Appendix 3 to lecture 18 of “Machine Learning”

# of Language Technology (A Machine Learning Approach)

Antal van den Bosch & Walter Daelemans

antalb@uvt.nl
walter.daelemans@ua.ac.be

http://ilk.uvt.nl/~antalb/ltua

Programme

24/2	[Antal]	Intro ML for NLP & WEKA / Decision Trees
3/3	[Walter]	ML for shallow parsing
10/3	[Antal]	ML for morphology and phonology
17/3	[Antal]	ML for Information extraction
24/3	[Antal]	ML for discourse
31/3	[Véronique]	ML for coreference
21/4	[Antal]	Memory & Representation
28/4	[Antal]	Modularity / More Data
5/5	[Walter]	ML for document classification

Evaluation

- Assignments with different modules (fixed deadline)
- Final assignment

Language Technology overview

LT Components

Lexical / Morphological Analysis

Tagging

Chunking

Syntactic Analysis

Word Sense Disambiguation

Grammatical Relation Finding

Named Entity Recognition

Semantic Analysis

Reference Resolution

Discourse Analysis

Text

Meaning

Applications

OCR

Spelling Error Correction

Grammar Checking

Information retrieval

Document Classification

Information Extraction

Summarization

Question Answering

Ontology Extraction and Refinement

Dialogue Systems

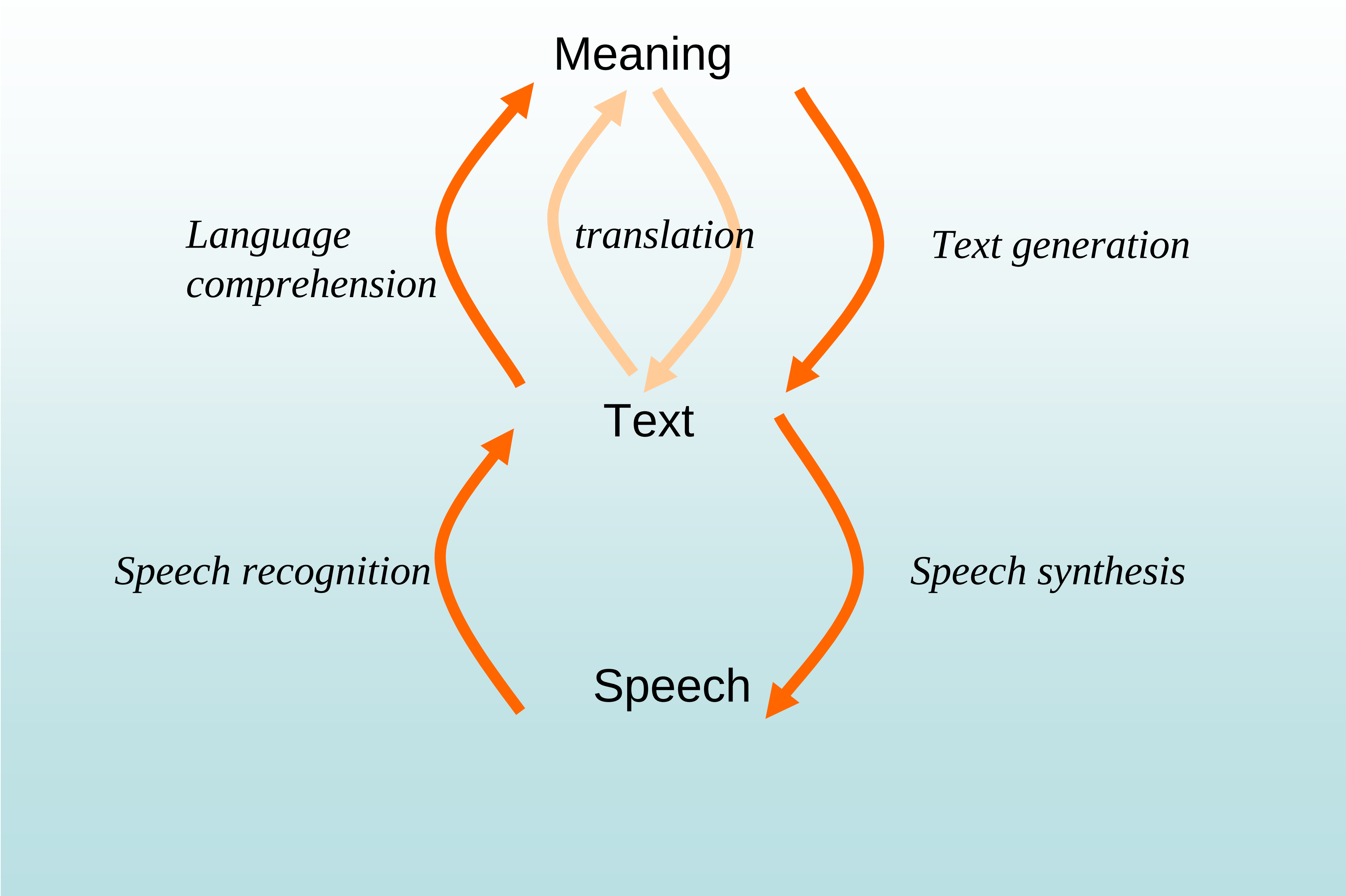
Machine Translation

Text Representation Units

- Character n-grams
- Words, phrases, heads of phrases
- POS tags
- Parse tree (fragment)s
- Grammatical Relations
- Frames and scripts
- “meaning” (?)

Text is a special kind of data

- Direct entry, OCR (.99 accuracy), Speech Recognition output (.50-.90 accuracy), ...
- What we have:
  - Characters, character n-grams, words, word n-grams, lay-out, counts, lengths, ...
- What we want:
  - Meaning (answering questions, relating with previous knowledge)
- Bridging the gap:
  - Tagging, lemmatization, phrase chunking, grammatical relations, ... I.e.: *Language Technology*


Meaning
Language comprehension
translation
Text generation
Text
Speech recognition
Speech synthesis
Speech

- Language Technology (Natural Language Processing, Computational Linguistics) is based on the complex transformation of linguistic representations
- Examples
  - from text to speech
  - from words to morphemes
  - from words to syntactic structures
  - from syntactic structures to conceptual dependency networks

- In this transformation, two processes play a role
  - segmentation of representations
  - disambiguation of possible transformations of representation units
- Similar representations at input level correspond to similar representations at the output level
- Complexity because of context-sensitivity (regularities, subregularities, exceptions)

gebruiksvriendelijkheid

ge+bruik+s+vriend+elijk+heid

Systran: Fremdzugehen -> External train marriages

The old man the boats

det N-plur V-plur det N-plur Punc

The old man the boats

(S (NP (DET the) (N old)) (VP (V man) (NP (DET the) (N boats))))

Systran: De oude man de boten

Systran: De prins bespreekt (zijn huwelijk) (met Verhofstadt)

The prince discusses (its marriage to Verhofstadt)

(S (NP (DET the) (N old)) (VP (V man) (NP (DET the) (N boats))))

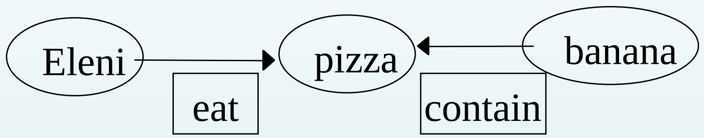
(man-action (agent (def plur old-person)) (object (def plur boat)))

How to reach Language Understanding ?

- A fundamental solution for the problem of language understanding presupposes
  - Representation and use of knowledge / meaning
  - Acquisition of human-level knowledge

What is meaning ?

Eleni eats a pizza with banana



Semantic networks, Frames

$\exists(x): \text{pizza}(x) \wedge \text{eat}(\text{Eleni},x) \wedge \text{contain}(x, \text{banaan})$

First-order predicate calculus

Pizza = {p1, p2, p3, …}
Eat ={<Nicolas,p1>,<Nicolas,p3>,<Eleni,p2>,…}
Contain ={<p1,ansjovis>,<p1,tomaat>,<p2,banaan>,…}
x=p2

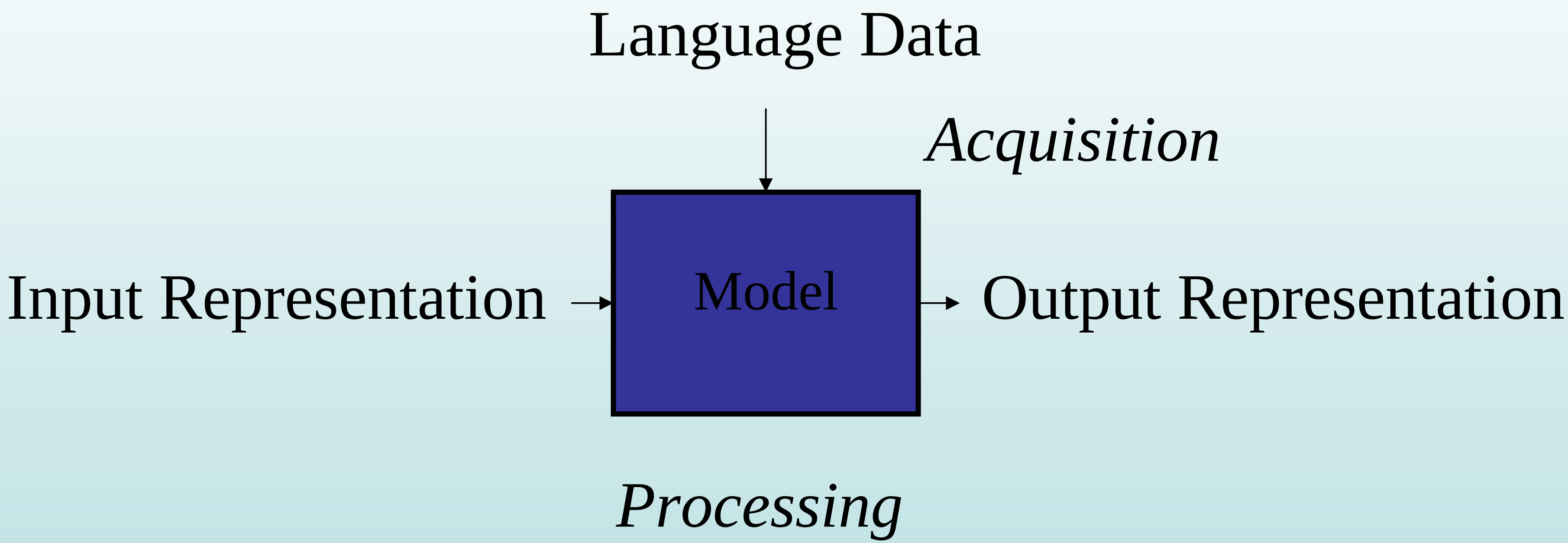
Set theory

"Symbol grounding" problem

Representation and processing of time, causality, modality, defaults, common sense, …

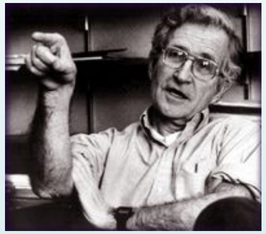
"Meaning is in the mind of the beholder"

Language Technology



Deductive Route

Language Data



Input Representation → Model → Output Representation

S -> NP VP
NP -> ART A N
NP -> ART N
VP -> V NP
...

Deductive Route

- Acquisition

  Construct a (rule-based) model about the domain of the transformation.

- Processing

  Use rule-based reasoning, deduction, on these models to solve new problems in the domain.

Inductive Route

Language Data

De computertaalkunde, in navolging van de taalkunde aanvankelijk sterk regel-gebaseerd, is onder druk van toepasbaarheid en grotere rekenkracht de laatste tien jaar geleidelijk geevolueerd naar een meer statistische, corpus-gebaseerde en inductieve benadering. De laatste jaren hebben ook technieken uit de theorie van zelflerende systemen (Machine Learning, zie Mitchell, 1998 voor een inleiding) aan belang gewonnen. Deze technieken zijn in zekere zin

Input Representation → Model → Output Representation

p(corpustaalkunde|de), p(in|corpustaalkunde),
p(corpustaalkunde), p(de), p(in), ...

Inductive Route

- Acquisition

  Induce a stochastic model from a corpus of "examples" of the transformation.

- Processing

  Use statistical inference (generalization) from the stochastic model to solve new problems in the domain.

Advantages

Deductive Route

- Linguistic knowledge and intuition can be used
- Precision

Inductive Route

- Fast development of model
- Good coverage
- Good robustness (preference statistics)
- Knowledge-poor
- Scalable / Applicable

Problems

Deductive Route

- Representation of sub/irregularity
- Cost and time of model development
- (Not scalable / applicable)

Inductive Route

- Sparse data
- Estimation of relevance statistical events

Text Mining

- Automatic extraction of reusable information (knowledge) from text, based on linguistic features of the text
- Goals:
  - Data mining (KDD) from unstructured and semi-structured data
  - (Corporate) Knowledge Management
  - “Intelligence”
- Examples:
  - Email routing and filtering
  - Finding protein interactions in biomedical text
  - Matching resumes and vacancies

Document

Set of Documents

Author Recognition
Document Dating
Language Identification
Text Categorization
Information Extraction
Summarization
Question Answering
Topic Detection and Tracking
Document Clustering
Terminology Extraction
Ontology Extraction

Structured Information

+ existing data

Knowledge Discovery

Information Extraction

- Analyzing unrestricted unstructured text
- Extracting specific structured information
- Enabling technology
  - Converting text to a database (data mining)
  - Summarization
- Compare:
  - Text Understanding
  - Information Retrieval

Example: MUC-terrorisme

Input:

- San Salvador, 19 Apr 89. Salvadoran President-elect Alfredo Cristiani. condemned the terrorist killing of Attorney general Roberto Garcia Alvarado and accused the Farabundo Marti National Liberation Front (FMLN) of the crime. (...)
- Garcia Alvarado, 56, was killed when a bomb placed by urban guerrillas on his vehicle exploded as it came to a halt at an intersection in downtown San Salvador.
- Vice President-elect Francisco Merino said that when the attorney-general's car stopped at a light on a street in downtown San Salvador, an individual placed a bomb on the roof of the armored vehicle. (...)
- According to the police and Garcia Alvarado's driver, who escaped unscathed, the attorney general was traveling with two bodyguards. One of them was injured.

Output template:

- Incident: Date 19 APR 89
- Incident: Location El Salvador: San Salvador
- Incident: Type Bombing
- Perpetrator: Individual ID urban guerrillas
- Perpetrator: Organization ID FMLN
- Perpetrator: Organization conf suspected or accused
- Physical target: description vehicle
- Physical target: effect some damage
- Human target: name Roberto Garcia Alvarado
- Human target: description attorney general Alvarado, driver, bodyguards
- Human target: effect death: alvarado,no injury: driver, injury: bodyguards

IEX System Architecture

- Local text analysis
  - Lexical analysis
    - tokenization, tagging, lemmatization
  - Named Entity Recognition
    - person name, company name, time expression, ...
  - Shallow Parsing (phrases and relations)
- Extraction
  - Pattern matching of simple facts
  - Integration of extracted facts into
    - Larger facts (reference resolution)
    - Additional facts (inference)
- Output template generation

Question Answering

- Give answer to question
  (document retrieval: find documents relevant to query)
- Who invented the telephone?
  - Alexander Graham Bell
- When was the telephone invented?
  - 1876

QA System: Shapaqa

- Parse question
  When was the telephone invented?
  - Which slots are given?
    - Verb invented
    - Object telephone
  - Which slots are asked?
    - Temporal phrase linked to verb
- Document retrieval on internet with given slot keywords
- Parsing of sentences with all given slots
- Count most frequent entry found in asked slot (temporal phrase)

Shapaqa: example

- *When was the telephone invented?*
- Google: invented AND "the telephone"
  - produces 835 pages
  - 53 parsed sentences with both slots and with a temporal phrase

is through his interest in Deafness and fascination with acoustics that the telephone was invented in 1876 , with the intent of helping Deaf and hard of hearing

The telephone was invented by Alexander Graham Bell in 1876

When Alexander Graham Bell invented the telephone in 1876 , he hoped that these same electrical signals could

Shapaqa: example (2)

- So when was the phone invented?
- Internet answer is noisy, but robust
  - 17: 1876
  - 3: 1874
  - 2: ago
  - 2: later
  - 1: Bell
  - ...

- System was developed quickly
- Precision 76% (Google 31%)
- International competition (TREC): MRR 0.45

- Silence -

Empiricism, analogy, induction, language

- A lightweight historical overview

- De Saussure:

  Any creation [of a language utterance] must be preceded by an unconscious comparison of the material deposited in the storehouse of language, where productive forms are arranged according to their relations. (1916, p. 165)

Lightweight history (2)

- Bloomfield:

  The only useful generalizations about language are inductive generalizations. (1933, p. 20).

- Zipf:

  $nf^2=k$ (1935), $rf=k$ (1949)

Lightweight history (3)

- Harris:

  With an apparatus of linguistic definitions, the work of linguistics is reducible […] to establishing correlations. […] And correlations between the occurrence of one form and that of other forms yield the whole of linguistic structure. (1940)

- Hjelmslev:

  Induction leads not to constancy but to accident. (1943)

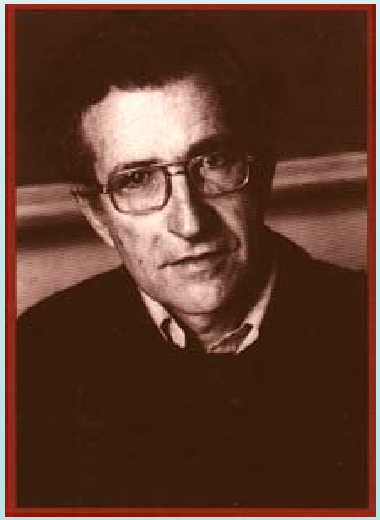
Lightweight history (4)

- Firth:

  A [linguistic] theory derives its usefulness and validity from the aggregate of experience to which it must continually refer. (1952, p. 168)

- Chomsky:

  I don't see any way of explaining the resulting final state [of language learning] in terms of any proposed general developmental mechanism that has been suggested by artificial intelligence, sensorimotor mechanisms, or anything else. (in Piatelli-Palmarini, 1980, p. 100)



Lightweight history (5)

- Halliday:

  The test of a theory [on language] is: does it facilitate the task at hand? (1985)

- Altmann:

  After the blessed death of generative linguistics, a linguist does no longer need to find a competent speaker. Instead, he needs to find a competent statistician. (1997)

Analogical memory-based language processing

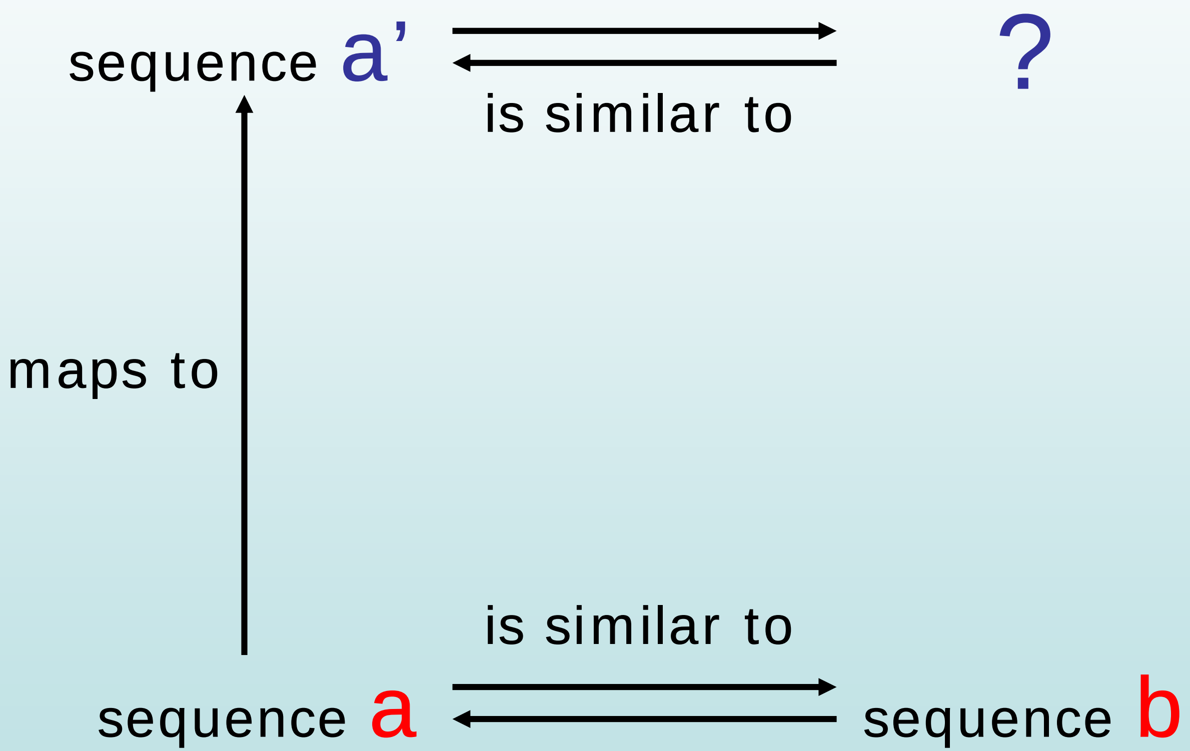
- With a memory filled with instances of language mappings
  - from text to speech,
  - from words to syntactic structure,
  - from utterances to acts, …
- With the use of analogical reasoning,
- Process new instances from input
  - text, words, utterances

  to output
  - speech, syntactic structure, acts

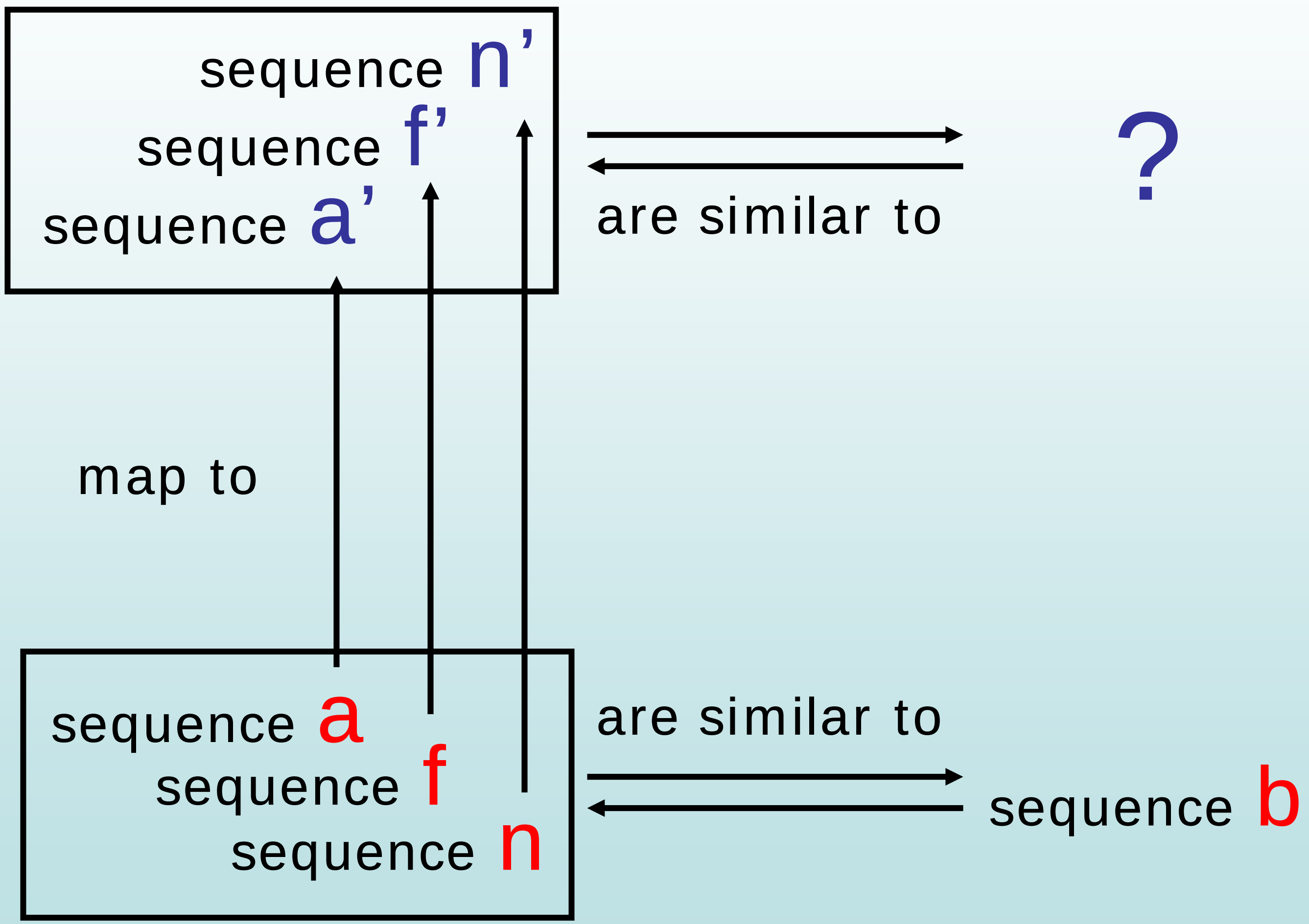
Analogy (1)

$$\begin{tikzcd}[column sep=huge, row sep=huge]
\text{sequence } a' \arrow[r, shift left] & \text{sequence } b' \arrow[l, shift left, "\text{is similar to}"] \\
\text{sequence } a \arrow[u, "\text{maps to}"] \arrow[r, shift left, "\text{is similar to}"] & \text{sequence } b \arrow[l, shift left] \arrow[u, "\text{maps to}"']
\end{tikzcd}$$

Analogy (2)


sequence a'
?
is similar to
maps to
is similar to
sequence a
sequence b

Analogy (3)



sequence n'
sequence f'
sequence a'
are similar to
?
map to
sequence a
sequence f
sequence n
are similar to
sequence b

Memory-based parsing

zo werd het Grand een echt theater

... **zo**$^{MOD/S}$ wordt er ...

... zo **gaat**$^{HD/S}$ het ...

... en dan werd **[**$_{NP}$ **het**^{DET} <*> dus ...

... dan is het **<*Naam>**$^{HD/SUBJ}$ $_{NP}$**]** bijna erger ...

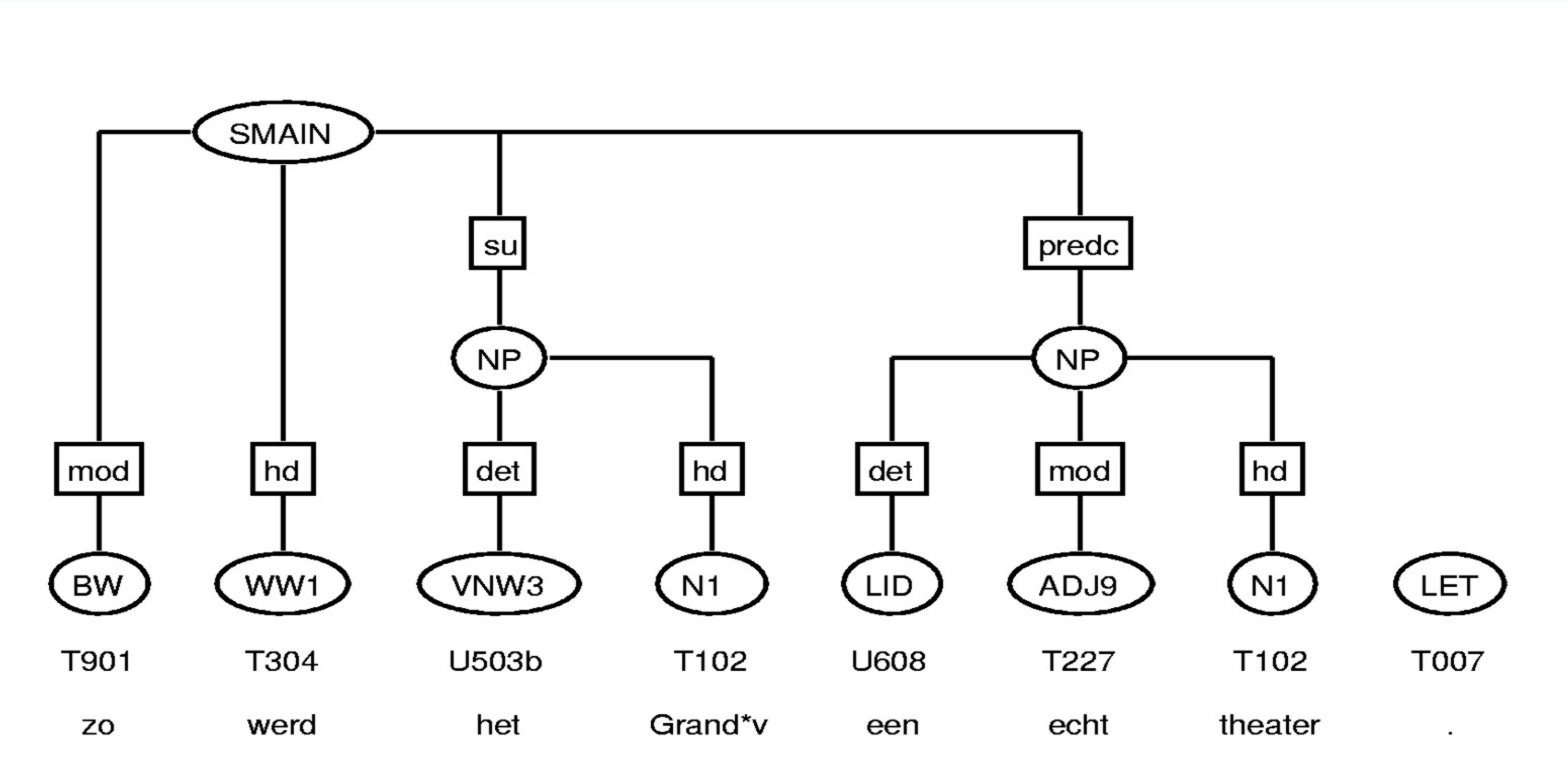
... ergens ene keer **[**$_{NP}$ **een**^{DET} echt <*> ...

... ben ik een **echt**^{MOD} <*> maar ...

... een echt **bedrijf**$^{HD/PREDC}$ $_{NP}$ **]**

zo$^{MOD/S}$ werd$^{HD/S}$ [$_{NP}$ het^{DET} Grand$^{HD/SUBJ}$ $_{NP}$]
[$_{NP}$ een^{DET} echt^{MOD} theater$^{HD/PREDC}$ $_{NP}$]

CGN treebank



Make data (1)

```
#BOS 54 2 1011781542 0
zo                        BW      T901             MOD     502
werd                      WW1     T304             HD      502
het                       VNW3    U503b            DET     500
Grand*v                   N1      T102             HD      500
een                       LID     U608             DET     501
echt                      ADJ9    T227             MOD     501
theater                   N1      T102             HD      501
.                         LET     T007             --      0
#500                      NP      --               SU      502
#501                      NP      --               PREDC   502
#502                      SMAIN   --               --      0
#EOS 54
```

Make data (2)

- Given context, map individual words to function+chunk code:

1. zo	MOD	O
2. werd	HD	O
3. het	DET	B-NP
4. Grand	HD/SU	I-NP
5. een	DET	B-NP
6. echt	MOD	I-NP
7. theater	HD/PREDC	I-NP

Make data (3)

- Generate instances with context:

1. _ _ _ **zo** werd het Grand MOD-O
2. _ _ zo **werd** het Grand een HD-O
3. _ zo werd **het** Grand een echt DET-B-NP
4. zo werd het **Grand** een echt theater HD/SU-I-NP
5. werd het Grand **een** echt theater _ DET-B-NP
6. het Grand een **echt** theater _ _ MOD-I-NP
7. Grand een echt **theater** _ _ _ HD/PREDC-I-NP

Crash course: Machine Learning

The field of machine learning is concerned with the question of how to construct computer programs that automatically learn with experience. (Mitchell, 1997)

- Dynamic process: learner L shows improvement on task T *after* learning.
- Getting rid of programming.
- Handcrafting versus learning.
- Machine Learning is **task-independent**.

Machine Learning: Roots

- Information theory
- Artificial intelligence
- Pattern recognition
- Took off during 70s
- Major algorithmic improvements during 80s
- Forking: neural networks, data mining

Machine Learning: 2 strands

- Theoretical ML (what can be proven to be learnable by what?)
  - Gold, *identification in the limit*
  - Valiant, *probably approximately correct learning*

- Empirical ML (on real or artificial data)
  - Evaluation Criteria:
    - Accuracy
    - Quality of solutions
    - Time complexity
    - Space complexity
    - Noise resistance

Empirical ML: Key Terms 1

- Instances: individual examples of input-output mappings of a particular type
- Input consists of features
- Features have values
- Values can be
  - Symbolic (e.g. letters, words, ...)
  - Binary (e.g. indicators)
  - Numeric (e.g. counts, signal measurements)
- Output can be
  - Symbolic (classification: linguistic symbols, ...)
  - Binary (discrimination, detection, ...)
  - Numeric (regression)

Empirical ML: Key Terms 2

- A set of instances is an instance base
- Instance bases come as labeled training sets or unlabeled test sets (you know the labeling, not the learner)
- A ML experiment consists of training on the training set, followed by testing on the disjoint test set
- Generalisation performance (accuracy, precision, recall, F-score) is measured on the output predicted on the test set
- Splits in train and test sets should be systematic: n-fold cross-validation
  - 10-fold CV
  - Leave-one-out testing
- Significance tests on pairs or sets of (average) CV outcomes

Empirical ML: 2 Flavours

- Greedy
  - Learning
    - abstract model from data
  - Classification
    - apply abstracted model to new data
- Lazy
  - Learning
    - store data in memory
  - Classification
    - compare new data to data in memory

Greedy learning

QuickTime™ and a GIF decompressor are needed to see this picture.

Greedy learning

QuickTime™ and a GIF decompressor are needed to see this picture.

Lazy Learning

QuickTime™ and a GIF decompressor are needed to see this picture.

Lazy Learning

QuickTime™ and a GIF decompressor are needed to see this picture.

Greedy vs Lazy Learning

Greedy:

- Decision tree induction
  - CART, C4.5
- Rule induction
  - CN2, Ripper
- Hyperplane discriminators
  - Winnow, perceptron, backprop, SVM
- Probabilistic
  - Naïve Bayes, maximum entropy, HMM
- (Hand-made rulesets)

Lazy:

- *k*-Nearest Neighbour
  - MBL, AM
  - Local regression

Greedy vs Lazy Learning

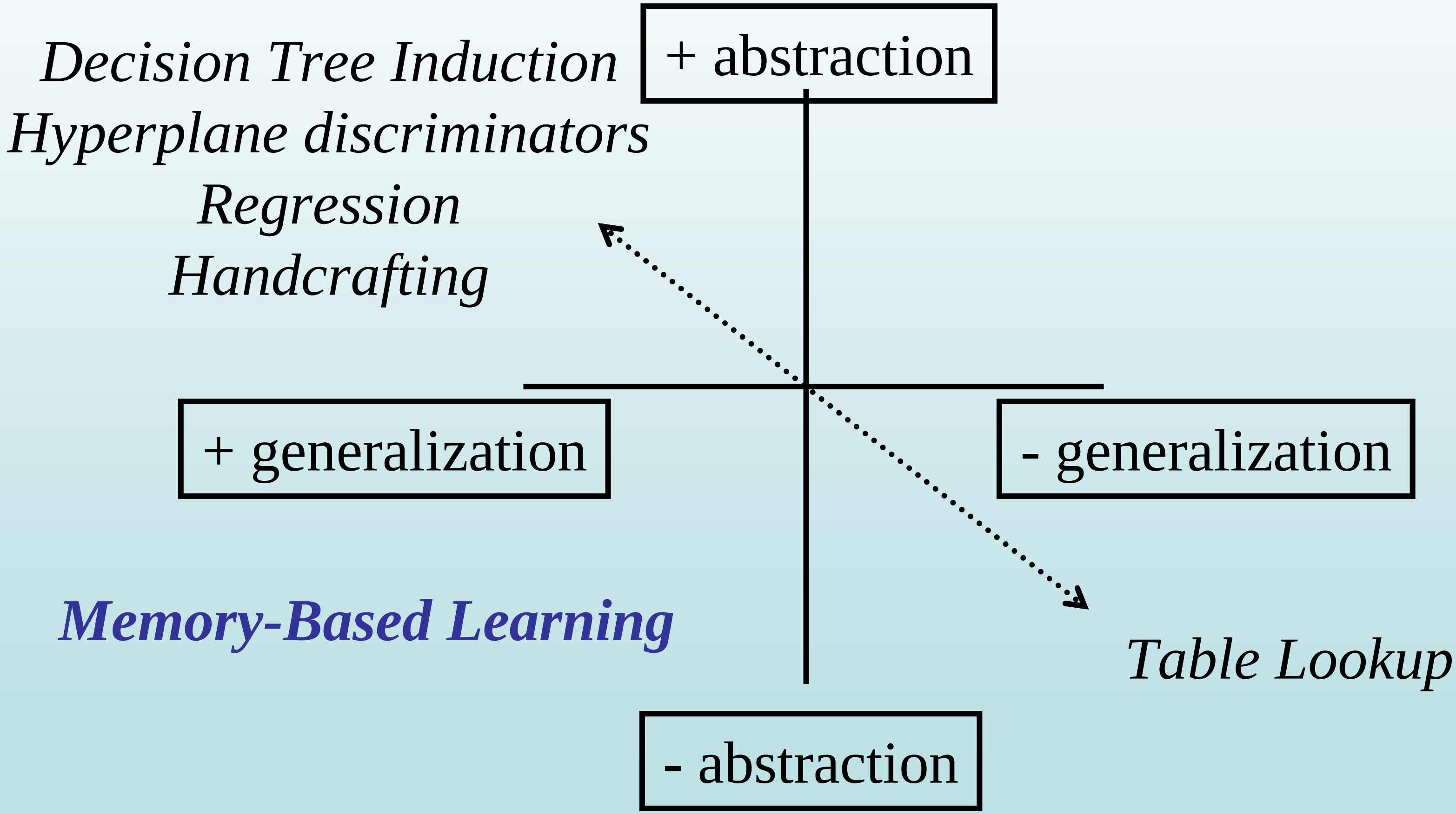
- Decision trees keep the smallest amount of informative decision boundaries (in the spirit of MDL, Rissanen, 1983)
- Rule induction keeps smallest number of rules with highest coverage and accuracy (MDL)
- Hyperplane discriminators keep just one hyperplane (or vectors that support it)
- Probabilistic classifiers convert data to probability matrices

- k-NN retains every piece of information available at training time

Greedy vs Lazy Learning

- Minimal Description Length principle:
  - Ockham's razor
  - Length of abstracted model (covering core)
  - Length of productive exceptions not covered by core (periphery)
  - Sum of sizes of both should be minimal
  - More minimal models are better
- "Learning = compression" dogma
- In ML, length of abstracted model has been focus; not storing periphery

Greedy vs Lazy Learning



Greedy vs Lazy: So?

- Highly relevant to ML of NL
- In language data, what is core? What is periphery?
- Often little or no noise; productive exceptions
- (Sub-)subregularities, pockets of exceptions
- “disjunctiveness”
- Some important elements of language have different distributions than the “normal” one
- E.g. word forms have a Zipfian distribution

ML and Natural Language

- Apparent conclusion: ML could be an interesting tool to do linguistics
  - Next to probability theory, information theory, statistical analysis (natural allies)
  - "Neo-Firthian" linguistics
- More and more annotated data available
- Skyrocketing computing power and memory

Entropy & IG: Formulas

$$H(D) = -\sum_i p_i log_2 p_i$$

$$H(D_{[f_i]}) = \sum_{v_j \in V} H(D_{[f_i = v_j]}) \frac{|D_{[f_i = v_j]}|}{|D|}$$

$$G(f_i) = H(D) - H(D_{[f_i]})$$