Foundations of Natural Language Processing
Lecture 2
Introduction

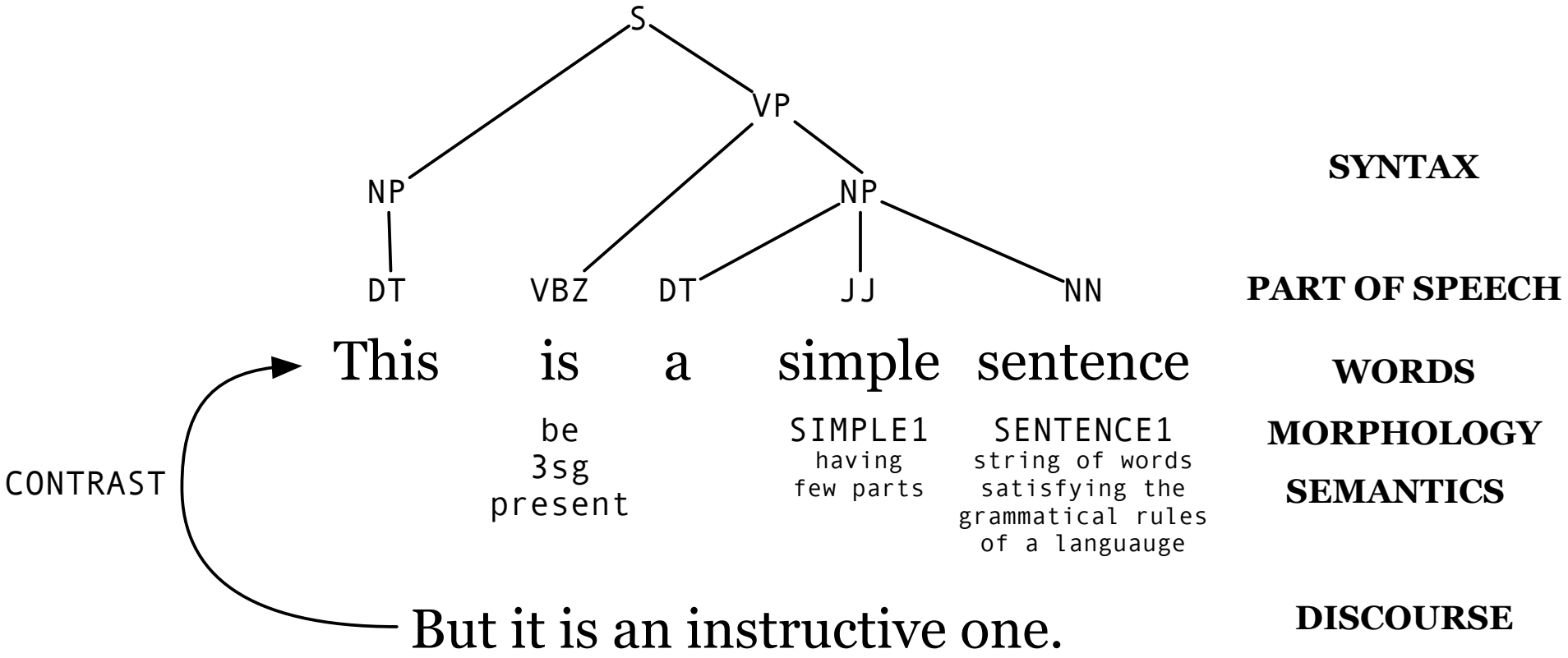
Ivan Titov
(Slides based on those of Philipp Koehn, Alex Lascarides, Sharon Goldwater, Shay Cohen, Khalil Sima'an)

18 January 2024

Plan for today

- Recap
- Continues discussing challenges making NLP hard (sparsity, diversity, ...)
- Why use *probabilistic* models (and machine learning) for NLP?

Levels of Structure



Levels of structure / processing in an application

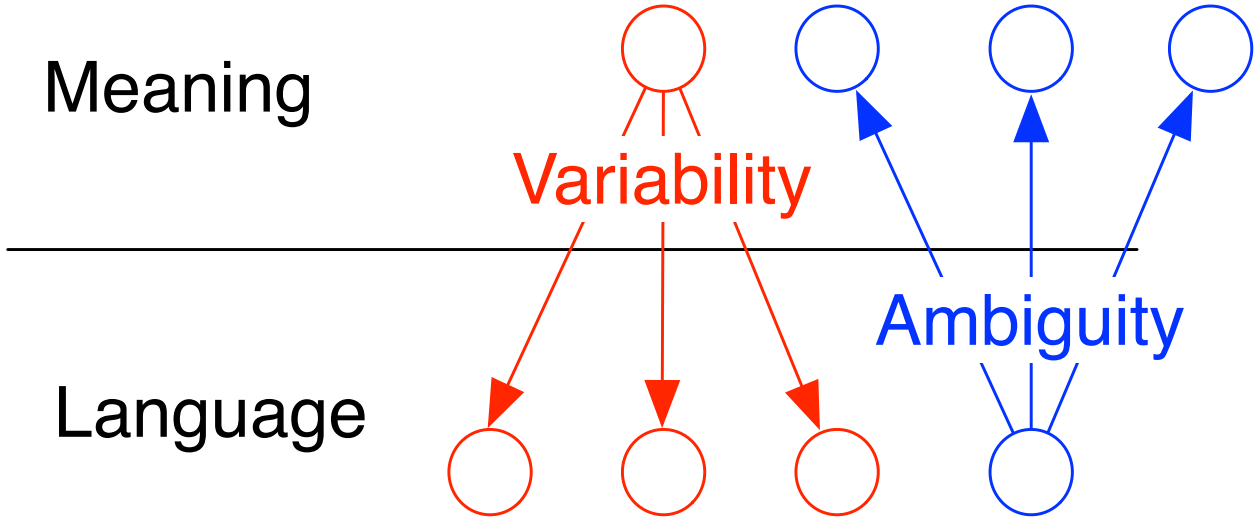
London Bridge really is falling down. The bridge is being taken apart and moved. Its new home will be a small town in Arizona. This bridge is hundreds of years old. It stretches across the Thames River. Robert McCulloch saw this bridge and decided to bring it to the USA.

He paid more than two million dollars. It will cost him more than three million dollars to move it. Each stone will be marked. The pieces must fit when they reach their new home. All that work will not take place overnight. The job will take six years. The bridge is not small. It is longer than three football fields. It is almost as wide as one football field. In time, the London Bridge will stand high above a new river. Flags will be placed at both ends. Cars will cross it. A small town will be built next to the bridge. Most people in Arizona will never see London. But they will see a part of it in their own state.

1. Who bought a bridge?
2. Where will the bridge be re-built?
3. How long will it take?

Syntactic parsing, shallow semantic analysis (argument identification)

Why is NLP hard?



Variability:

He drew the house

He made a sketch of the house

He showed me his drawing of the house

He portrayed the house in his paintings

He drafted the house in his sketchbook

…

Ambiguity:

She drew a picture of herself

A cart drawn by two horses…

He drew crowds wherever he went …

The driver slowed as he drew even …

The officer drew a gun..

Ambiguity

Happens at different levels (syntax, lexical semantics, reference, discourse, …)

- Leads to a combinatorial explosion of possible analysis

  - Example with *3* preposition phrases, *5* interpretations:
    - *Put the block ((in the box on the table) in the kitchen)*
    - *Put the block (in the box (on the table in the kitchen))*
    - *Put ((the block in the box) on the table) in the kitchen.*
    - *Put (the block (in the box on the table)) in the kitchen.*
    - *Put (the block in the box) (on the table in the kitchen)*
  - A general case:

$$Cat_n = \binom{2n}{n} - \binom{2n}{n-1} \sim \frac{4^n}{n^{3/2}\sqrt{\pi}}$$

Catalan numbers

$$1, 2, 5, 14, 42, 132, 429, 1430, 4862, 16796, 58786, \ldots$$

Why is NLP hard?

Reason 3. **Sparse data** due to **Zipf's Law**.

- To illustrate, let's look at the frequencies of different words in a large text corpus.
- Assume a "word" is a string of letters separated by spaces (a great oversimplification, we'll return to this issue...)

Word Counts

Most frequent words (word **types**) in the English Europarl corpus (out of 24m word **tokens**)

any word

Frequency	Type
1,698,599	the
849,256	of
793,731	to
640,257	and
508,560	in
407,638	that
400,467	is
394,778	a
263,040	I

nouns

Frequency	Type
104,325	Mr
92,195	Commission
66,781	President
62,867	Parliament
57,804	Union
53,683	report
53,547	Council
45,842	States

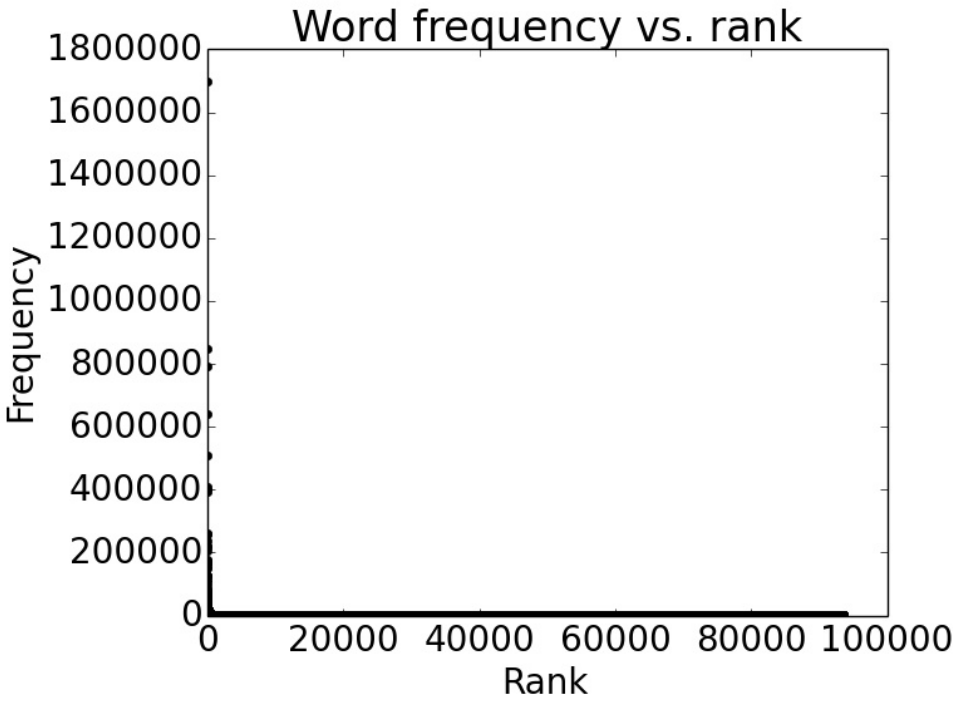
Word Counts

But also, out of 93638 distinct word types, 36231 occur only once.
Examples:

- cornflakes, mathematicians, fuzziness, jumbling
- pseudo-rapporteur, lobby-ridden, perfunctorily,
- Lycketoft, UNCITRAL, H-0695
- policyfor, Commissioneris, 145.95, 27a

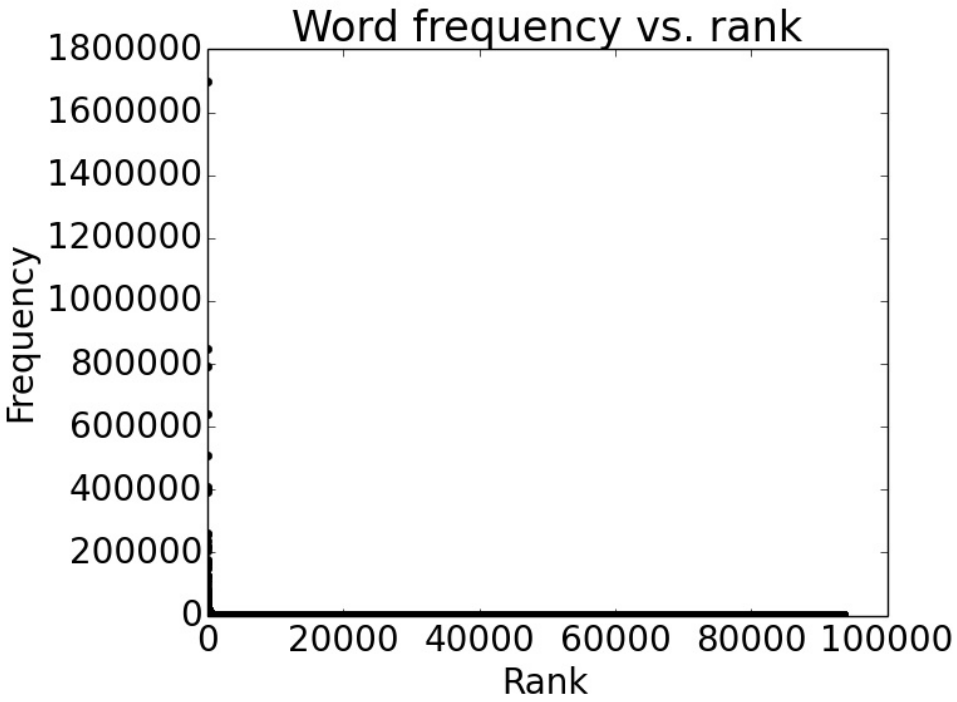
Plotting word frequencies

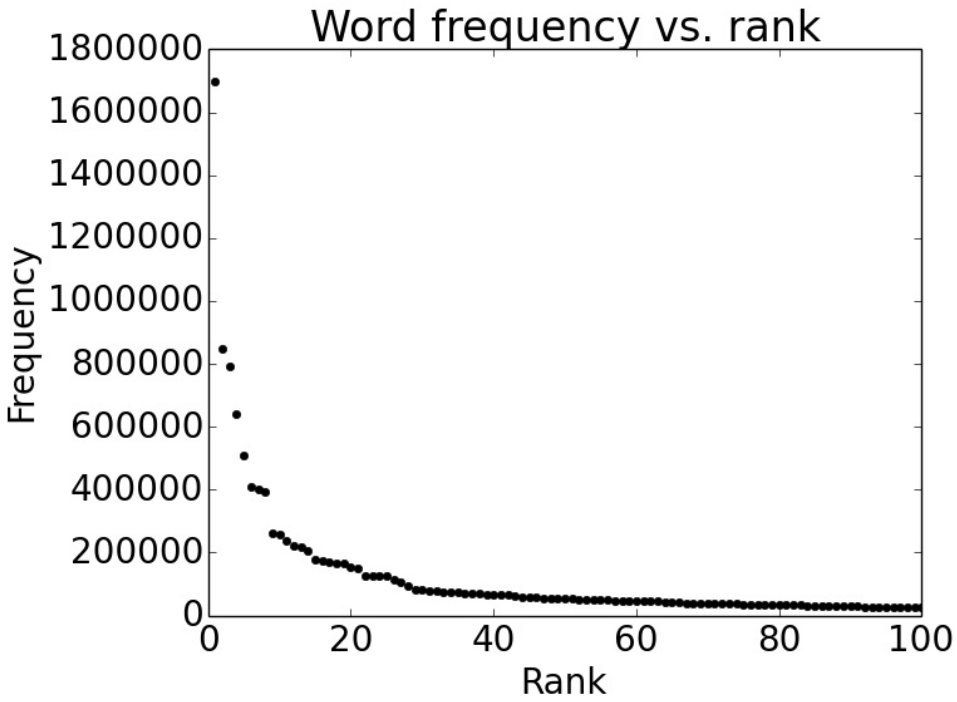
Order words by frequency. What is the frequency of nth ranked word?



Plotting word frequencies

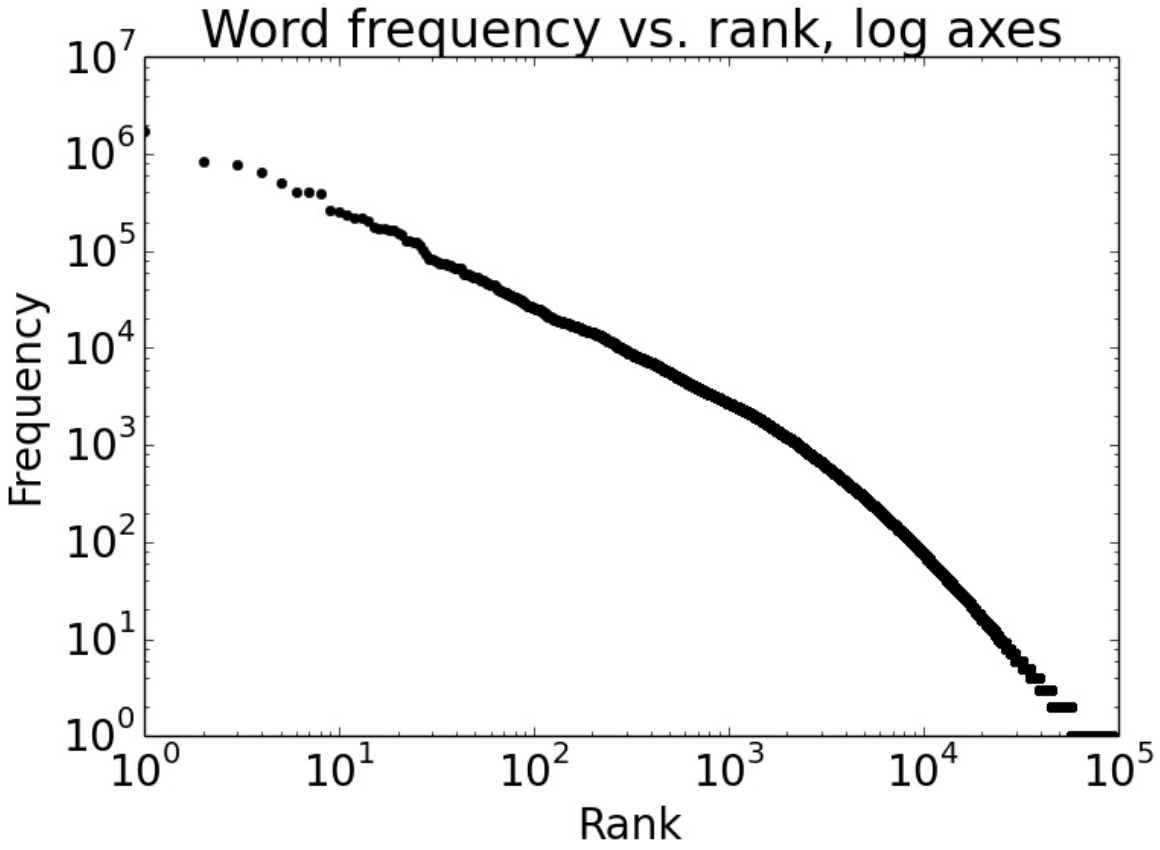
Order words by frequency. What is the frequency of nth ranked word?

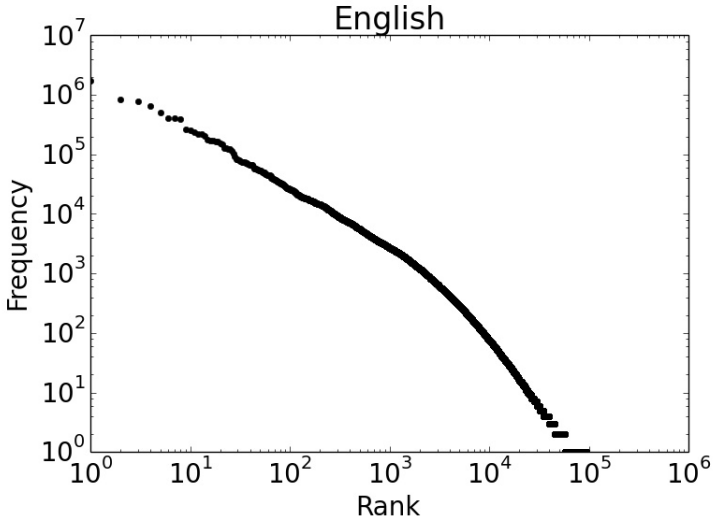


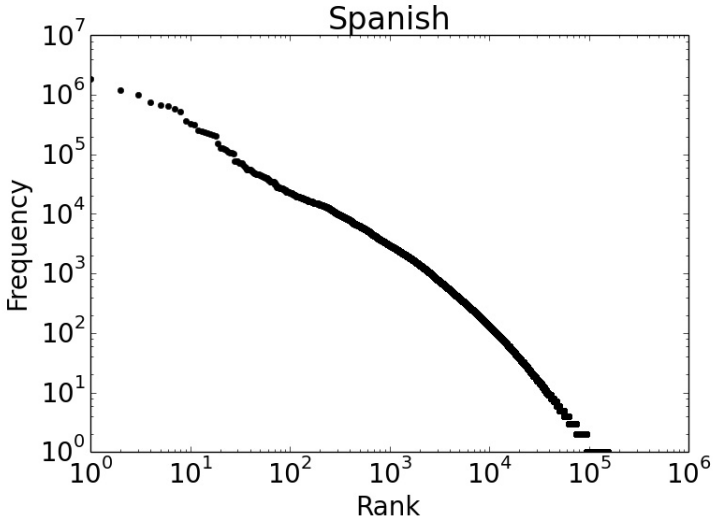


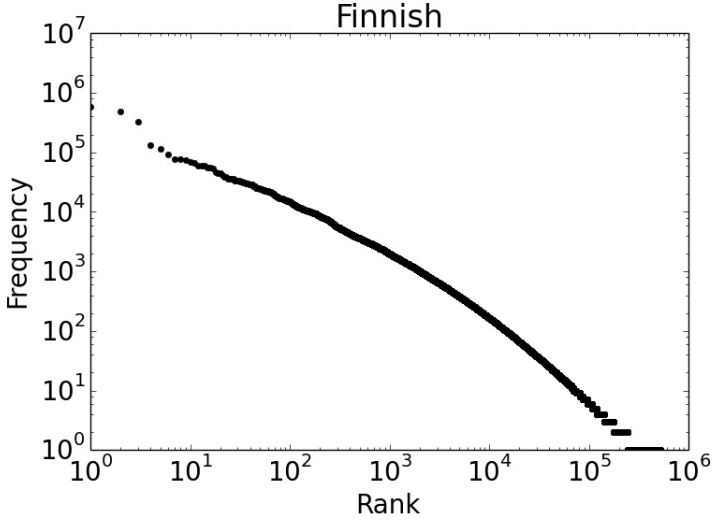
Rescaling the axes

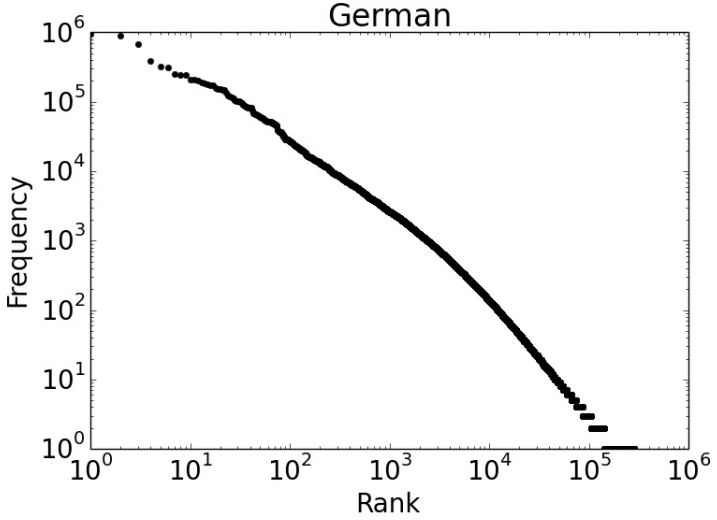
To really see what's going on, use logarithmic axes:




English
Frequency
Rank


Spanish
Frequency
Rank


Finnish
Frequency
Rank


German
Frequency
Rank

Zipf's law

Summarizes the behaviour we just saw:

$$f \times r \approx k$$

- $f =$ frequency of a word
- $r =$ rank of a word (if sorted by frequency)
- $k =$ a constant

Zipf's law

Summarizes the behaviour we just saw:

$$f \times r \approx k$$

- $f =$ frequency of a word
- $r =$ rank of a word (if sorted by frequency)
- $k =$ a constant

Why a line in log-scales? $fr = k \quad \Rightarrow \quad f = \frac{k}{r} \quad \Rightarrow \quad \log f = \log k - \log r$

Implications of Zipf's Law

- Any guesses?

Implications of Zipf's Law

- Regardless of how large our corpus is, there will be a lot of infrequent (and zero-frequency!) words.
- In fact, the same holds for many other levels of linguistic structure (e.g., syntactic rules in a CFG).
- Why is this a problem?

Implications of Zipf's Law

- Regardless of how large our corpus is, there will be a lot of infrequent (and zero-frequency!) words.
- In fact, the same holds for many other levels of linguistic structure (e.g., syntactic rules in a CFG).
- Why is this a problem?
  - This means we need to find clever ways to estimate probabilities for things we have rarely or never seen during training.

Why is NLP hard?

Reason 4. **Robustness**. In practice, the situation is often even trickier for NLP systems.

- Suppose we train a part of speech tagger on the Wall Street Journal:

  Mr./NNP Vinken/NNP is/VBZ chairman/NN of/IN Elsevier/NNP
  N.V./NNP ,/, the/DT Dutch/NNP publishing/VBG group/NN ./.

- What will happen if we try to use this tagger for social media??

  ikr smh he asked fir yo last name

- Does it have implications for fairness?

Twitter example due to Noah Smith

Why is NLP hard?

Robustness is a desired property for an NLP system

Relative grammaticality: processing input with questionable grammaticality:

"How many friends has Thomas painted a picture of?"

People disagree on how "grammatical" this utterance is.

Applications: in practical situations, language is often 'noisy', resulting in ambiguous input (as in speech recognition, spelling correction, etc.).

Why is NLP hard?

Reasons 5 and 6. **Context dependence** and **Unknown representation**

- Last example also shows that correct interpretation is context-dependent and often requires world knowledge.
- Very difficult to capture, since we don't even know how to represent the knowledge a human has/needs: What is the "meaning" of a word or sentence? How to model context? Other general knowledge?

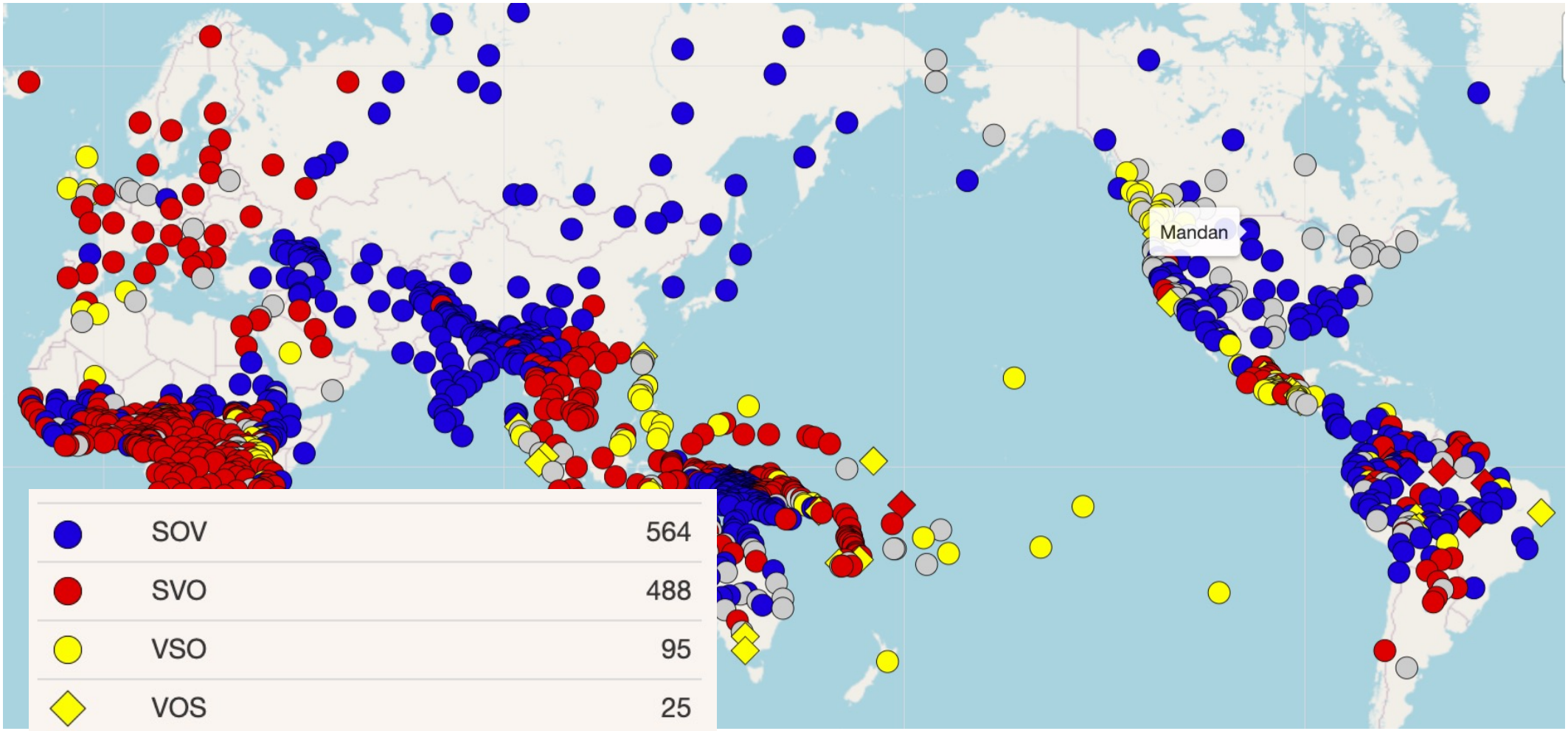
That is, in the limit NLP is hard because *AI* is hard

Why is NLP hard? Language Diversity

- In this class, we will focus on English
- ... as much of NLP community does
- However, we would like to develop technologies which work for other languages
- One example: syntactic diversity

From Benjamin Muller

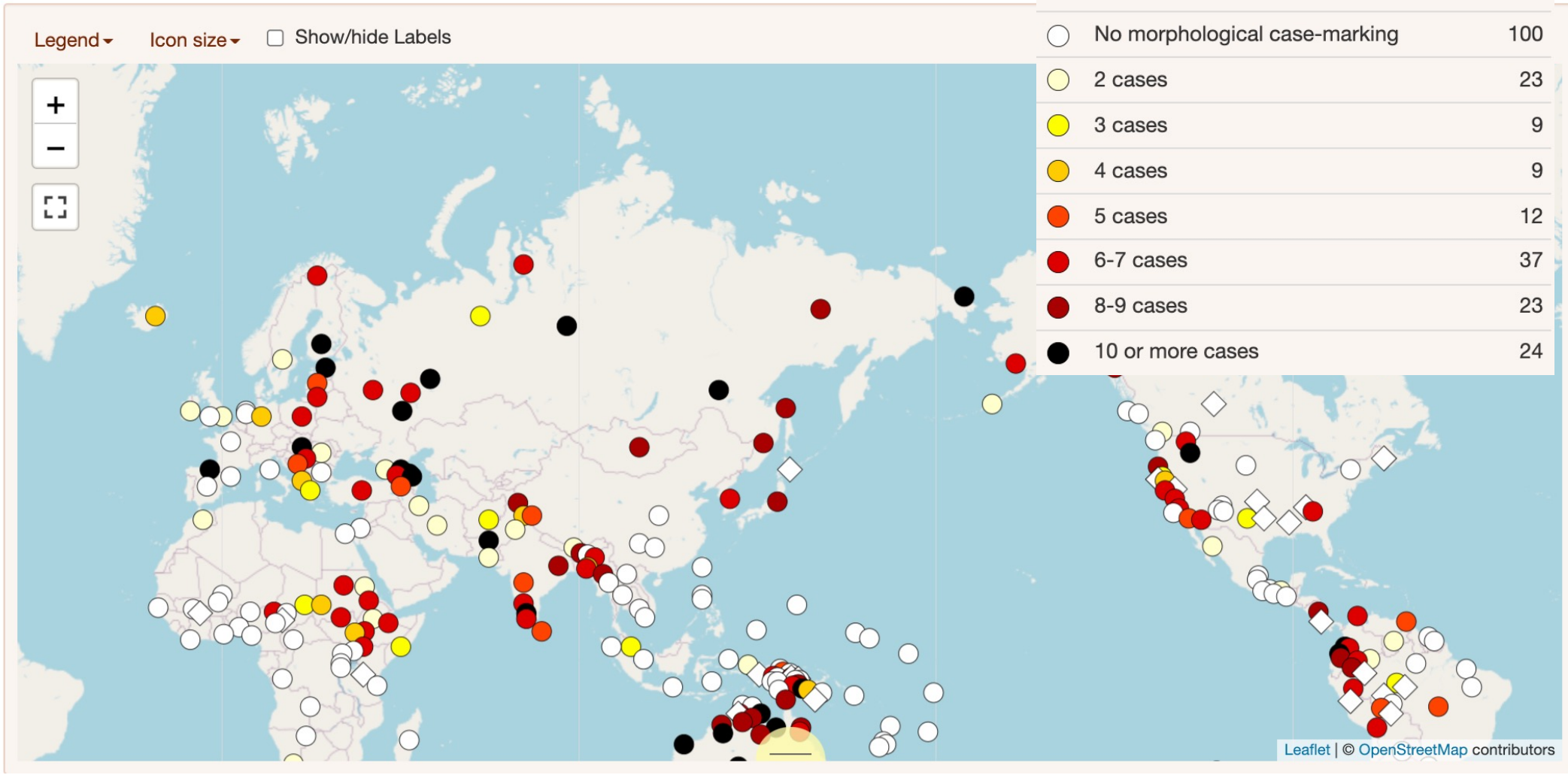
Syntactic diversity



(S)ubject, (V)erb, (O)bject order

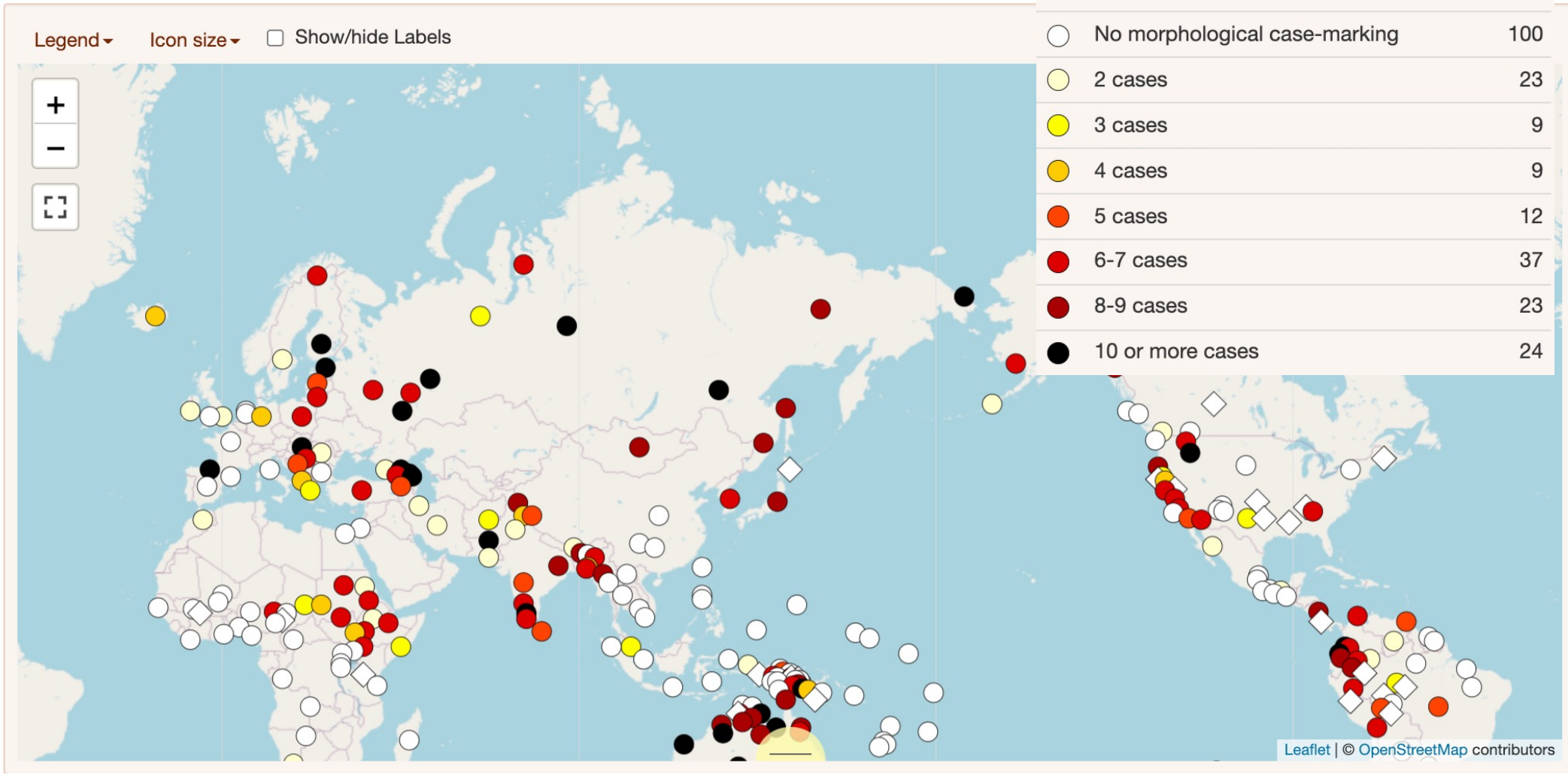
World Atlas of Language Structures (wals.info)

Number of cases



World Atlas of Language Structures (wals.info)

Number of cases



World Atlas of Language Structures (wals.info)

Morphology and Syntax

- Freedom in word order and morphology are inter-related
- The more freedom in the word order
  - The less information is conveyed by word positions
  - The more information should be included in the ”tokens”
  - The richer morphology

- E.g., English vs Russian (object marked with –ей)
  - Cats eat mice
  - Кошки едят мышей
  - Мышей едят кошки
  - Едят кошки мышей
  - Едят мышей кошки.

Constrained word order
Limited or no morphological marking

(Relatively) free word order
Rich morphology

Recap: Summary of challenges

1. ambiguity
2. variability
3. sparsity
4. robustness
5. context dependence
6. unknown representation
7. language diversity

Competence view on language

Formal language (set) theory as the tool:

A language	is	a set of word-sequences
A sentence	is	a sequence of words in the language
A grammar	is	a formal device defining the language
Grammaticality	is	a set membership test
Analyzing an utterance	is	assigning the correct structure

Is this view sufficient for NLP?

Manifistations of Uncertainty

Most challenges we discussed can be regarded as manifistations of uncertainty!

Ambiguity: uncertainty with respect to interpretation

Variability: uncertainty in a specific realization for a semantic concept

Robustness: uncertainty with respect to potential inputs

Lack of knowledge (cf sparsity / context dependence): uncertainty!

We need: probabilistic models / machine learning

Not everyone agrees

- Noam Chomsky in 1960: "it must be recognized that the notion of 'probability of a sentence' is an entirely useless one, under any known interpretation of this term"

- Even earlier in 1955: "Neither (a) 'colorless green ideas sleep furiously' nor (b) 'furiously sleep ideas green colorless', nor any of their parts, has ever occurred in the past linguistic experience of an English speaker. But (a) is grammatical, while (b) is not."

- Do you agree?

Let's check

- Fernando Pereira in 2001 showed that, even with an extremely simple ('ngram') model, the first sequence is much more likely than the second one

- (to be fair, 'useless' may have been used in a different sense, not from the engineering perspective)

Dealing with uncertainty / ambiguity

Inf2-iads started to discuss methods of dealing with ambiguity.

- non-probabilistic methods (FSMs for morphology, CKY parsers for syntax) return **all possible analyses**.

- probabilistic models (HMMs for POS tagging, PCFGs for syntax) and algorithms (Viterbi, probabilistic CKY) return the **best possible analysis**, i.e., the most probable one according to the model.

This "best" analysis is only good if our model's probabilities are accurate. Where do they come from?

Statistical NLP

Like most other parts of AI, NLP today is dominated by statistical methods.

- Typically more robust than earlier rule-based methods.
- Relevant statistics/probabilities are **learned from data** (cf. Inf2-fds).
- Normally requires **lots of data** about any particular phenomenon.

Brief history of statistical NLP

- 70s
  - Researchers at IBM introduced statistical approaches based on ideas from information theory to speech recognition
- 80s
  - First real successes in speech recognition
  - Researchers started to look into statistical methods for machine translation and other problems
- 90s
  - Success in syntactic parsing of natural language
  - First successes in machine translation – learning from parallel data (sentences and their translation)
  - Statistical models, often with strong linguistic biases (e.g., through features)
- 2010s - now
  - Neural networks
  - Huge pretrained models

Sketch of a probabilistic model

Functions: Inputs e.g. set of words+context / utterances,
Outputs e.g. set of POS-tags / syntactic analyzes,

Example: Part of Speech (PoS) tagging:

input	output		input	output
$input_1$	$output_1$		$\langle the, \underline{list} \rangle$	NN
$input_2$	$output_2$		$\langle We, \underline{list} \rangle$	VB
$\vdots$	$\vdots$		$\vdots$	$\vdots$

Model: A probability function over input–output pairs

$$P : Inputs \times Outputs \longrightarrow [0, 1]$$

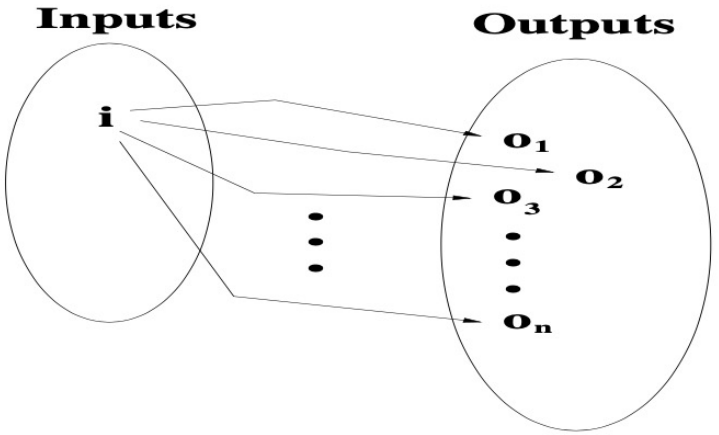
Sketch of a probabilistic model

Given $P : \mathit{Inputs} \times \mathit{Outputs} \longrightarrow [0, 1]$

Suppose $o_1, \cdots, o_n$ are possible outputs for input i

Q: how to select the preferred output o^* for input i?

A: select $o^* = \arg\max_{o \in \{o_1, \cdots, o_n\}} P(\langle i, o \rangle)$



$\arg\max_{a \in A} P(a)$: select that a from A for which $P(a)$ is maximal

Sketch of a probabilistic model

Questions	Potential answers
How to define set of input-output pairs?	Use formal grammars
How to define the probabilities $P(i, o)$?	Probabilistic grammars
How to obtain these probabilities?	Statistics from corpora
What (efficient) algorithms we need?	Technical solutions
How to measure success of a model?	Evaluation methods

Take-home messages

- There are many challenges which makes NLP hard
- Ambiguity is (?) the most fundamental one
  - happens at many levels
  - can lead to a combinatorial explosion in a number 'interpretations'
- Probabilistic modeling is a way to deal with many of these challenges
- Requires decisions at different levelts (probability models, algorithms, ...)

Where we are headed

- Focusing on **real data** with all its complexities.
- Probabilistic models
- Algorithms for learning (estimating probabilities) and inference (finding the most likely interpretation)
- Connections to linguistics and applications
- before that, we will discuss corpora (tomorrow) and evaluation methods (next Tuesday)