui ont été expulsés de la province tibétaine du Qinghai "n'étaient pas dans l'illégalité".

Les faits Le dalaï-lama dénonce l'"enfer" imposé au Tibet depuis sa fuite, en 1959

Vidéo Anniversaire de la rébellion tibétaine : la Chine sur ses gardes



"It is impossible for journalists to enter Tibetan areas"

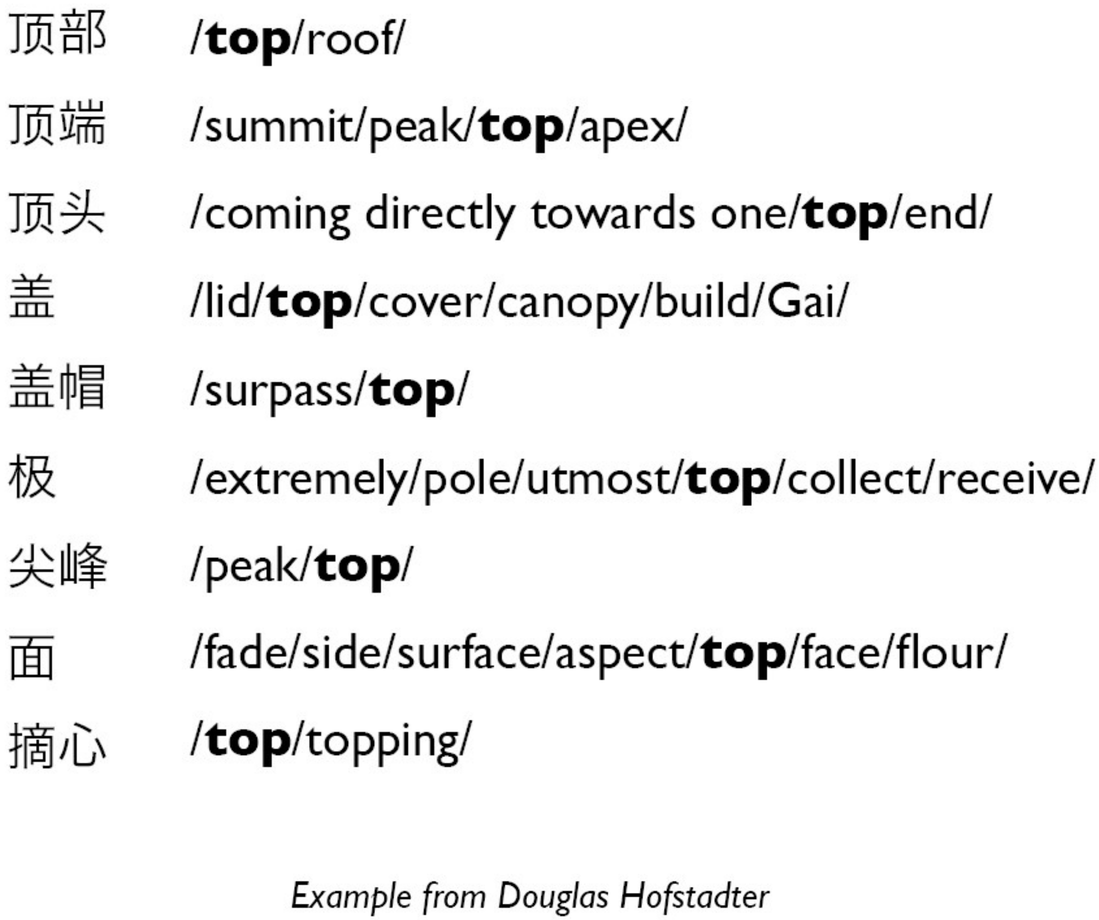
Philip Bruno, correspondent for "World" in China, said that journalists of the AFP who have been deported from the Tibetan province of Qinghai "were not illegal."

Facts The Dalai Lama denounces the "hell" imposed since he fled Tibet in 1959

Video Anniversary of the Tibetan rebellion: China on guard

- 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]

The Problem with Dictionary Lookups



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/

Example from Douglas Hofstadter

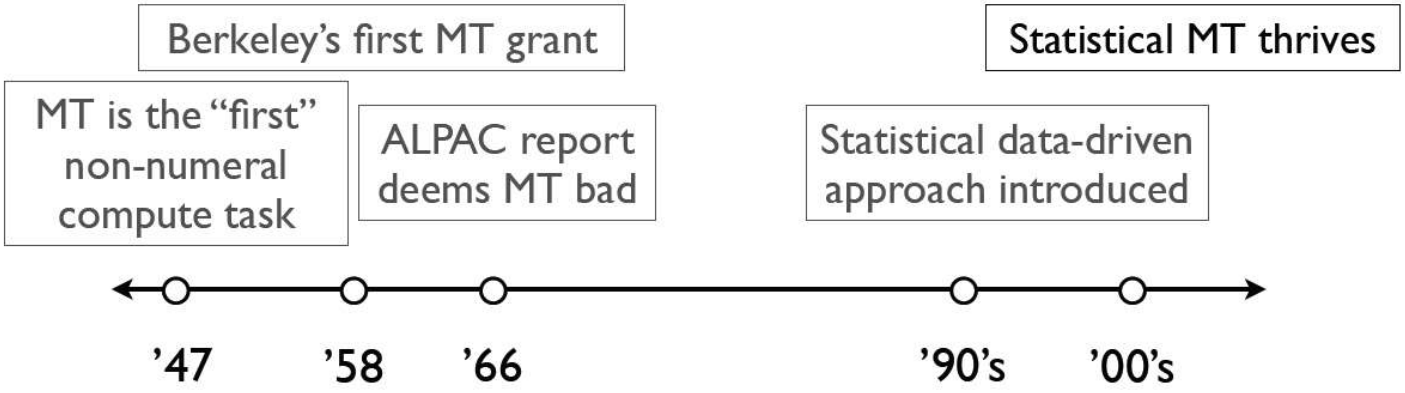
MT: 60 Years in 60 Seconds

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."

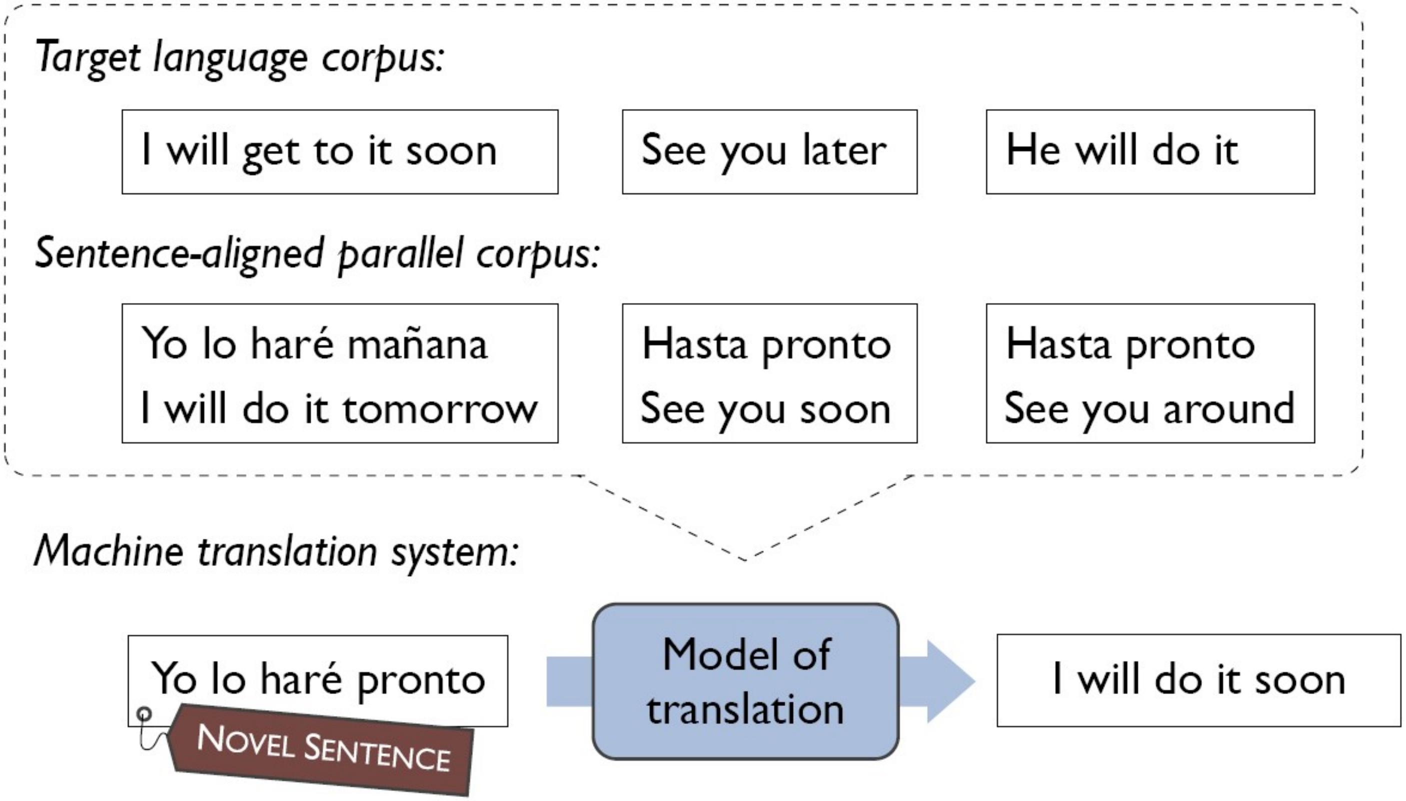
John Pierce

"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...



MT is the "first" non-numeral compute task
Berkeley's first MT grant
ALPAC report deems MT bad
Statistical data-driven approach introduced
Statistical MT thrives
'47
'58
'66
'90's
'00's

Data-Driven 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:

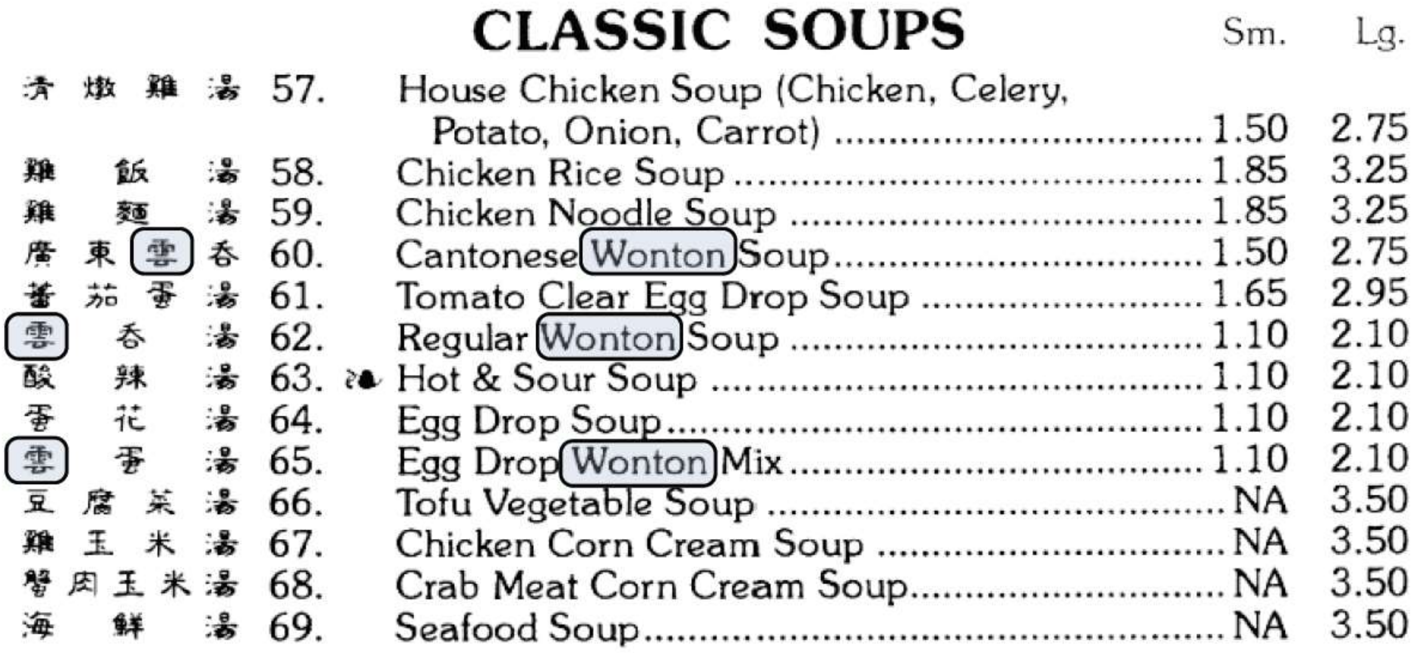
Yo lo haré pronto

NOVEL SENTENCE

Model of translation

I will do it soon

Learning to Translate

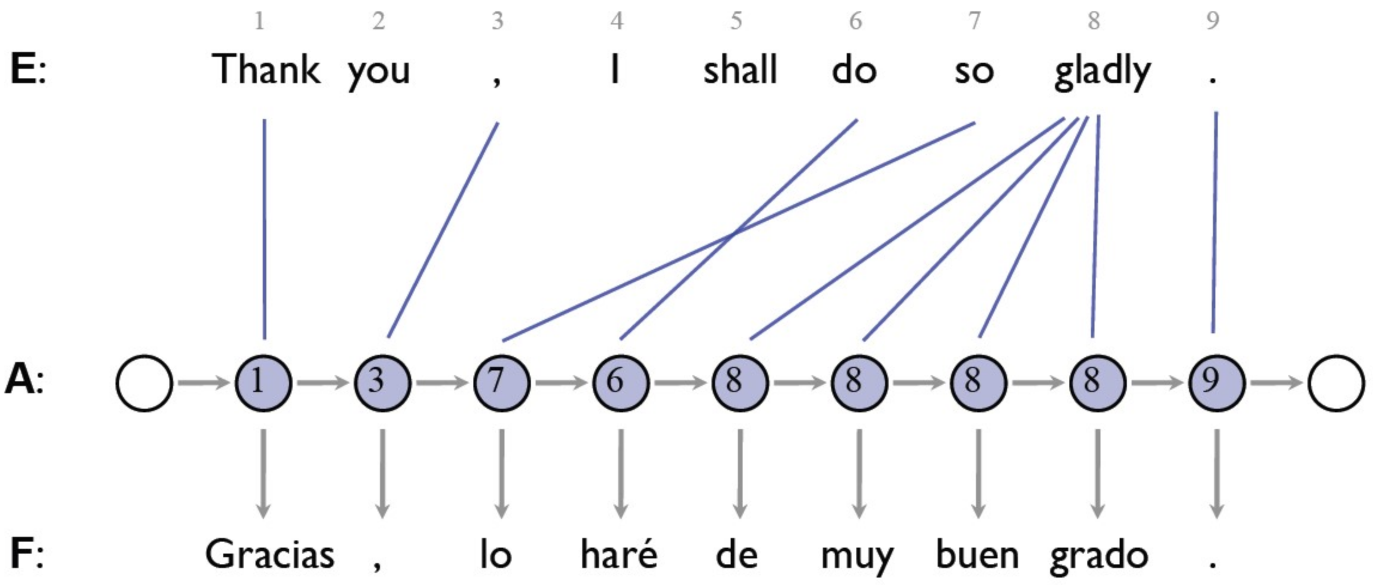


CLASSIC SOUPS

			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

Example from Adam Lopez

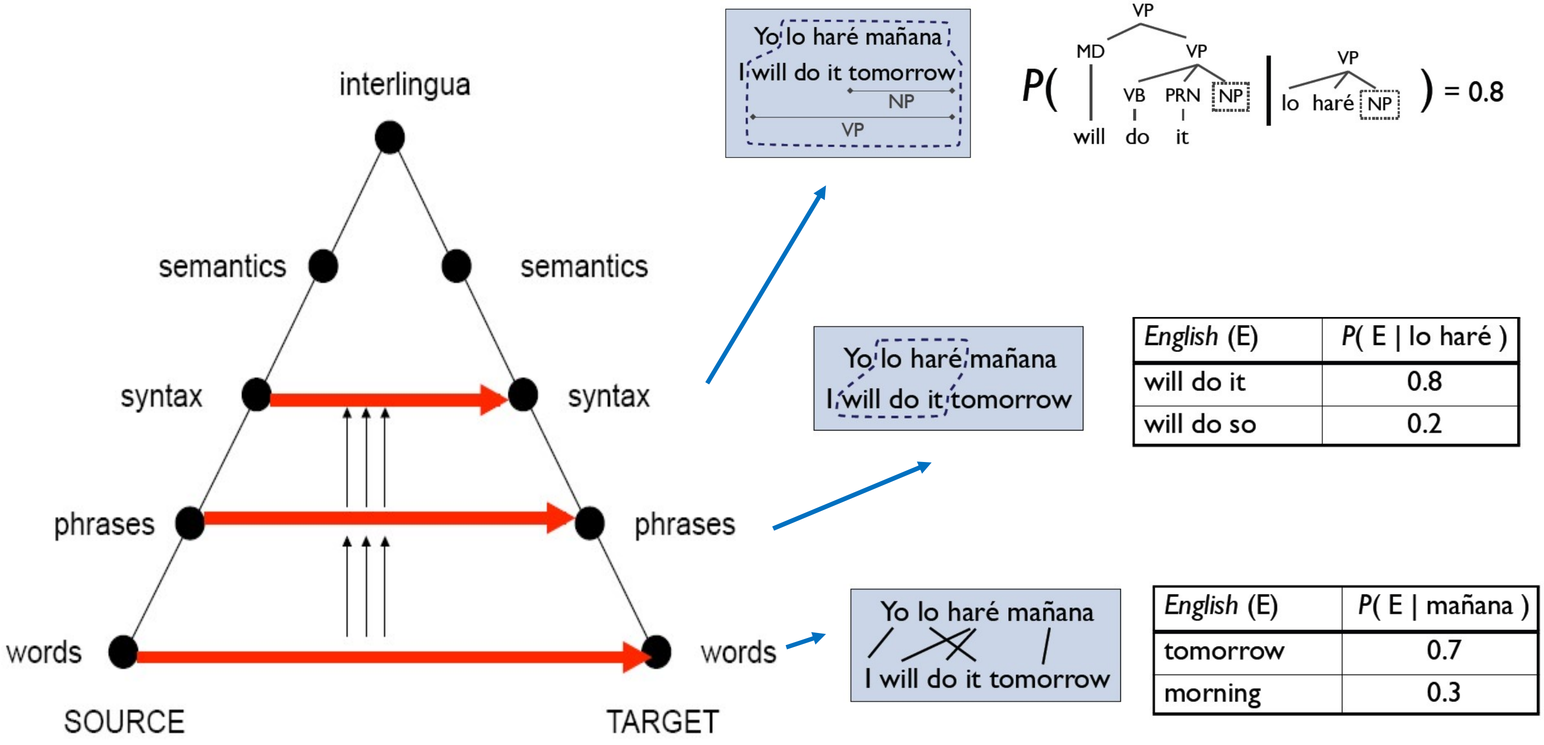
An HMM Translation Model


E:
1 2 3 4 5 6 7 8 9
Thank you , I shall do so gladly .
A:
1 3 7 6 8 8 8 8 9
F:
Gracias , lo haré de muy buen grado .

Model Parameters

Emissions: $P(F_1 = \text{Gracias} \mid E_{A_1} = \text{Thank})$ *Transitions*: $P(A_2 = 3 \mid A_1 = 1)$

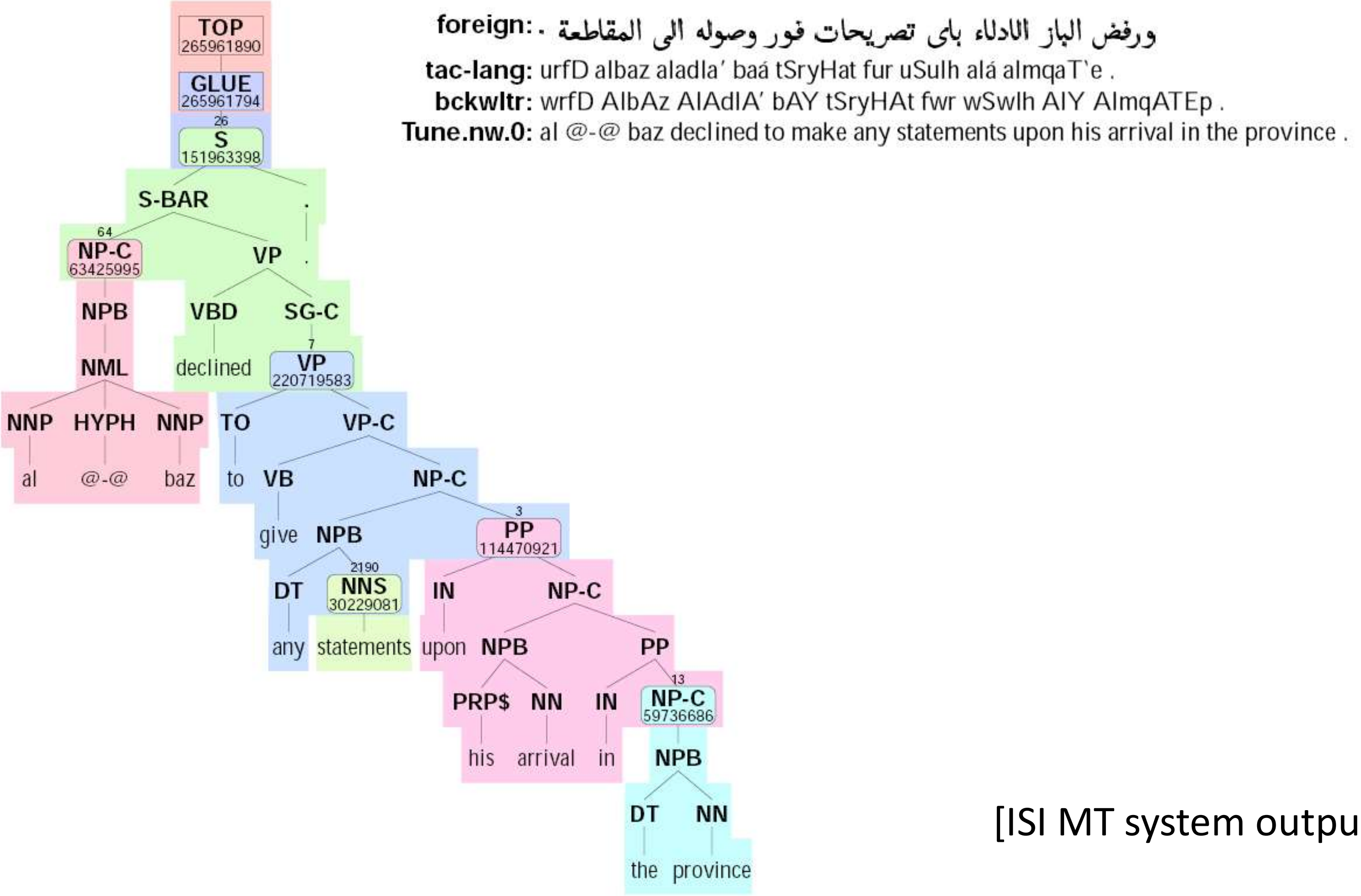
Levels of Transfer



English (E)	*P*(E | lo haré)
will do it	0.8
will do so	0.2

English (E)	*P*(E | mañana)
tomorrow	0.7
morning	0.3

Example: Syntactic MT Output



foreign: ورفض الباز الادلاء باى تصريحات فور وصوله الى المقاطعة .

tac-lang: urfD albaz aladla’ baá tSryHat fur uSulh alá almqaT`e .

bckwltr: wrfD AlbAz AlAdlA’ bAY tSryHAt fwr wSwlh AlY AlmqATEp .

Tune.nw.0: al @-@ baz declined to make any statements upon his arrival in the province .

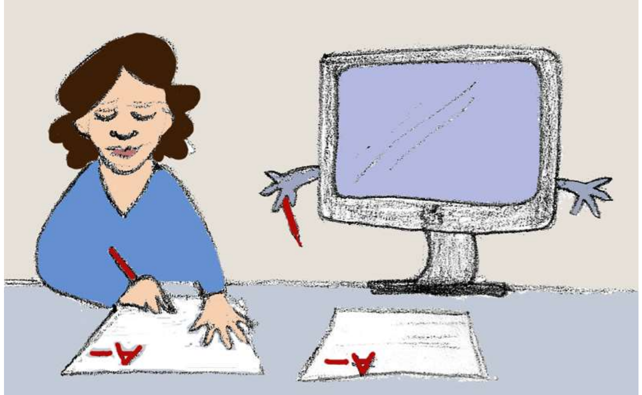
[ISI MT system output]

Document Analysis with LSA: Outline

- Motivation
- Bag-of-words representation
- Stopword elimination, stemming, reference vocabulary
- Vector-space representation
- Document comparison with the cosine similarity measure
- Latent Semantic Analysis

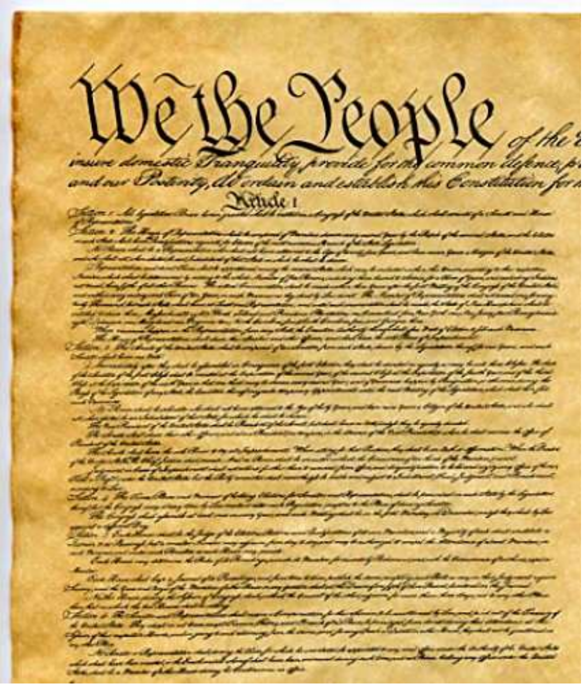
Motivation

- Document analysis is a highly active area, very relevant to information science, the World Wide Web, and search engines.
- Algorithms for document analysis span a wide range of techniques, from string processing to large matrix computations.
- One application: automatic essay grading.



Representations for Documents

- Text string
- Image (I.e., .jpg, .gif, and .png files)
- linguistically structured files: PostScript, Portable Doc. Format (PDF), XML.
- Vector: e.g., bag-of-words
- Hypertext, hypermedia



We the People of the

insure domestic Tranquility, provide for the common defence, p

and our Posterity, do ordain and establish this Constitution for

Article 1

[illegible]

Fundamental Problems

- Representation*
- Lexical Analysis (tokenizing)*
- Information Extraction*
- Comparison (similarity, distance)*
- Classification (e.g., for net-nanny service)*
- Indexing (to permit fast retrieval)
- Retrieval (querying and query processing)

*important for AI

Bag-of-Words Representation

A *multiset* is a collection like a set, but which allows duplicates (any number of copies) of elements.

{ a, b, c} is a set. (It is also a multiset.)

{ a, a, b, c, c, c } is not a set, but it is a multiset.

{ c, a, b, a, c, c } is the same multiset. (Order doesn’t matter).

A multiset is also called a *bag*.

Bag-of-Words (continued)

Let document D =

“The big fox jumped over the big fence.”

The bag representation is:

{ big, big, fence, fox, jumped, over, the, the }

For notational consistency, we use alphabetical order. Also, we omit punctuation and normalize the case.

The ordering information in the document is lost. But this is OK for some applications.

Eliminating Stopwords

In information retrieval and some other types of document analysis, we often begin by deleting words that don’t carry much meaning or that are so common that they do little to distinguish one document from another. Such words are called *stopwords*.

Examples: (articles) a, an, the; (quantifiers) any, some, only, many, all, no; (pronouns) I, you, it, he, she, they, me, him, her, them, his, hers, their, theirs, my, mine, your, our, yours, ours, this, that, these, those, who, whom, which; (prepositions) above, at, behind, below, beside, for, in, into, of, on, onto, over, under; (verbs) am, are, be, been, is, were, go, gone, went, had, have, do, did, can, could, will, would, might, may, must; (conjunctions) and, but, if, then, not, neither, nor, either, or; (other) yes, perhaps, first, last, there, where, when.

Stemming

In order to detect similarities among words, it often helps to perform stemming. We typically stem a word by removing its suffixes, leaving the basic word, or "uninflecting" the word

- apples → apple
- cacti → cactus
- swimming → swim
- swam → swim

Reference Vocabulary

A counterpart to stopwords is the *reference vocabulary*.

These are the words that ARE allowed in document representations.

These are all stemmed, and are not stopwords.

There might be several hundred or even thousands of terms in a reference vocabulary for real document processing.

Vector representation

Assume we have a reference vocabulary of words that might appear in our documents.

{apple, big, cat, dog, fence, fox, jumped, over, the, zoo}

We represent our bag

{ big, big, fence, fox, jumped, over, the, the }

by giving a vector (list) of occurrence counts of each reference term in the document:

[0, 2, 0, 0, 1, 1, 1, 1, 2, 0]

If there are n terms in the reference vocabulary, then each document is represented by a point in an n-dimensional space.

Indexing

Create links from terms to documents or document parts

(a) concordance

(b) table of contents

(c) book index

(d) index for a search engine

(e) database index for a relation (table)

Concordance

A *concordance* for a document is a sort of dictionary that lists, for each word that occurs in the document the sentences or lines in which it occurs.

“document”:
A concordance for a *document* is a sort of dictionary that lists, for each word that occurs in the *document* the

“occurs”:
that lists, for each word that *occurs* in the document the sentences or lines in which it *occurs*.

Search Engine Index

Query terms are organized into a large table or tree that can be quickly searched.

(e.g., large hash-table in memory, or a B-Tree with its top levels in memory).

Associated with each term is a list of occurrences, typically consisting of Document IDs or URLs.

Document Comparison

Typical problems:

•Determine whether two documents are slightly different versions of the same document. (applications: search engine hit filtering, plagiarism detection).

•Find the longest common subsequence for a pair of documents. (can be useful in genetic sequencing).

•Determine whether a new document should be placed into the same category as a model document. (essay grading, automatic response generation, etc.)

Cosine Similarity Function

Document 1:
"All Blues. First the key to last night's notes."

Document 2:
"How to get your message across. Restate your key points first and last. "

Reference vocabulary:
{ across, blue, first, key, last, message, night, note, point, restate, zebra }

Cosine Similarity (cont)

Document 1 reduced:
 blue first key last night note

Document 2 reduced:
 message across restate key point first last

Document 1 vector representation:
[0, 1, 1, 1, 1, 0, 1, 1, 0, 0, 0]

Document 2 vector representation:
[1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 0]

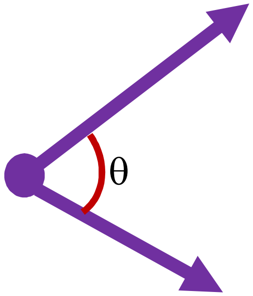
Cosine Similarity (cont)

Dot product (same as "inner product")
$[0, 1, 1, 1, 1, 0, 1, 1, 0, 0, 0] \cdot [1, 0, 1, 1, 1, 1, 0, 0, 1, 1, 0]$

$= 0 \cdot 1 + 1 \cdot 0 + 1 \cdot 1 + 1 \cdot 1 + 1 \cdot 1 + 0 \cdot 1 + 1 \cdot 0 + 1 \cdot 0 + 0 \cdot 1 + 0 \cdot 1 + 0 \cdot 0 = 3$

Normalized:
$\cos \theta = (v_1 \cdot v_2) / (|| v_1 || \, || v_2 ||)$

$|| v || = \sqrt{v \cdot v}$ $\cos \theta = 3 / (\sqrt{6} \sqrt{7}) \approx 0.4629.$



Properties of the Cosine Similarity

$\cos \theta = 0$ means that the document vectors are orthogonal and the documents have no reference vocabulary occurrences in common.

$\cos \theta = 1$ means that the documents are either identical or the vectors point in the same direction in the n-dim space. That is, the documents share the same distribution of occurrences of the reference terms.

Latent Semantic Analysis

A problem with the cosine similarity function:
Unless both documents use the same term for something, the similarity is not recognized.

“Computer learning environments have a great future.”

“Educational technology offers wonderful potential.”

cosine similarity is 0.

LSA (continued)

With Latent Semantic Analysis, the vector for each document is first transformed into a vector in another space -- a “semantic space” in which related terms get mapped to the same element or set of elements.

After that, the cosine similarity between the new vectors will be greater, if the documents share RELATED terms.

LSA (continued)

The semantic space for LSA is obtained from a set of documents given in advance.

The space is created using matrix factorization via the Singular Value Decomposition (SVD) method.

This is computationally costly, but modern computers are powerful enough to do it.

For more details, see Chapter 16 of *Introduction to Python for Artificial Intelligence.*

Singular Value Decomposition

Given term-document matrix A, having t rows and d columns, find TSD such that:

A = TSD
T is a t by t orthonormal matrix
D is a d by d orthonormal matrix
S is an m by m diagonal matrix, where m is the rank of A.

```
import LinearAlgebra as LA
(TSD) = LA.singular_value_decomposition(A)
```

Latent Semantic Model

Given TSD, form a reduced (and generalized) product $T_r\ S_r\,D_r$ by deleting the rows and columns of S that contain the n-k smallest diagonal values. Then eliminate the last n-k columns of T to get T_r and eliminate the last n-k rows of D to get D_r.

$A_r = T_r\ S_r\,D_r$

To compare two documents in the latent semantic space, first map the documents into the space and then compute their cosine similarity.

$doc_1' = D_r\,doc_1$; $doc_1' = D_r\,doc_1$; $\text{cossim}\,(doc_1' , doc_2')$

Example

d_1 = "the brown weasel followed the fox and stole the eggs"
d_2 = "behind the fence the thief fled with half a dozen“
d_3 = "artificial limbs can offer full mobility"

Documents used to create a semantic space:
"the lazy brown fox jumped over the fence"
"the thief jumped the lazy fence and fled"
"artificial intelligence is full of surprises"

cossim(d_1, d_2) = 0 Without LSA, d_1 and d_2 seem dissimilar.
cossim(d_1', d_2') = 1 With LSA, they are completely similar.
cossim(d_1, d_3) = cossim(d_1', d_3') = 0 But LSA does not make d_3 any more similar to the others.