

# Computerlinguistik II

Vorlesung im SoSe 2018  
(M-GSW-10)

**Prof. Dr. Udo Hahn**

Lehrstuhl für Computerlinguistik  
Institut für Germanistische Sprachwissenschaft  
Friedrich-Schiller-Universität Jena

<http://www.julielab.de>

# Allgemeine Hinweise

- Vorlesung: Mi, 10-12h (Humboldt 8, SR 1)
- Übung zV: Mo 10-12h (Johannisfr.hof.3, SR 2)
  - beginnt am **16.4.2018**
- Vorlesungsmaterialien im Netz
  - <http://www.julielab.de/> ⇒ „Students“
- **M-GSW-10 besteht aus VL+ÜB und Seminar!**
- Sprechstunde: Mi, 12-13h (bA) (FG 30, R 004)
- Email: [udo.hahn@uni-jena.de](mailto:udo.hahn@uni-jena.de)
- URL: <http://www.julielab.de>
- Fachliteratur ist überwiegend in Englisch

# Veranstaltungen im SS 2018

- **Seminar „Linked Open Data“**
  - Do, 16-18 Uhr
- **Software-Praktikum: „Softwaretechnologien für Natürlichsprachliche Systeme“**
  - Di, 14-16 Uhr
- **Theoreticum: Methoden der Computerlinguistik,**
- **Technicum: Praxis sprachtechnologischer Systeme**
  - Alterierend: Fr, 10-12 Uhr

# Two Paradigms for NLP

- Symbolic Specification Paradigm
  - Manual acquisition procedures
  - Lab-internal activities
  - Intuition and (few!) subjectively generated examples drive progress based on individual (competence) judgments
    - “I have a system that parses all of my nine-teen sentences!”

# Symbolic Specification Paradigm

- **Manual rule specification**
  - Source: linguist's intuition
- **Manual lexicon specification**
  - Source: linguist's intuition
- **Each lab has its own (home-grown) set of NLP software**
  - Hampers reusability
  - Limits scientific progress
  - Waste of human and monetary resources (we “burnt” thousands of Ph.D. student all over the world 😞)

# Shortcomings of the “Classical” Linguistic Approach

- Huge amounts of background knowledge req.
  - Lexicons (approx. 100,000 – 150,000 entries)
  - Grammars (>> 15,000 – 20,000 rules)
  - Semantics (>> 15,000 – 20,000 rules)
- As the linguistic and conceptual coverage of classical linguistic systems increases (slowly), it still remains insufficient; systems also reveal ‘spurious’ ambiguity, and, hence, tend to become overly “brittle” and unmaintainable
- More fail-soft behavior is required at the expense of ... ? (e.g., full-depth understanding)

# Two Paradigms for NLP

## Symbolic Specification Paradigm

- Manual acquisition procedures
- Lab-internal activities
- Intuition and (few!) subjectively generated examples drive progress based on individual (competence) judgments
  - “I have a system that parses all of my nine-teen sentences!”

## • Empirical (Learning) Paradigm

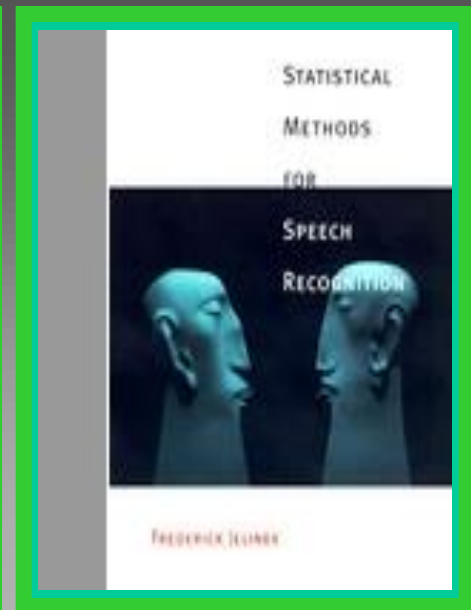
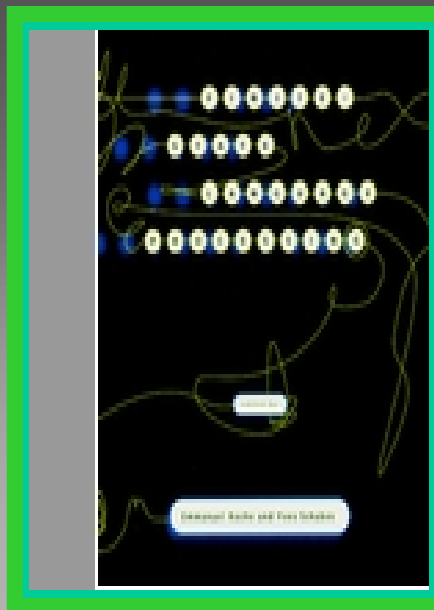
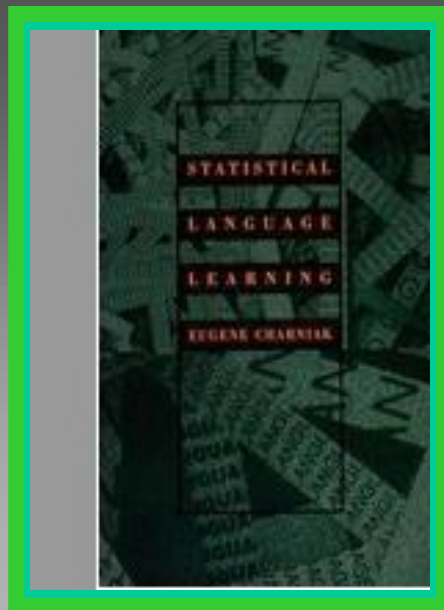
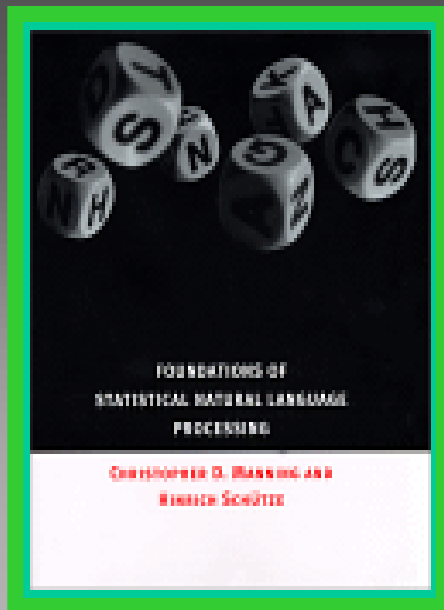
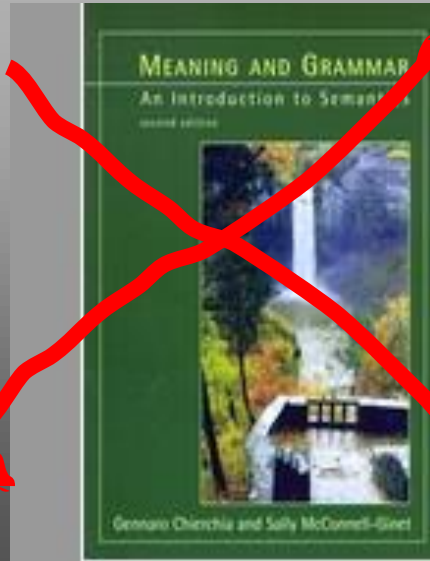
- Automatic acquisition procedures
- Community-wide sharing of common knowledge and resources
- Large and ‘representative’ data sets drive progress according to experimental standards
  - “The system was tested on 1,7 million words taken from the WSJ segment of the MUC-7 data set and produced 4.9% parsing errors, thus yielding a statistically significant 1.6% improvement over the best result by parser X on the same data set & a 40.3% improvement over the baseline system!”

# Empirical Paradigm

- **Large repositories of language data**
  - Corpora (plain or annotated, i.e., enriched by meta-data)
- **Large, community-wide shared repositories of language processing modules**
  - Tokenizers, POS taggers, chunkers, NE recognizers, ...
- **Shared repositories of machine learning algos**
- **Automatic acquisition of linguistic knowledge**
  - Applying ML algos to train linguistic processors by using large corpora with valid linguistic metadata (linguist as educated data supplier, „language expert“) rather than manual intuition (linguist as creative rule inventor)
- **Shallow analysis rather than deep understanding**
- **Large, community-wide self-managed, task-oriented competitions, comparative evaluation rounds**
- **Change of mathematics:**
  - Statistics rather than algebra and logics

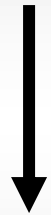


# Paradigm Shift – We Exchanged our Textbooks...



# POS Tagging

A severe infection ended the pregnancy .



DET

ADJ

NOUN

VERB

DET

NOUN

ST

# Penn Treebank Tag Set

Tag	Description	Examples
.	sentence terminator	. ! ?
DT	determiner	all an many such that the them these this
JJ	adjective, numeral	first oiled separable battery-powered
NN	common noun	cabbage thermostat investment
PRP	personal pronoun	herself him it me one oneself theirs they
IN	preposition	among out within behind into next
VB	verb (base form)	ask assess assign begin break bring
VBD	verb (past tense)	asked assessed assigned began broke
WP	WH-pronoun	that what which who whom

In total,  
45 tags

# Transformation Rules for Tagging [Brill, 1995]

- Initial State: Based on a number of features, guess the most likely POS tag for a given word:
  - die/DET Frau/NOUN ,/COMMA die/DET singt/VFIN
- Learn transformation rules to reduce errors:
  - *Change DET to PREL whenever the preceding word is tagged as COMMA*
- Apply learned transformation rules:
  - die/DET Frau/NOUN, /COMMA die/PREL singt/VFIN

# First 20 Transformation Rules

#	Change Tag		Condition
	From	To	
1	NN	VB	Previous tag is <i>TO</i>
2	VBP	VB	One of the previous three tags is <i>MD</i>
3	NN	VB	One of the previous two tags is <i>MD</i>
4	VB	NN	One of the previous two tags is <i>DT</i>
5	VBD	VBN	One of the previous three tags is <i>VBZ</i>
6	VBN	VBD	Previous tag is <i>PRP</i>
7	VBN	VBD	Previous tag is <i>NNP</i>
8	VBD	VBN	Previous tag is <i>VBD</i>
9	VBP	VB	Previous tag is <i>TO</i>
10	POS	VBZ	Previous tag is <i>PRP</i>
11	VB	VBP	Previous tag is <i>NNS</i>
12	VBD	VBN	One of previous three tags is <i>VBP</i>
13	IN	WDT	One of next two tags is <i>VB</i>
14	VBD	VBN	One of previous two tags is <i>VB</i>
15	VB	VBP	Previous tag is <i>PRP</i>
16	IN	WDT	Next tag is <i>VBZ</i>
17	IN	DT	Next tag is <i>NN</i>
18	JJ	NNP	Next tag is <i>NNP</i>
19	IN	WDT	Next tag is <i>VBD</i>
20	JJR	RBR	Next tag is <i>JJ</i>

Taken from: Brill (1995), Transformation-Based Error-Driven Learning

# Towards Statistical Models of Natural Language Processing ...

# Letter-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
-

# Letter-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
- **W**



# Letter-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
- **Wh**

# Letter-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
- **Wha**

# Letter-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
- **What**

# Letter-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
- **What d**

# Letter-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
- **What do**

# Letter-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
- **What do you think the next letter is?**

# Word-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
  - **What do you think the next letter is?**
- **Guess the next word:**
-

# Word-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
- **What do you think the next letter is?**
- **Guess the next word:**
- **We**



# Word-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
- **What do you think the next letter is?**
- **Guess the next word:**
- **We are**

# Word-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
- **What do you think the next letter is?**
- **Guess the next word:**
- **We are now**

# Word-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
  - **What do you think the next letter is?**
- **Guess the next word:**
  - **We are now entering**

# Word-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
  - **What do you think the next letter is?**
- **Guess the next word:**
  - **We are now entering statistical**

# Word-based Language Models

- **Shannon's Game**
- **Guess the next letter:**
  - **What do you think the next letter is?**
- **Guess the next word:**
  - **We are now entering statistical territory**

# Approximating Natural Language Words

- **zero-order approximation:**  
letter sequences are independent of each other and all equally probable:
  - xfoml rxkhrjffjuj zlpwcwky  
ffjeyvkcqsghyd

# Approximating Natural Language Words

- **first-order approximation:**  
letters are independent, but occur  
with the frequencies of English text:
  - ocro hli rgwr nmielwis eu ll  
nbnesebya th eei alhenhtppa oobttva  
nah

# Approximating Natural Language Words

- **second-order approximation:**  
the probability that a letter appears depends on the previous letter
  - on ie antsoutinys are t inctore st bes  
deamy achin d ilonasive tucoowe at  
teasonare fuzo tizin andy tobe seace  
ctisbe



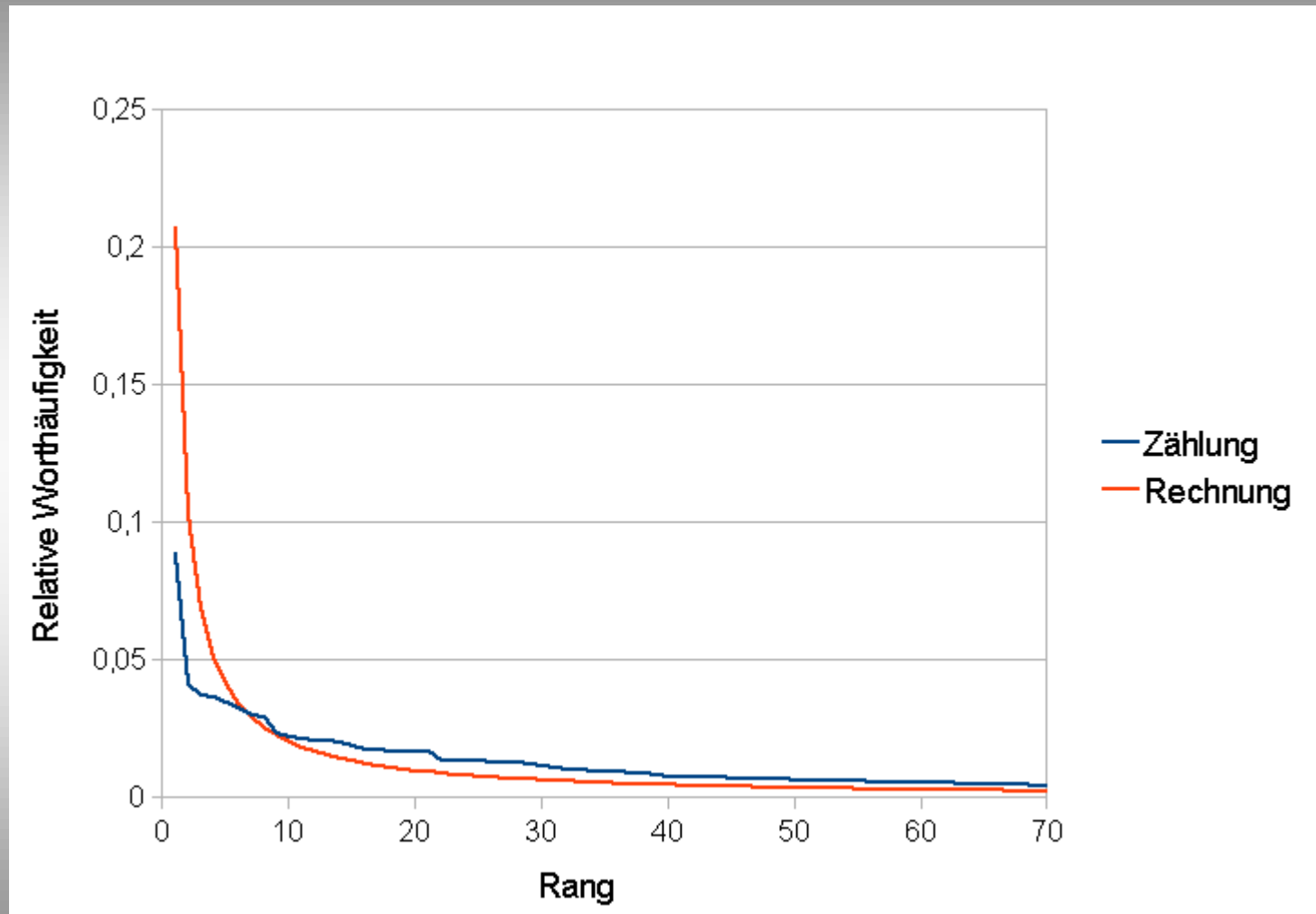
# Approximating Natural Language Words

- **third-order approximation:**  
the probability that a certain letter appears depends on the two previous letters
  - in no ist lat whey cratict froure birs  
groid pondenome of demonstures  
of the reptagin is regoactiona of cre

# Approximating Natural Language Words

- Higher frequency trigrams for different languages:
  - English: THE, ING, ENT, ION
  - German: EIN, ICH, DEN, DER
  - French: ENT, QUE, LES, ION
  - Italian: CHE, ERE, ZIO, DEL
  - Spanish: QUE, EST, ARA, ADO

# Zipfsches Gesetz



Wortverteilung im Vergleich zu einer einfachen Zipf-Verteilung ( $\sim 1/n$ . Wortanzahl: 70;  
Texte aus: <http://www.gutenberg.org/dirs/etext04/8effi10.txt>)

# Terminology

- **Sentence:** unit of written language
- **Utterance:** unit of spoken language
- **Word Form:** the inflected form that appears literally in the corpus
- **Lemma:** lexical forms having the same stem, part of speech, and word sense
- **Types (V):** number of distinct words that might appear in a corpus (vocabulary size)
- **Tokens ( $N_T$ ):** total number of words in a corpus (note:  $V \ll N_T$ )
- **Types seen so far (T):** number of distinct words seen so far in corpus (note:  $T \leq V \ll N_T$ )

# Word-based Language Models

- A model that enables one to compute the probability, or likelihood, of a sentence  $S$ ,  $P(S)$ .
- **Simple: Every word follows every other word with equal probability (0-gram)**
  - Assume  $|V|$  is the size of the vocabulary  $V$
  - Likelihood of sentence  $S$  of length  $n$  is  $1/|V| \times 1/|V| \dots \times 1/|V|$
  - If English has 100,000 words, the probability of each next word is  $1/100000 = .00001$

# Relative Frequency vs. Conditional Probability

- Smarter: *Relative* Frequency

Probability of each next word is related to word frequency within a corpus (unigram)

- Likelihood of sentence  $S = P(w_1) \times P(w_2) \times \dots \times P(w_n)$
- Assumes probability of each word is independent of probabilities of other words

# Relative Frequency vs. Conditional Probability

- Smarter: *Relative* Frequency

Probability of each next word is related to word frequency within a corpus (unigram)

- Likelihood of sentence  $S = P(w_1) \times P(w_2) \times \dots \times P(w_n)$
- Assumes probability of each word is independent of probabilities of other words

- Even smarter: *Conditional* Probability

Look at probability given previous words (n-gram)

- Likelihood of sentence  $S = P(w_1) \times P(w_2|w_1) \times \dots \times P(w_n|w_{n-1})$
- Assumes probability of each word is dependent on probabilities of previous words

# Generalization of Conditional Probability via Chain Rule

- **Conditional Probability for Two Events,  $A_1$  and  $A_2$** 
  - $P(A_1, A_2) = P(A_1) \cdot P(A_2|A_1)$
- **Chain Rule** generalizes to multiple ( $n$ ) events
  - $P(A_1, \dots, A_n) =$   
 $P(A_1) \times P(A_2|A_1) \times P(A_3|A_1, A_2) \times \dots \times P(A_n|A_1 \dots A_{n-1})$
- **Examples:**
  - $P(\text{the dog}) = P(\text{the}) \times P(\text{dog} | \text{the})$
  - $P(\text{the dog bites}) = P(\text{the}) \times P(\text{dog} | \text{the}) \times P(\text{bites} | \text{the dog})$



# Relative Frequencies and Conditional Probabilities

- Relative word frequencies are better than equal probabilities for all words
  - In a corpus with 10K word types, each word would have  $P(w) = 1/10K$
  - Does not match our intuitions that different words are more likely to occur
    - (e.g. “the” vs. “shop” vs. “aardvark”)
- Conditional probability is more useful than individual relative word frequencies
  - **dog** may be relatively rare in a corpus
  - but if we see **barking**,  $P(\text{dog}|\text{barking})$  may be large